Channel charting based beamforming

Luc Le Magoarou*†, Taha Yassine*†, Stéphane Paquelet* and Matthieu Crussière*†
*b<>com, Rennes, France
†Univ Rennes, INSA Rennes, CNRS, IETR - UMR 6164, F-35000 Rennes

***Abstract*—Channel charting (CC) is an unsupervised learning method allowing to locate users relative to each other without reference. From a broader perspective, it can be viewed as a way to discover a low-dimensional latent space charting the channel manifold. In this paper, this latent modeling vision is leveraged together with a recently proposed location-based beamforming (LBB) method to show that channel charting can be used for mapping channels in space or frequency. Combining CC and LBB yields a neural network resembling an autoencoder. The proposed method is empirically assessed on a channel mapping task whose objective is to predict downlink channels from uplink channels.**

***Index Terms*—channel charting, cell-less network, dimensionality reduction, MIMO signal processing, machine learning.**

I. Introduction

MACHINE learning techniques have been applied with success to several wireless communication problems in recent years. One can cite for example detection [1], [2], channel estimation [3], [4], [5], [6] and even end-to-end system learning [7], [8] (see [9], [10] for exhaustive surveys on this topic). However, most of these techniques fall within the supervised learning paradigm and thus require labeled data to operate. The acquisition of such data may be impractical or complex within existing communication systems.

For instance, location-based beamforming (LBB) [11] is a method allowing to choose precoders based on users' locations, thus suppressing the need for base stations to perform channel estimation or beam sweeping. LBB has recently been tackled successfully with supervised machine learning techniques [12]. However, the main drawback of LBB is that it requires having access to the precise users locations, which are difficult to acquire within current systems.

On the other hand, channel charting (CC) [13] recently appeared as an *unsupervised* method allowing to locate users relative to each other, based solely on channel knowledge. Predicting this way the relative locations of users from channel measurements has many potential applications, ranging from SNR prediction [14] and pilot reuse [15], to user grouping, proactive handover management or beam-finding (see [13] for more details on potential applications). CC is an active area of research [16] [17] [18] [19] and rapid progress is expected in this direction. The channel chart can actually be seen as a pseudo-location of the users. In this regard, channel charting could be used to relax the constraints of LBB, by replacing the needed location (difficult to estimate) by a pseudo-location obtained by channel charting.

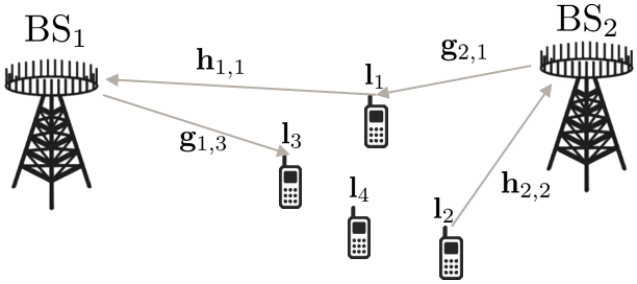


Fig. 1. Considered setting with $B = 2$ and $U = 4$.

A. Contributions

In this paper, channel charting is taken as input for location based beamforming. This allows a base station to choose appropriate precoders based on chart locations instead of spatial locations. This alleviate the need for a precise estimation of user locations and opens the way to several applications such as channel mapping in space and frequency [20]. The proposed method is empirically assessed on realistic synthetic channels, clearly showing the potential of the approach.

B. Related work

To the best of the authors' knowledge, this paper is the first to propose a combination between CC and LBB. The approach relies on previous work done in both areas. The CC method chosen here is inspired by the ones of [16], [19]. The used LBB method is very close to the one of [12], despite the fact that its input is different (chart locations instead of spatial locations). On the application side, one contribution pursues an objective quite similar to this paper [21]. Indeed, it is proposed in [21] to learn a mapping from chart locations to beam indices. However there are major differences between this approach and the present paper. First of all, the beam is taken from a discrete set of candidate beams in [21] (classification problem) whereas a continuous beamforming is sought here (infinite number of beams, regression problem). Second, features of much higher dimension than the channel vectors and requiring temporal averaging (second order moments) are used in [21], whereas an especially designed distance measure that operates directly on instantaneous channel vectors (thus much less complex) is used here. Finally, the neural network used for the mapping is generic (multilayer perceptron) in [21], whereas a specialized architecture based on random Fourier features is used here.

II. Problem formulation

Let us formulate the problem by describing the studied system and stating clearly the pursued objective.

A. Setting

A cell-less multi-user massive MIMO network is considered, in which B base stations (BS) each comprising A antennas communicate over S subcarriers with U user equipments (UE) each having a single antenna. The location of the jth user equipment is denoted $\mathbf{l}_j$, and the vectorized uplink (respectively downlink) channel between it and the ith base station is denoted $\mathbf{h}_{i,j} \in \mathbb{C}^{AS}$ (respectively $\mathbf{g}_{i,j} \in \mathbb{C}^{AS}$). Note that the uplink and downlink channel are not necessarily over the same band. The considered setting is shown on Fig.1.

B. Objective

The goal of the proposed contribution is to map channels in space or frequency, which amounts to determine a downlink precoder $\mathbf{w}_{k,j}$ that should be highly correlated with the downlink (target) channel $\mathbf{g}_{k,j}$, based on the knowledge of the uplink (origin) channel $\mathbf{h}_{i,j}$. The origin and target channels are not necessarily on the same band nor even on the same base station (k and i may be different).

Performance measure. In order to evaluate precoders, the normalized correlation between the precoder $\mathbf{w}_{k,j}$ and the target channel $\mathbf{g}_{k,j}$ is used. It is expressed as

$$\eta_{k,j} \triangleq \frac{|\mathbf{w}_{k,j}^H \mathbf{g}_{k,j}|^2}{\|\mathbf{g}_{k,j}\|_2^2}. \tag{1}$$

It is between zero and one (for a perfect precoder), and is tightly linked to the downlink channel capacity (considering a single user), whose expression is

$$\log(1 + \eta_{k,j}.\text{SNR}_{\text{opt}}),$$

for received signal of the form $y = \sqrt{P}\mathbf{w}_{k,j}^H \mathbf{g}_{k,j} s + n$ where $n \sim \mathcal{N}(0, \sigma^2)$ is additive noise, s is the sent symbol and P the transmit power. In that setting, $\text{SNR}_{\text{opt}} \triangleq \frac{P\|\mathbf{g}_{k,j}\|_2^2}{\sigma^2}$ is the highest achievable signal to noise ratio (SNR) [22]. In summary, the correlation is a single number between zero and one allowing to determine the maximum achievable downlink spectral efficiency for any transmit power and noise variance considering a given precoder.

III. Proposed method

In this section, the inference and training phases of the proposed channel mapping method are described in details.

A. Inference

The proposed method comprises three steps:

1) At first, D latent variables corresponding to the vector $\mathbf{z}_{i,j} \in \mathbb{R}^D$ are computed from the origin channel $\mathbf{h}_{i,j}$ using channel charting methods [13], [16] (with $D \ll SA$) at the ith BS. In particular, the efficient channel charting method of [16] is used in this paper.

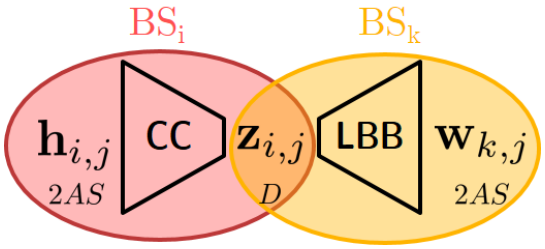


Fig. 2. Schematic view of the proposed method (the real dimension being shown below each variable).

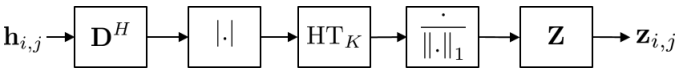


Fig. 3. Proposed similarity-based neural network for the CC step.

2) Then, the latent variables $\mathbf{z}_{i,j}$, that can be seen as a compressed version of the channel are sent from the ith BS to the kth BS.
3) Finally, the kth BS performs location based beamforming using the latent variables as inputs, so as to output a precoder $\mathbf{w}_{k,j}$.

These three steps are depicted on Fig. 2. It is important to notice that the dimension is greatly reduced when performing channel charting, which limits the overhead induced by transmitting the latent variables from the ith to the kth BS.

B. Training

In order to calibrate the CC and LBB steps, a training phase is required.

The CC step is calibrated using the method proposed in [16]. A phase-insensitive distance between training channels

$$\{\mathbf{h}_{i,n}\}_{n=1}^N$$

is first computed and then used within the Isomap nonlinear dimensionality reduction method to obtain chart locations

$$\{\mathbf{z}_{i,n}\}_{n=1}^N.$$

The resulting matrices $\mathbf{D} \triangleq (\mathbf{h}_{i,1}, \ldots, \mathbf{h}_{i,N}) \in \mathbb{C}^{AS \times N}$ and $\mathbf{Z} \triangleq (\mathbf{z}_{i,1}, \ldots, \mathbf{z}_{i,N}) \in \mathbb{R}^{D \times N}$ are then used to form the feedforward neural network shown on Fig. 3 used at inference time to rapidly compute chart location (as first proposed in [19]).

The LBB step is calibrated using the method proposed in [12], except that the inputs are here chart locations instead of true spatial locations. To do so, a database containing chart locations and associated target channels has to be built (classical downlink channel estimation methods have to be used here):

$$\{\mathbf{z}_{i,n}, \mathbf{g}_{k,n}\}_{n=1}^N.$$

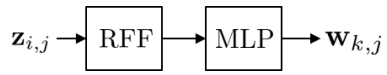


Fig. 4. Proposed RFF-based neural network for the LBB step.

Then, a neural network comprising a random Fourier features (RFF) layer (see detailed architecture in [12]) is trained with a cost function expressed as

$$\mathsf{CF}_k \triangleq 1 - \frac{1}{N}\sum_{n=1}^{N} \frac{|\mathbf{w}_{k,n}^H \mathbf{g}_{k,n}|^2}{\|\mathbf{g}_{k,n}\|_2^2}. \tag{2}$$

This cost measures the misalignment of the predicted precoders with respect to the training target channels and is between 0 (if the precoders are perfectly aligned with the target channels) and 1 (if the precoders are orthogonal to the target channels).

C. Potential applications

The proposed method can be applied to several tasks, depending on the source and target channels. For example, if CC and LBB are done in the same base station ($i = k$) but the frequency bands are different for the uplink (source) and downlink (target) channels, then channel mapping in frequency is carried out. In the more general case where CC and LBB are done in two distinct base stations ($i \neq k$), the proposed method allows to determine a precoder at one BS relying on a channel estimated at another BS (potentially in another frequency band), so that no channel estimation is required at the precoding BS. This is particularly interesting if the CC done at one BS is used to perform LBB at several other BSs (say $B - 1$), since in that case channel estimation would be needed at only one BS to determine precoders at B BSs. The overhead due to channel estimation would then be reduced by a factor of B. Moreover, the backhaul requirements are kept reasonable, since only D real numbers (dimension of the channel chart which is much smaller than the dimension of the channel) have to be shared for each user.

IV. Experiments

To evaluate the proposed approach, a set of experiments is conducted on synthetic data and the results are analysed in this section.

A. Simulation settings

Multipath channel vectors are generated using the DeepMIMO dataset [23]. It consists of channels obtained using a ray-tracing software in an urban outdoor scenario ('O1') (c.f. [23, Figure 2]). Two base stations (i.e. $B = 2$) on opposite sides of a street communicate with $U = 10000$ UEs spread across the map. The first BS corresponds to BS 16 in the original scenario and the second one corresponds to BS 15. Both BSs are equipped with $A = 8 \times 8$ UPAs. The first one operates at 3.5 GHz while the second one operates at 28 GHz and both transmit over sixteen subcarriers (i.e. $S = 16$) spanning a bandwidth of 20 MHz. Note that for simplicity, the precoding results reported in this paper are for a single subcarrier (the central one). All in all, the obtained dataset allows to have access to a triplet $\{\mathbf{h}_{1,j}, \mathbf{g}_{2,j}, \mathbf{l}_j\}$ for each UE (uplink channel to the first BS, downlink channel from the second BS, user location).

B. Implementation details

The PyTorch library [24] is used to implement and conduct the experiments, and implicitly handles the intricacies of neural network training (gradient computation, backpropagation, etc.). Complex vectors are converted to their real representation by stacking the real and imaginary parts resulting in vectors of twice the original size. The dataset is split into a training set and a test set. The training set size is set at $N = \%70 \times U = 7000$.

CC. The channel charting step could be performed in one of two manners: in a *one-shot* (to form a baseline) or *on the fly* (as proposed). The one-shot variant is obtained by applying Isomap to the distance matrix of all of the U channels. Only then is the chart split into the training and test sets. On the other hand, the on the fly variant is obtained by first computing matrices $\mathbf{D}$ and $\mathbf{Z}$, restricted to the training set of size N, that serve to initialize the NN. The test set is then obtained by computing the chart for the rest of the dataset using the latter. The neighborhood for Isomap is of size 5 for both variants.

The on-the-fly setting has the obvious advantage of being independent from training once performed, but is generally less robust than its one-shot counterpart.

LBB. The input is of size $D = 5$ (i.e. dimension of the learned chart), unless otherwise indicated. The number of RFF is set to 600. The MLP consists of 4 layers. Each layer but the last one contains 300 neurons followed by a ReLU nonlinearity. The model is trained for 30 epochs using the Adam optimizer [25] with a batch size of 100. Finally, the model outputs a vector $\mathbf{w} \in \mathbb{R}^{2A}$. For simplicity and to limit the complexity of the model in these initial experiments, the task is restricted to the prediction of the precoder corresponding to the central subcarrier only. Extension to the prediction of the full precoder is trivial but likely requires increasing the number of parameters of the NN.

C. Compared variants

Five variants of the model are considered for comparison.

- LBB is performed at the first BS using the real UEs' locations. Performance is evaluated using the uplink channels. This serves as a **baseline**.
- CC (one-shot) and LBB are both performed at the 1st BS and chained together as described in III. The uplink channels are fed as input for training and used to evaluate the performance.
- CC (one-shot) is performed first at the 1st BS on the uplink channels. The learned chart is then transmitted to the 2nd BS for the LBB step on the downlink channels. This variant corresponds to the channel mapping setting.

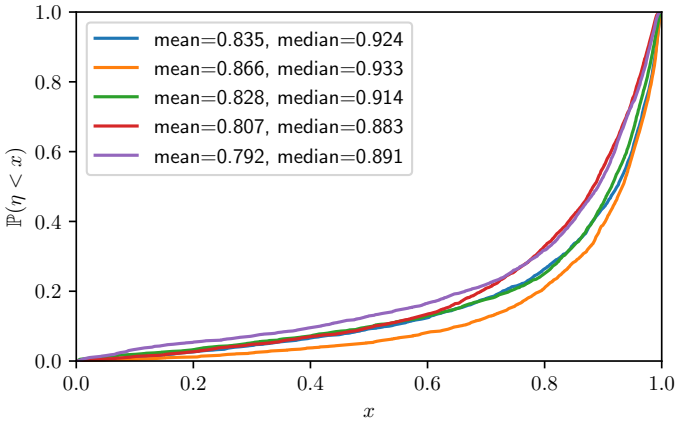


Fig. 5. CDF of the correlations. Blue curve corresponds to the original LBB at BS1. Orange curve corresponds to CC (one-shot) and LBB both at BS1 at the same frequency. Green curve corresponds to CC (one-shot) at BS1 and LBB at BS2 at different frequencies. Red curve corresponds to CC and LBB both at BS1 at the same frequency with a chart of dimensionality 3. Purple curve corresponds to CC (on the fly) at BS1 and LBB at BS2 at different frequencies.

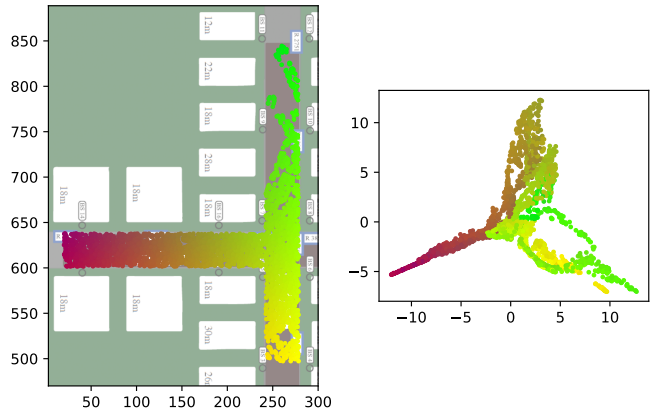


Fig. 6. Spatial distribution of the UEs in the test set (**left**) and their corresponding learned chart obtained on the fly (**right**). Color is added to distinguish local neighborhoods.

- This variant is similar to the second one, except this time the learned chart is of dimension $D = 3$. The chart is computed in a one-shot.
- This variant is similar to the third one, except this time CC is computed on the fly. This last variant corresponds to the most realistic setting that is described in section III-B.

D. Results

The five variants of the model are evaluated on the test set. Fig. 5 shows the cumulative distribution function (CDF) of the correlations η—the closer to a Dirac at $x = 1$ the better. All methods perform relatively well and achieve a satisfactory level of performance. Interestingly, the CC based LBB variant performs better than the classical one, meaning that the 5 dimensions of the learned chart allow to convey more information than the 3-dimensional spatial locations. This hypothesis is consolidated when comparing classical LBB to the CC based one when keeping only 3 dimensions to the chart where we observe a small decline in performance. Furthermore, the figure shows that in the channel mapping setting, the model performs slightly worse but is on par with the classical one. Finally, the CC based LBB model where the chart is computed on the fly performs the worst, which is expected and highlights the influence of the quality of the learned chart on the subsequent tasks. It worth noting that in this last case, an end-to-end training of the network (i.e., training both the CC NN and the LBB NN) would certainly improve the performance.

Fig. 7 shows the spatial distribution of the correlations for the last approach described in IV-C. The model succeeds in learning an adequate precoder for most of the spatial locations for both LoS and NLoS channels. Some areas show a high concentration in blue dots (top left and bottom left of the map) corresponding to precoders of low quality and could

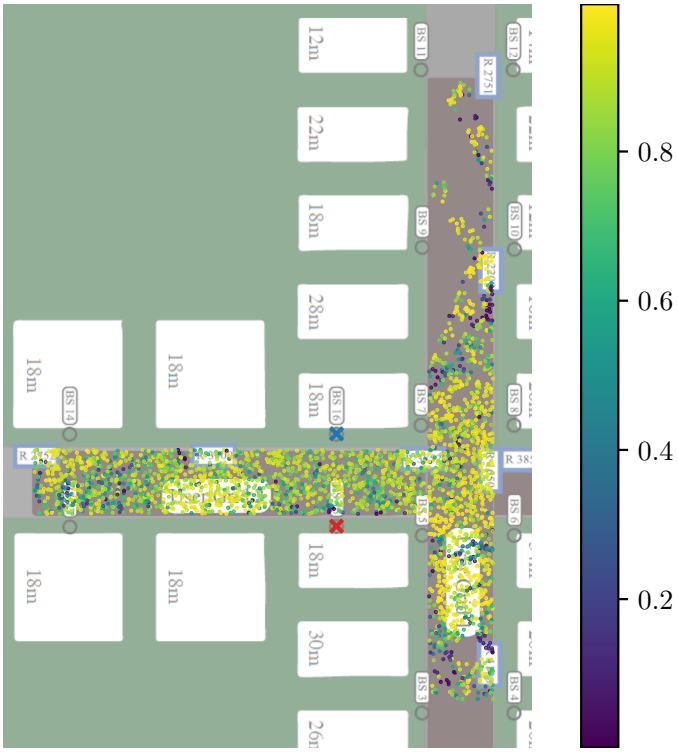


Fig. 7. Spatial distribution of the correlations. The blue and red crosses correspond to the 1st and 2nd BS respectively.

be explained by locations where propagation conditions with respect to the two base stations are very different (very weak channel between the user and one of the BSs).

V. Conclusion and Outlook

In this paper, a way to directly predict precoders from channel chart locations was proposed. It amounts to combining wireless channel charting and location-based beamforming.

It has the potential of greatly reducing the overhead due to channel estimation in cell-less systems, since CC can be carried out in a base station, and LBB in another one (or several others) that does not need to perform channel estimation nor beam sweeping. The amount of data to transfer between the two BS is reduced since the channel chart is of very low dimension. The proposed method was empirically assessed on realistic synthetic urban channels, showing the great potential of the approach, since it was found that the chart information (in dimension five) was as good as the real location information.

In future work, it would be interesting to train the whole system end-to-end (CC and LBB), which raises the question of how to handle the transmission of gradients and parameters between base stations (decentralized federated learning). Moreover, a theoretical motivation based on the channel manifold in order to set hyperparameters in a principled way would be valuable. Finally, a similar approach could be envisioned for beam prediction within a finite set (as studied in [21]), since the problem can then be seen as a discrete version of the problem studied here.

ACKNOWLEDGEMENT

This work has been partly funded by the European Commission through the H2020 project Hexa-X (Grant Agreement no. 101015).

REFERENCES

[1] Neev Samuel, Tzvi Diskin, and Ami Wiesel. Learning to detect. *IEEE Transactions on Signal Processing*, 67(10):2554–2564, 2019.

[2] Hengtao He, Chao-Kai Wen, Shi Jin, and Geoffrey Ye Li. Model-driven deep learning for mimo detection. *IEEE Transactions on Signal Processing*, 68:1702–1715, 2020.

[3] Hengtao He, Chao-Kai Wen, Shi Jin, and Geoffrey Ye Li. Deep learning-based channel estimation for beamspace mmwave massive mimo systems. *IEEE Wireless Commun. Lett.*, 7(5):852–855, 2018.

[4] Eren Balevi, Akash S. Doshi, and Jeffrey G. Andrews. Massive MIMO channel estimation with an untrained deep neural network. *IEEE Trans. Wireless Commun.*, 19(3):2079–2090, mar 2020.

[5] Taha Yassine and Luc Le Magoarou. mpNet: variable depth unfolded neural network for massive MIMO channel estimation. *IEEE Trans. Wireless Commun.*, PP:1–1, 2022.

[6] Baptiste Chatelier, Luc Le Magoarou, and Getachew Redieteab. Efficient deep unfolding for siso-ofdm channel estimation. *arXiv preprint arXiv:2210.06588*, 2022.

[7] Timothy O'Shea and Jakob Hoydis. An introduction to deep learning for the physical layer. *IEEE Trans. on Cogn. Commun. Netw.*, 3(4):563–575, 2017.

[8] Fayçal Ait Aoudia and Jakob Hoydis. Model-free training of end-to-end communication systems. *IEEE Journal on Selected Areas in Communications*, 37(11):2503–2516, 2019.

[9] Chaoyun Zhang, Paul Patras, and Hamed Haddadi. Deep learning in mobile and wireless networking: A survey. *IEEE Communications surveys & tutorials*, 21(3):2224–2287, 2019.

[10] Tianqi Wang, Chao-Kai Wen, Hanqing Wang, Feifei Gao, Tao Jiang, and Shi Jin. Deep learning for wireless physical layer: Opportunities and challenges. *China Commun*, 14(11):92–111, 2017.

[11] Petteri Kela, Mario Costa, Jussi Turkka, Mike Koivisto, Janis Werner, Aki Hakkarainen, Mikko Valkama, Riku Jantti, and Kari Leppanen. Location based beamforming in 5G ultra-dense networks. In *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, pages 1–7. IEEE, 2016.

[12] Luc Le Magoarou, Taha Yassine, Stéphane Paquelet, and Matthieu Crussière. Deep learning for location based beamforming with NLOS channels. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8812–8816. IEEE, 2022.

[13] Christoph Studer, Saïd Medjkouh, Emre Gönültaş, Tom Goldstein, and Olav Tirkkonen. Channel charting: Locating users within the radio environment using channel state information. *IEEE Access*, 6:47682–47698, 2018.

[14] Parham Kazemi, Hanan Al-Tous, Christoph Studer, and Olav Tirkkonen. Snr prediction in cellular systems based on channel charting. In *2020 IEEE Eighth International Conference on Communications and Networking (ComNet)*, pages 1–8, 2020.

[15] Lucas Ribeiro, Markus Leinonen, Hamza Djelouat, and Markku Juntti. Channel charting for pilot reuse in mmtc with spatially correlated mimo channels. In *2020 IEEE Globecom Workshops (GC Wkshps*, pages 1–6, 2020.

[16] Luc Le Magoarou. Efficient channel charting via phase-insensitive distance computation. *IEEE Wireless Commun. Lett.*, 10:2634–2638, 2021.

[17] Paul Ferrand, Alexis Decurninge, Luis G. Ordoñez, and Maxime Guillaud. Triplet-based wireless channel charting: Architecture and experiments. *IEEE J. Sel. Areas Commun.*, 39:2361–2373, 2021.

[18] Brian Rappaport, Emre Gönültaş, Jakob Hoydis, Maximilian Arnold, Pavan Koteshwar Srinath, and Christoph Studer. Improving channel charting using a split triplet loss and an inertial regularizer. In *Proc. IEEE 17th Int. Symp. Wireless Commun. Syst. (ISWCS)*, pages 1–6, Berlin, Germany, 2021. IEEE.

[19] Taha Yassine, Luc Le Magoarou, Stéphane Paquelet, and Matthieu Crussière. Leveraging triplet loss and nonlinear dimensionality reduction for on-the-fly channel charting. In *IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2022.

[20] Muhammad Alrabeiah and Ahmed Alkhateeb. Deep learning for tdd and fdd massive mimo: Mapping channels in space and frequency. In *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, pages 1465–1470. IEEE, 2019.

[21] Tushara Ponnada, Parham Kazemi, Hanan Al-Tous, Ying-Chang Liang, and Olav Tirkkonen. Best beam prediction in non-standalone mm wave systems. In *2021 Joint European Conference on Networks and Communications; 6G Summit (EuCNC/6G Summit)*. IEEE, jun 2021.

[22] Emil Björnson, Jakob Hoydis, Luca Sanguinetti, et al. Massive mimo networks: Spectral, energy, and hardware efficiency. *Foundations and Trends® in Signal Processing*, 11(3-4):154–655, 2017.

[23] A. Alkhateeb. DeepMIMO: A generic deep learning dataset for millimeter wave and massive MIMO applications. In *Proc. of Information Theory and Applications Workshop (ITA)*, pages 1–8, San Diego, CA, Feb 2019.

[24] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[25] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.