

Channel Charting in Real-World Coordinates

Sueda Taner, Victoria Palhares, and Christoph Studer

*Department of Information Technology and Electrical Engineering, ETH Zurich, Switzerland
email: taners@iis.ee.ethz.ch, vmenesca@ethz.ch, and studer@ethz.ch*

Abstract—Channel charting is an emerging self-supervised method that maps channel state information (CSI) to a low-dimensional latent space, which represents pseudo-positions of user equipments (UEs). While this latent space preserves local geometry, i.e., nearby UEs are nearby in latent space, the pseudo-positions are in arbitrary coordinates and global geometry is not preserved. In order to enable channel charting in real-world coordinates, we propose a novel bilateration loss for multipoint wireless systems in which only the access point (AP) locations are known—no geometrical models or ground-truth UE position information is required. The idea behind this bilateration loss is to compare the received power at pairs of APs in order to determine whether a UE should be placed closer to one AP or the other in latent space. We demonstrate the efficacy of our method using channel vectors from a commercial ray-tracer.

I. INTRODUCTION

Channel charting is a self-supervised method that extracts user equipment (UE) pseudo-position solely by processing estimated channel state information (CSI) at infrastructure basestations (BSs) or access points (APs) [2]. The key idea is to apply dimensionality reduction to a large database of CSI features that represent large-scale properties of the wireless channel; this leads to a low-dimensional latent space—the so-called channel chart—that is tied to UE position. This UE pseudo-position information can be used to assist a wide range of applications in wireless communication systems, such as pilot allocation [3], beam management [4], [5], channel capacity prediction [6], and many more [7].

Although channel charts preserve the local geometry of UE positions, i.e., nearby UEs correspond to points that are nearby in the channel chart, global geometry is typically distorted and the learned channel charts are represented in arbitrary coordinates—all of this is a consequence of self-supervised learning. Multipoint channel charting techniques, which process CSI acquired at multiple distributed BSs or APs, have been shown to improve global geometry [8]–[10]. Nonetheless, the resulting channel charts remain to be pseudo-positions with no direct ties to real-world coordinates.

The work of ST, VP, and CS was supported in part by the Swiss National Science Foundation (SNSF) grant 207314 and an ETH Research grant. The work of CS was also supported in part by the US National Science Foundation (NSF) under grants CNS-1717559 and ECCS-1824379.

The authors would like to thank Paul Ferrand, Maxime Guillaud, and Olav Tirkkonen for discussions on channel charting and Gian Marti for comments on the final version of this paper. The authors also thank Remcom for providing a license for Wireless InSite [1].

A channel chart in real-world coordinates would improve spatial awareness, e.g., for handover purposes, and allow for more effective deployment planning. Therefore, methods that associate channel charts with real-world coordinates have been proposed recently. One strain of methods first employs self-supervised multipoint channel charting and then, maps the resulting pseudo-positions to real-world coordinates [11], [12]. The method in [11] relies on the assumption that the received power is the highest for UE positions nearest to the AP, which may not hold true as we will demonstrate in Sec. IV-B; the method in [12] relies on accurately synchronized APs to incorporate time-of-flight-based ranging. Another strain of methods deploys semi-supervised learning to anchor the channel chart in real-world coordinates [13]–[18]; all of these methods require ground-truth UE position information during training.

A. Contributions

In contrast to prior work, we propose a novel method that learns neural network-based channel charting functions that generate channel charts in real-world coordinate systems. The proposed method is weakly-supervised as it only exploits knowledge of the AP locations in a multipoint scenario—no geometric models, accurate AP synchronization, or labeled CSI samples with UE ground-truth positions are required. More specifically, we postulate a novel bilateration loss, which compares the power of the received CSI between a transmitting UE and a pair of APs with the goal of placing the UE closer to the AP that is receiving higher power. While this bilateration loss alone already produces utilizable channel charts in real-world coordinates, their quality can be improved substantially when combined with the timestamp-based triplet loss from [19]. We demonstrate the efficacy of our method using channel vectors from a commercial ray-tracer [1].

B. Notation

Column vectors and matrices are denoted by lowercase and uppercase boldface letters, respectively; sets are denoted by uppercase calligraphic letters. For a vector \mathbf{a} , the Euclidean norm is $\|\mathbf{a}\|$ and the entry-wise absolute value is $|\mathbf{a}|$. For a matrix \mathbf{A} , the Frobenius norm is $\|\mathbf{A}\|_F$ and the column-wise vectorization is $\text{vec}(\mathbf{A})$. The cardinality of a set \mathcal{A} is $|\mathcal{A}|$. The operator $(x)^+ = \max\{x, 0\}$ is the rectified linear unit (ReLU).

II. CHANNEL CHARTING BASICS

We now briefly outline the basics of channel charting, detail the system model, and discuss channel charting with neural

networks using the triplet-based learning approach, which is part of the method we propose in Sec. III.

A. Operating Principle

Channel charting typically operates in two phases [2]. In the first phase, CSI from a large number of different UE positions is acquired and CSI features that capture large-scale fading properties of the channel are stored in a database. By applying parametric dimensionality reduction [20] to the CSI feature database, one then learns a channel charting function (e.g., using a neural network), which maps CSI features to a low-dimensional representation: the channel chart. Channel charts have the useful property that the UEs transmitting from nearby positions are also placed nearby in the channel chart. We reiterate that channel chart learning is self-supervised, meaning that no ground-truth information about the UEs' position is required. In the second phase, the channel charting function is used to map new CSI features to points in the channel chart, which represent the transmitting UEs' pseudo-positions.

B. System Model

We consider a single-input multiple-output (SIMO) communication system in which one or multiple single-antenna UEs transmit pilots to A distributed APs with M_R antennas each, leading to $B = AM_R$ receive antennas in total. We consider orthogonal frequency-division multiplexing (OFDM) transmission with W occupied subcarriers. We assume that the AP $a \in \mathcal{A} = \{1, \dots, A\}$ is at position $\mathbf{x}^{(a)} \in \mathbb{R}^3$ in physical space, and that these positions are known.¹

Suppose that we have N pilot transmissions from one UE at (unknown) positions $\mathbf{x}^{(n)} \in \mathbb{R}^3$ at timestamps t_n for $n \in \mathcal{N} = \{1, \dots, N\}$. The n th transmission from UE position $\mathbf{x}^{(n)}$ enables the AP a to estimate the associated CSI vector $\mathbf{h}_w^{(n,a)} \in \mathbb{C}^{M_R}$ at timestamp t_n and subcarrier w . By stacking the channel vectors $\{\mathbf{h}_w^{(n,a)}\}_{a=1}^A$ from all APs, we can construct a CSI vector $\mathbf{h}_w^{(n)} \in \mathbb{C}^B$ that contains CSI for all B receive antennas. It is important to note that we do not require the A APs to be perfectly synchronized while acquiring CSI; the only requirement is that the CSI estimated at each AP belongs to the same UE transmitting from approximately the same position $\mathbf{x}^{(n)}$ at approximately the same timestamp t_n . Finally, we concatenate the channel vectors from all subcarriers to construct a $B \times W$ CSI matrix associated with the UE at timestamp t_n as $\mathbf{H}^{(n)} = [\mathbf{h}_1^{(n)}, \dots, \mathbf{h}_W^{(n)}]$. The entire CSI database is given by the set of matrices $\{\mathbf{H}^{(n)}\}_{n \in \mathcal{N}}$.

C. CSI Feature Extraction

In order to extract large-scale fading properties of the wireless channel [2] and to render channel charting resilient to system and hardware impairments (e.g., phase offsets between APs) [19], [21], one transforms the CSI matrices into CSI features. We follow the CSI feature design from [14], [22]. First, we transform frequency-domain CSI into the delay domain by applying an inverse discrete Fourier transform over the W

occupied subcarriers. Since most of the received power is typically concentrated in the first few taps, we truncate the delay-domain CSI by taking the first C columns. In order to render our CSI features resilient to changes in the UEs' transmit power, we normalize the truncated delay-domain CSI matrix to have unit Frobenius norm and denote the resulting matrix by $\hat{\mathbf{H}}^{(n)} \in \mathbb{C}^{B \times C}$. Finally, we compute the $D' = BC$ dimensional vector $\mathbf{f}^{(n)} = |\text{vec}(\hat{\mathbf{H}}^{(n)})|$, which ignores phase shifts (e.g., from APs that are not synchronized in phase). The entire CSI feature database is given by the set of vectors $\{\mathbf{f}^{(n)}\}_{n \in \mathcal{N}}$.

D. Channel Charting with Neural Networks

With the CSI feature database, one can now learn a channel charting function $g_\theta : \mathbb{R}^{D'} \rightarrow \mathbb{R}^D$, which maps a CSI feature vector $\mathbf{f}^{(n)}$ to a D -dimensional pseudo-position as follows: $\hat{\mathbf{x}}^{(n)} = g_\theta(\mathbf{f}^{(n)})$. The function g_θ is implemented as a neural network that is parametrized by the vector θ which includes all of the network's weights and biases. While real-world positions are in three-dimensional space with coordinates (x, y, z) , the channel chart can be embedded in fewer dimensions if, for example, the z -coordinate is fixed across the AP and UE positions. Hence, assuming that $D \leq 3$, we denote the truncated vector consisting of the first D entries of $\mathbf{x}^{(n)}$ and $\mathbf{x}^{(a)}$ as $\tilde{\mathbf{x}}^{(n)} \in \mathbb{R}^D$ and $\tilde{\mathbf{x}}^{(a)} \in \mathbb{R}^D$, respectively.

The literature describes a variety of methods to learn neural network-based channel charting functions, such as autoencoders [2], Siamese neural networks [14], and neural networks trained with a timestamp-based triplet loss [19], [23], [24]. In what follows, we focus on the triplet-loss channel charting approach from [19].

E. Triplet-Based Channel Charting

Assuming that the timestamps associated with all CSI features are available, reference [19] proposes to use this side information when comparing pairwise distances in latent space. To this end, one defines a set of triplets from the set of sample indices \mathcal{N} as follows:

$$\mathcal{T} = \{(n, c, f) \in \mathcal{N}^3 : 0 < |t_n - t_c| \leq T_c < |t_n - t_f|\}, \quad (1)$$

where $T_c > 0$ is a parameter that categorizes the CSI features as close or far in time. If t_n is closer to t_c than t_f , then we would expect the Euclidean distance between the UE positions at timestamp t_n and t_c to be smaller than that of t_f . This property can be expressed with the following triplet loss [19]:

$$\mathcal{L}_t = \frac{1}{|\mathcal{T}|} \sum_{(n,c,f) \in \mathcal{T}} (\|g_\theta(\mathbf{f}^{(n)}) - g_\theta(\mathbf{f}^{(c)})\| - \|g_\theta(\mathbf{f}^{(n)}) - g_\theta(\mathbf{f}^{(f)})\| + M_t)^+. \quad (2)$$

Here, the margin parameter $M_t \geq 0$ enforces $g_\theta(\mathbf{f}^{(n)})$ to be at least M_t closer to $g_\theta(\mathbf{f}^{(c)})$ than to $g_\theta(\mathbf{f}^{(f)})$.

As demonstrated in [19], one can train a neural network that implements the channel charting function g_θ by minimizing the loss \mathcal{L}_t ; this approach is self-supervised as the loss only utilizes the training dataset without any additional ground-truth labels, such as UE positions. We reiterate that the coordinate system of the resulting channel chart is arbitrary, e.g., scaled,

¹It is a reasonable assumption to know the AP positions, as the APs are typically placed according to a carefully-crafted deployment plan.

rotated, and globally warped. The method we propose next addresses exactly this limitation.

III. CHANNEL CHARTING IN REAL-WORLD COORDINATES

We now propose a novel loss that enables one to directly learn channel charts in real-world coordinates. Our method only uses the known AP positions in a multipoint scenario—*no* geometric models, accurate time synchronization among APs, or CSI samples labeled with UE positions are required.

A. Bilateral Loss

Assuming that a UE's pilot signal is received at multiple APs, there is one clue about the position of the UE that our CSI features contains: the relative receive power at each AP. As one would expect, an AP with higher relative power tends to be closer to the UE under line-of-sight conditions. Let $\hat{\mathbf{H}}^{(n,a)} \in \mathbb{C}^{M_R \times C}$ denote the submatrix of $\hat{\mathbf{H}}^{(n)}$ which consists of the M_R rows corresponding to AP a . We now compute the relative power of the channel between the UE and the a th AP at timestamp t_n as $P^{(n,a)} = 20 \log_{10}(\|\hat{\mathbf{H}}^{(n,a)}\|_F)$, $a \in \mathcal{A}$, and define a set of AP pairs

$$\mathcal{P}^{(n)} = \{(a_c, a_f) \in \mathcal{A}^2 : P^{(n,a_c)} > P^{(n,a_f)} + M_p\}. \quad (3)$$

The set $\mathcal{P}^{(n)}$ generates pairs of AP indices (a_c, a_f) , where the two selected APs act as close and far reference points for the n th UE position. Here, the margin parameter $M_p \geq 0$ enforces the set $\mathcal{P}^{(n)}$ to only include AP pairs whose channel powers differ by at least M_p . The assumption that the UE n should be closer to AP a_c than AP a_f as determined by the set $\mathcal{P}^{(n)}$ in (3) leads to the following *bilateration loss*:

$$\mathcal{L}_b = \frac{1}{|\mathcal{N}||\mathcal{P}^{(n)}|} \sum_{n \in \mathcal{N}} \sum_{(a_c, a_f) \in \mathcal{P}^{(n)}} (\|g_\theta(\mathbf{f}^{(n)}) - \tilde{\mathbf{x}}^{(a_c)}\| - \|g_\theta(\mathbf{f}^{(n)}) - \tilde{\mathbf{x}}^{(a_f)}\| + M_b)^+. \quad (4)$$

Here, the margin $M_b \geq 0$ enforces $g_\theta(\mathbf{f}^{(n)})$ to be at least M_b closer to AP a_c than AP a_f . We emphasize that the loss \mathcal{L}_b requires no assumption of how far each AP should be from the UE based on their channel power; we merely deduce relative distances to the two APs. Moreover, while the underlying assumption on the power-distance relation between UE and APs may not always hold true, we seek no such guarantee—we support this claim with a concrete example in Sec. IV-B. One can train a neural network that implements the channel charting function g_θ by minimizing the loss \mathcal{L}_b in (4); this approach is weakly-supervised as the loss incorporates only partial information on the ground-truth labels (i.e., AP positions).

We note that a loss based on received power was also proposed in [16]; this loss, however, compares the power of two CSI vectors corresponding to the channel between one AP and the UE at two timestamps t_n and t_f in order to deduce whether $\mathbf{x}^{(n)}$ or $\mathbf{x}^{(f)}$ is closer to the AP. This loss requires the UEs to have fixed transmit power over time and does not leverage the fact that the UEs are simultaneously transmitting to multiple APs. In contrast, our bilateration loss in (4) takes

TABLE I
SUMMARY OF SIMULATION PARAMETERS

Parameter	Value or Type
Number of APs	$A = 8$
Number of antennas per AP	$M_R = 4$
Number of UE positions	15 912
Spacing between UE positions	0.8 m
AP antenna height	10 m
AP antenna spacing	Half-wavelength
UE antenna height	1.5 m
AP and UE antenna type	Omnidirectional
Carrier frequency	1.9 GHz
Bandwidth	20 MHz
Number of used subcarriers	$W = 1200$
Received SNR per AP	−9.7 dB to 33.8 dB

each CSI feature into account individually and compares the channel power between a pair of APs; this means that our loss does not rely on timestamps and does not require the transmit power to stay constant between samples. Thus, our bilateration loss could also be used in scenarios in which timestamps are irrelevant, e.g., a multiuser scenario with stationary UEs.

B. Combined Triplet and Bilateral Loss

In order to improve channel charting, we propose to combine the two loss functions \mathcal{L}_t in (2) and \mathcal{L}_b in (4), since \mathcal{L}_t helps in preserving local neighborhood relations whereas \mathcal{L}_b helps to anchor the channel chart in real-world coordinates; moreover, false triplets (i.e., triplets for which the anchor point is closer in space to the point whose timestamp is further apart) in \mathcal{L}_t and false AP pairs (see Sec. IV-B for a detailed definition) in \mathcal{L}_b can be counterbalanced. We define the combined triplet and bilateration loss as follows:

$$\mathcal{L}_{t,\lambda b} = \mathcal{L}_t + \lambda \mathcal{L}_b. \quad (5)$$

Here, $\lambda > 0$ is a parameter that balances the two losses; the choice of this parameter is briefly discussed in Sec. IV-D.

One can train a neural network that implements the channel charting function g_θ by minimizing the loss $\mathcal{L}_{t,\lambda b}$; this approach is a hybrid between self- and weakly-supervised learning due to the reasons mentioned in Sec. II-E and Sec. III-A, respectively.

IV. RESULTS

We now present results for the proposed channel charting methods and comparisons to various baselines.

A. Simulation Scenario

We consider an urban scenario in which multiple APs are placed around a rectangular area of size 83 m \times 122 m; see Fig. 2 (a) for an illustration of the UE and AP positions. We use sub-6-GHz scenario with channel vectors from Remcom's Wireless InSite ray-tracing software [1]. Tbl. I summarizes the simulation parameters. Each CSI matrix $\mathbf{H}^{(n)}$, $n \in \mathcal{N}$ is of dimension 32×1200 ; each CSI feature vector is of dimension $D' = 32 \times 8 = 256$, where we use the $C = 8$ first taps and the feature extraction pipeline from Sec. II-C. Since the heights of all APs are the same and the height of all UEs is also fixed, we perform channel charting in $D = 2$ dimensions.

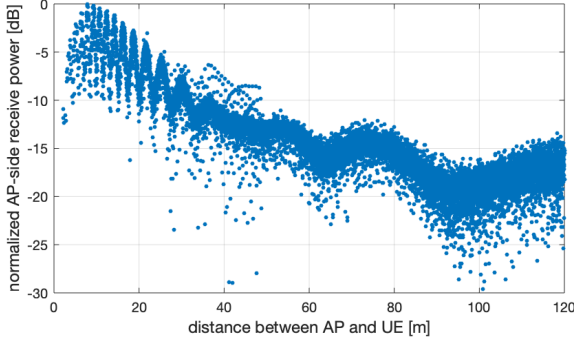


Fig. 1. Receive power (in decibels) at one AP dependent on the AP-to-UE distance (in meters) in the xy-plane. The receive power values are normalized to the range $(-\infty, 0]$ dB.

B. Is Received Power Inversely Proportional to Distance?

Before we show channel charting results, we first provide an example of the power-distance relation that is exploited by our bilateration loss in (4). Fig. 1 shows the (normalized) receive power at one AP depending on the distance to an UE in the xy-plane; we ignore the distance in the z-axis since the heights are fixed. Evidently, the power generally decreases with distance, but small-scale fading causes the power to fluctuate quite substantially among the different UE positions which are at approximately the same distance to the AP. Furthermore, the received power increases² with distance until about 10 m, which negatively affects approaches that rely on the power being the highest near the AP, e.g., the method in [11].

Since a smaller difference in power is more likely to be caused by small-scale fading (and not by distance), we would be less confident in deducing which AP might be closer. Hence, observing that the AP-side receive power is not perfectly inversely proportional to distance motivates the margin M_p in (3). By setting $M_p > 0$, we can avoid some false AP pairs, i.e., pairs $(a_c, a_f) \in \mathcal{P}^{(n)}$ for which $\|\tilde{\mathbf{x}}^{(n)} - \tilde{\mathbf{x}}^{(a_c)}\| > \|\tilde{\mathbf{x}}^{(n)} - \tilde{\mathbf{x}}^{(a_f)}\|$. For our CSI dataset, margins M_p of 0, 3, and 6 result in false AP pair ratios of 19.5%, 9.9%, and 4.9% for all UEs, while the average number of AP pairs in the set $\mathcal{A}^{(n)}$ given by $\frac{1}{N} \sum_{n=1}^N |\mathcal{A}^{(n)}|$ is 28, 17.9, and 10.1, respectively. Clearly, there exists a trade-off in choosing M_p : Increasing M_p leads to fewer false AP pairs at the cost of fewer AP pairs, which yields more accurate but fewer reference points for weakly-supervised learning (and vice versa).

C. Learning the Channel Charting Function

Following the neural network architecture in [19], we use a six-layer fully-connected neural network for the channel charting function g_θ . The neural network has $\{256, 128, 64, 32, 16, 2\}$ activations per layer; all layers except for the last one use ReLU activations, whereas the last layer uses linear activations. We initialize the weights with Glorot [25] and use the Adam optimizer to learn the channel charting function g_θ .

²This behavior can be attributed to the antenna radiation pattern, which causes the receive power to reduce if a UE approaches the area below an AP.

The CSI dataset described in Sec. IV-A is created for UE positions on a 104×153 rectangular grid with a total of 15 912 CSI samples. We split this CSI sample set randomly into training and test sets of 12 000 and 3912 samples, respectively. For training with the triplet loss, we timestamp the CSI dataset by imitating the following trajectory in the rectangular area: The UE first moves on the outermost (largest) rectangle in the area on the grid, then takes one step inwards, walks on the rectangle whose edges are now 1.6 m shorter than the first rectangle, then takes one step inwards, and so on, until the innermost (smallest) rectangle is completed and all points in the dataset are covered.

D. Proposed Methods and Baselines

In order to compare our proposed methods to baselines, we also consider two positioning methods and state-of-the-art channel charting. All of these methods (including ours) use the same neural network architecture (cf. Sec. IV-C) but are trained with different loss functions, which we detail next.

Proposed 1 (P1): For our first proposed method, we extract the relative power of the (normalized) CSI feature vector to each AP as described in Sec. III-A. We train the neural network only using the bilateration loss \mathcal{L}_b defined in (4) with parameters $M_p = 3$ and $M_b = 15$. We reiterate that, due to the reasons mentioned in Sec. III-A, this approach is weakly-supervised. In addition, this method also demonstrates the use of large λ values in (5) as $\mathcal{L}_b = \lim_{\lambda \rightarrow \infty} \lambda^{-1} \mathcal{L}_{t,\lambda b}$.

Proposed 2 (P2): For our second proposed method, we assume that timestamps are available for all CSI features in the training set; we also extract the relative power of the (normalized) CSI feature vector to each AP as described in Sec. III-A. We train the neural network using the combined triplet and bilateration loss $\mathcal{L}_{t,b}$ defined in (5) with parameters $T_c = 20$, $M_p = 3$, and $M_t = M_b = 15$. We reiterate that due to the reasons mentioned in Sec. II-E and Sec. III-A regarding each component of the loss \mathcal{L}_t and \mathcal{L}_b , this approach is a hybrid between self- and weakly-supervised learning.

Baseline 1 (B1): In order to compare our methods to supervised positioning that processes CSI features with neural networks (see, e.g., [21], [26] and the references therein), this baseline assumes that ground truth positions are available for all samples in the training set. Therefore, we train this neural network with the mean squared error (MSE) loss:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} \|g_\theta(\mathbf{f}^{(n)}) - \tilde{\mathbf{x}}^{(n)}\|^2. \quad (6)$$

Since the loss \mathcal{L}_{MSE} requires ground-truth labels (i.e., UE positions) for all training samples, this approach relies on supervised learning. This method serves as a baseline to show the best-possible positioning performance achievable with the neural network architecture described in Sec. IV-C.

Baseline 2 (B2): In order to compare our methods to the state of the art in channel charting, this baseline assumes that the timestamps are available for all CSI features in the training set. We train the neural network using the triplet loss \mathcal{L}_t defined in (2) with parameters $T_c = 20$ and $M_t = 1$. We reiterate that

this approach is self-supervised due to the reasons mentioned in Sec. II-E. Furthermore, this baseline also demonstrates the result for using small values of λ in (5) as $\mathcal{L}_t = \lim_{\lambda \rightarrow 0} \mathcal{L}_{t,\lambda b}$. Note that the resulting channel charts are typically of high quality but embedded in arbitrary coordinates.

Baseline 3 (B3): In order to compare our methods to semi-supervised positioning techniques [13]–[18], this baseline assumes that while the timestamps are available for all samples in the training set, UE ground-truth position information is available for a small subset \mathcal{N}' of the training set, i.e., $|\mathcal{N}'| = 0.05N$. Hence, the MSE loss can be calculated only for a subset of the available samples. Although the MSE loss is the same as in (6), we denote the loss in this case by $\mathcal{L}_{t,\widehat{\text{MSE}}}$ to distinguish it from the loss taken over all training samples in the fully-supervised baseline B1. We train the neural network using the combined triplet and $\widehat{\text{MSE}}$ loss function defined as

$$\mathcal{L}_{t,\widehat{\text{MSE}}} = \mathcal{L}_t + \mathcal{L}_{\widehat{\text{MSE}}} \quad (7)$$

with parameters $T_c = 20$ and $M_t = 15$. Since the loss $\mathcal{L}_{t,\widehat{\text{MSE}}}$ utilizes a subset of the UE ground-truth positions in addition to the triplet loss, this approach is semi-supervised. This baseline positioning method showcases whether our proposed methods can achieve the performance of semi-supervised techniques without the need for UE ground-truth position information.

E. Performance Metrics

We evaluate the performance of the proposed methods and baselines using six different metrics. The first four metrics measure the quality of the latent space; we refer to [27] for the details. The latter two metrics measure positioning accuracy: (i) *Trustworthiness (TW)* penalizes neighborhood relationships in latent space that are not present in the real-world positions [28]; TW values are in $[0, 1]$ with optimal value 1. (ii) *Continuity (CT)* measures how well neighborhood relationships between the real-world positions are preserved in latent space [28]; CT values are in $[0, 1]$ with optimal value 1. (iii) *Kruskal stress (KS)* measures the dissimilarity between pairwise distances in the real-world positions and pairwise distances in latent space [29]; KS values are in $[0, 1]$ with optimal value 0. (iv) *Rajski distance (RD)* measures the difference between the mutual information and joint entropy of the distribution of pairwise distances in the real-world positions and latent space [30]; RD values are in $[0, 1]$ with optimal value 0. (v) *Mean distance error* measures the average error in the UE position estimates in Euclidean norm over all UE positions; this metric is nonnegative with optimal value 0. (vi) *95th percentile distance error* measures the 95th percentile of the error in the UE position estimates in Euclidean norm over all UE positions; this metric is nonnegative with optimal value 0.

F. Results and Comparison

We now compare the proposed methods with the baselines discussed in Sec. IV-D using the metrics from Sec. IV-E; all of the channel charts, positioning plots, and evaluated metrics are based on the test set of 3912 CSI samples mentioned in Sec. IV-C. Fig. 2 visually compares the ground-truth positions

as well as channel charts in arbitrary and real-world coordinates; Tbl. II summarizes the types of supervision and the associated performance metrics. Note that the two distance error metrics are irrelevant for baseline B2 as it produces channel charts in arbitrary coordinates; hence, we do not report these metrics for baseline B2 in Tbl. II.

Figures 2(b-f) show that all of the considered methods preserve local geometry. As expected, the baselines B1 and B3 result in the most visually pleasing results, as they rely on ground-truth UE position information. In contrast, B2 provides a high-quality channel chart but in arbitrary coordinates and where global geometry is distorted. The proposed method P1 provides a channel chart in real-world coordinates, but of slightly worse quality than the triplet-loss-based channel charting baseline B2. The proposed method P2 combines the best of both worlds: the quality of the channel chart from B2 and representation in real-world coordinates from P1.

Tbl. II reveals that the supervised positioning baseline B1 achieves the best performance in all considered metrics. We observe that, while the proposed method P1 has the advantages of (i) generating a channel chart in real-world coordinates and (ii) not requiring *any* ground-truth AP or UE position labels, it is outperformed by all the other methods in terms of all latent space quality metrics; we attribute the lower quality in P1's channel chart to the false AP pairs mentioned in Sec. IV-B. We also observe that the proposed method P2 outperforms B2, B3, and P1 in all latent space quality metrics, while its mean and 95th percentile positioning errors are only slightly higher than those of B3 by 0.44 m and 1.72 m, respectively. Furthermore, the mean and 95th percentile positioning errors of P2 are only 5.42 m and 12.31 m higher than those of the fully-supervised positioning baseline B1. Here, we reiterate that the proposed method P2 does not require any ground-truth UE position information as opposed to B1 and B3; hence, it is remarkable that P2 achieves similar positioning performance to the supervised positioning baseline B1 and approximately the same positioning performance as the semi-supervised baseline B3, without the need for ground-truth UE position information.

V. CONCLUSIONS AND FUTURE WORK

We have proposed a novel bilateration loss that enables weakly-supervised channel charting in real-world coordinates. This loss utilizes the known AP locations in a multipoint scenario by comparing the received power at two APs and placing the UE closer to the one with higher power. Using channel vectors from a commercial ray-tracer, we have demonstrated that combining the bilateration loss with the triplet loss from [19] is sufficient to generate high-quality channel charts in real-world coordinates, without the need of geometric models, accurate AP synchronization, or ground-truth UE positions.

There are many avenues for future work. We expect that loss functions that compare AP angle-of-arrival information besides receive power would further improve the quality of the learned channel charts. Extending our approach to indoor scenarios and measured channels under non-line-of-sight conditions, which requires AP selection, is part of ongoing work.

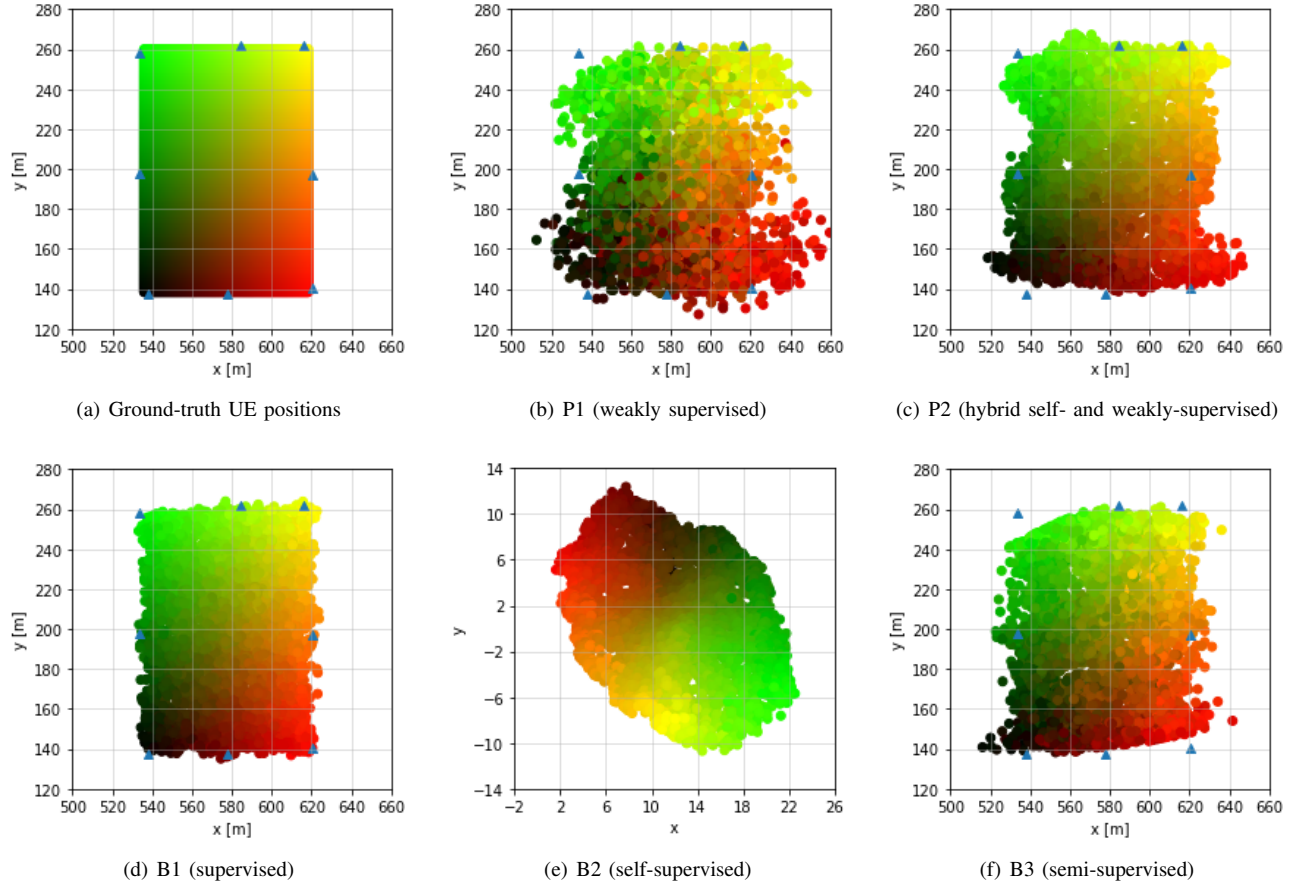


Fig. 2. (a) UE ground-truth positions (green-to-red gradient-colored area), AP positions (blue triangles), and the channel charts or positioning estimates (b-f) for the proposed (P) and baseline (B) methods. Since the output of baseline B2 is in arbitrary coordinates, the AP positions are not shown in (e). The proposed method P2 achieves comparable results in real-world coordinates as the semi-supervised baseline B3 but without requiring known UE positions during training.

TABLE II
CHANNEL CHARTING AND POSITIONING PERFORMANCE COMPARISON.

Method	Loss function to learn g_θ	Learning type	Figure	Latent space quality metrics				Positioning error [m]	
				TW \uparrow	CT \uparrow	KS \downarrow	RD \downarrow	Mean \downarrow	95th percentile \downarrow
P1	Eq. 4, bilateration	Weakly-supervised	2 (b)	0.918	0.922	0.26	0.88	15.02	32.05
P2	Eq. 5, bilateration and triplet	Hybrid self- and weakly-supervised	2 (c)	0.990	0.990	0.12	0.71	7.64	16.82
B1	Eq. 6, MSE	Fully-supervised	2 (d)	0.998	0.998	0.04	0.45	2.22	4.51
B2	Eq. 2, triplet	Self-supervised	2 (e)	0.991	0.992	0.17	0.79	—	—
B3	Eq. 7, triplet and $\widetilde{\text{MSE}}$	Semi-supervised	2 (f)	0.980	0.980	0.13	0.72	7.20	15.10

REFERENCES

- [1] Remcom, Inc. (2022) Wireless InSite. [Online]. Available: <https://www.remcom.com/wireless-insite-em-propagation-software/>
- [2] C. Studer, S. Medjkouh, E. Gönültaş, 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, Aug. 2018.
- [3] 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, Dec. 2022.
- [4] L. L. Magoarou, T. Yassine, S. Paquelet, and M. Crussière, “Channel charting based beamforming,” in *Proc. Asilomar Conf. Signals, Syst., Comput.*, Oct. 2022, pp. 1185–1189.
- [5] P. Kazemi, H. Al-Tous, C. Studer, and O. Tirkkonen, “Channel charting assisted beam tracking,” in *Proc. IEEE Veh. Technol. Conf. (VTC)*, Jun. 2022, pp. 1–5.
- [6] T. Kallehauge, A. E. Kalør, P. Ramírez-Espinosa, M. Guillaud, and P. Popovski, “Delivering ultra-reliable low-latency communications via statistical radio maps,” *IEEE Wireless Commun. Mag.*, vol. 30, no. 2, pp. 14–20, Apr. 2023.
- [7] P. Ferrand, M. Guillaud, C. Studer, and O. Tirkkonen, “Wireless channel charting: Theory, practice, and applications,” *IEEE Commun. Mag.*, vol. 61, no. 6, pp. 124–130, Jun. 2023.
- [8] J. Deng, S. Medjkouh, N. Malm, O. Tirkkonen, and C. Studer, “Multipoint channel charting for wireless networks,” in *Proc. Asilomar Conf. Signals, Syst., Comput.*, Oct. 2018, pp. 286–290.
- [9] F. Euchner, P. Stephan, M. Gauger, S. Dörner, and S. Ten Brink, “Improving triplet-based channel charting on distributed massive MIMO measurements,” in *Proc. IEEE Int. Workshop Signal Process. Advances Wireless Commun. (SPAWC)*, Jul. 2022, pp. 1–5.
- [10] P. Agostini, Z. Utkovski, and S. Stańczak, “Federated learning for multipoint channel charting,” in *Proc. IEEE Int. Workshop Signal Process. Advances Wireless Commun. (SPAWC)*, Jul. 2022, pp. 1–5.
- [11] J. Pihlajasalo, M. Koivisto, J. Talvitie, S. Ali-Löytty, and M. Valkama, “Absolute positioning with unsupervised multipoint channel charting for

- 5G networks,” in *Proc. IEEE Veh. Technol. Conf. (VTC)*, Nov. 2020, pp. 1–5.
- [12] M. Stahlke, G. Yammine, T. Feigl, B. M. Eskofier, and C. Mutschler, “Indoor localization with robust global channel charting: A time-distance-based approach,” *IEEE Trans. on Mach. Learn. in Commun. and Netw.*, pp. 1–1, Mar. 2023, early access.
- [13] P. Huang, O. Castañeda, E. Gönültaş, S. Medjkouh, O. Tirkkonen, T. Goldstein, and C. Studer, “Improving channel charting with representation-constrained autoencoders,” in *Proc. IEEE Int. Workshop Signal Process. Advances Wireless Commun. (SPAWC)*, Aug. 2019, pp. 1–5.
- [14] E. Lei, O. Castañeda, O. Tirkkonen, T. Goldstein, and C. Studer, “Siamese neural networks for wireless positioning and channel charting,” in *Proc. Allerton Conf. Commun., Contr., Comput.*, Sep. 2019, pp. 200–207.
- [15] J. Deng, O. Tirkkonen, J. Zhang, X. Jiao, and C. Studer, “Network-side localization via semi-supervised multi-point channel charting,” in *Int. Wireless Commun. and Mobile Comput. (IWCMC)*, Jun. 2021, pp. 1654–1660.
- [16] I. Karmanov, F. G. Zanjani, I. Kadampot, S. Merlin, and D. Dijkman, “WiCluster: Passive indoor 2D/3D positioning using WiFi without precise labels,” in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Dec. 2021, pp. 1–7.
- [17] Q. Zhang and W. Saad, “Semi-supervised learning for channel charting-aided IoT localization in millimeter wave networks,” in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Dec. 2021, pp. 1–6.
- [18] J. Deng, W. Shi, J. Hu, and X. Jiao, “Semi-supervised t-SNE for millimeter-wave wireless localization,” in *Int. Conf. on Comput. and Commun. (ICCC)*, Dec. 2021, pp. 1015–1019.
- [19] P. Ferrand, A. Decurninge, L. G. Ordoñez, and M. Guillaud, “Triplet-based wireless channel charting: Architecture and experiments,” *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2361–2373, Jun. 2021.
- [20] L. van der Maaten, E. Postma, and J. van den Herik, “Dimensionality reduction: a comparative review,” *J. Mach. Learn. Research*, vol. 10, no. 1–41, Oct. 2009.
- [21] E. Gönültaş, E. Lei, J. Langerman, H. Huang, and C. Studer, “CSI-based multi-antenna and multi-point indoor positioning using probability fusion,” *IEEE Trans. Wireless Commun.*, vol. 21, no. 4, pp. 2162–2176, Apr. 2022.
- [22] J. Vieira, E. Leitinger, M. Sarajlic, X. Li, and F. Tufvesson, “Deep convolutional neural networks for massive MIMO fingerprint-based positioning,” in *Proc. IEEE Intl. Symp. Personal, Indoor, Mobile Radio Commun.*, Oct. 2017, pp. 1–6.
- [23] T. Yassine, L. L. Magoarou, S. Paquelet, and M. Crussière, “Leveraging triplet loss and nonlinear dimensionality reduction for on-the-fly channel charting,” in *Proc. IEEE Int. Workshop Signal Process. Advances Wireless Commun. (SPAWC)*, Jul. 2022, pp. 1–5.
- [24] B. Rappaport, E. Gönültaş, J. Hoydis, M. Arnold, P. K. Srinath, and C. Studer, “Improving channel charting using a split triplet loss and an inertial regularizer,” in *Int. Symp. on Wireless Commun. Sys.s (ISWCS)*, Sep. 2021, pp. 1–6.
- [25] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Proc. Thirteenth Int. Conf. Artificial Intelligence Stat.*, vol. 9, May 2010, pp. 249–256.
- [26] P. Ferrand, A. Decurninge, and M. Guillaud, “DNN-based localization from channel estimates: Feature design and experimental results,” in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Dec. 2020, pp. 1–6.
- [27] H. Al-Tous, P. Kazemi, C. Studer, and O. Tirkkonen, “Channel charting with angle-delay-power-profile features and earth-mover distance,” in *Proc. Asilomar Conf. Signals, Syst., Comput.*, Oct. 2022, pp. 1195–1201.
- [28] Á. Vathy-Fogarassy and J. Abonyi, “Graph-based clustering and data visualization algorithms,” in *Springer Briefs in Computer Science*, May 2013.
- [29] J. B. Kruskal, “Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis,” *Psychometrika*, vol. 29, no. 1, pp. 1–27, Mar. 1964.
- [30] C. Rajsiki, “A metric space of discrete probability distributions,” *Information and Control*, vol. 4, no. 4, pp. 371–377, Dec. 1961.