

# Channel Charting: Locating Users within the Radio Environment using Channel State Information

Christoph Studer, Saïd Medjkouh, Emre Gönültaş, Tom Goldstein, and Olav Tirkkonen

**Abstract**—We propose *channel charting (CC)*, a novel framework in which a multi-antenna network element learns a chart of the radio geometry in its surrounding area. The channel chart captures the local spatial geometry of the area so that points that are close in space will also be close in the channel chart and vice versa. CC works in a fully unsupervised manner, i.e., learning is only based on channel state information (CSI) that is passively collected at a single point in space, but from multiple transmit locations in the area over time. The method then extracts channel features that characterize large-scale fading properties of the wireless channel. Finally, the channel charts are generated with tools from dimensionality reduction, manifold learning, and deep neural networks. The network element performing CC may be, for example, a multi-antenna base-station in a cellular system and the charted area in the served cell. Logical relationships related to the position and movement of a transmitter, e.g., a user equipment (UE), in the cell can then be directly deduced from comparing measured radio channel characteristics to the channel chart. The unsupervised nature of CC enables a range of new applications in UE localization, network planning, user scheduling, multipoint connectivity, hand-over, cell search, user grouping, and other cognitive tasks that rely on CSI and UE movement relative to the base-station, without the need of information from global navigation satellite systems.

**Index Terms**—Autoencoders, deep learning, dimensionality reduction, localization, machine learning, manifold learning, massive multiple-input multiple-output (MIMO), Sammon’s mapping.

## I. INTRODUCTION

**F**UTURE wireless communication systems must sustain a massive increase in traffic volumes, number of terminals, and reliability/latency requirements [2], [3]. In order to cope with these challenges, researchers have proposed a range of new

C. Studer, S. Medjkouh, and E. Gönültaş are with the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY; e-mail: studer@cornell.edu, sm2685@cornell.edu, eg566@cornell.edu; website: <http://vip.ece.cornell.edu>

T. Goldstein is with the Department of Computer Science, University of Maryland, College Park, MD; e-mail: tomg@cs.umd.edu

O. Tirkkonen was a visiting professor at the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY, and is now at the School of Electrical Engineering, Aalto University, Finland; e-mail: olav.tirkkonen@aalto.fi

The work of CS, SM, and EG was supported in part by Xilinx Inc., and by the US NSF under grants ECCS-1408006, CCF-1535897, CAREER CCF-1652065, and CNS-1717559. The work of EG research was supported in part by a fellowship from the Ministry of National Education of the Republic of Turkey. T. Goldstein was supported by DARPA Lifelong Learning FA8650-18-2-7833, DARPA YFA D18AP00055, ONR N00014-17-1-2078, NSF CCF-1535902, and the Sloan Foundation. The work of OT was funded in part by the Kaute foundation, Nokia Foundation, and the Academy of Finland (grant 299916).

A short version of this paper will appear at the 2018 IEEE Global Communications Conference [1].

The authors would like to thank O. Castañeda, P. Huang, and J. Deng for helpful comments on this manuscript.

technologies that improve spectral efficiency through massive multiple-input multiple-output (mMIMO) [4]–[6], increase bandwidth by harnessing millimeter-wave (mmWave) bands for mobile communication [7], and rely on an extreme densification of network elements [8].

While the advantages of these emerging technologies are glaring, they entail severe practical challenges. Mobility, in particular, poses problems for dense small-cell networks [9], as well as for mMIMO and mmWave networks, which provide extremely fine-grained angular separation. In mmWave networks, coverage is often patchy and hand-over regions between cells are sharp [10], [11]. Hence, smooth cell hand-over, multipoint operation, and/or cell search requires multipoint channel-state information (CSI) [12]. However, potential solutions to some of these issues, such as integrated multiband operation [13] or mobile relaying [14], will require significant amounts of multi-point CSI.

To effectively manage and optimize these technologies, future wireless systems must lean heavily on the availability of large amounts of high-dimensional CSI acquired at a multi-antenna base-station (BS) over large bandwidths and at fast rates, and from a large number of user equipments (UEs). To effectively use the collected CSI, the network has to *learn* the radio geometry in which the UEs are moving. What needs to be learned is a *chart* of the network radio geometry, which represents UE location and velocity information *related to CSI*. In order to automate functions, dynamically adapt to changes in the environment, and avoid human intervention for training, learning the radio geometry should be *unsupervised*.

It is remarkable that this problem has not been approached in the literature. Significant effort has been spent on *wireless localization or positioning* [15]–[17]. In addition, use-case specific *fingerprinting* methods have been developed in, e.g., [9], [18]–[23], with recent developments applying state-of-the-art deep learning methods for mMIMO channel fingerprinting [24]. However, fingerprinting methods are *fully supervised*, do not lend themselves to automation while acquiring labeled data, and do not scale to complex channel environments that change dynamically. Note that supervision achieved by acquiring precise location information from application layer localization services, such as global navigation satellite systems (GNSS), does *not* apply when optimizing cellular networks—the application layer is not even present in the whole LTE radio access network [25].

### A. Contributions

We introduce *channel charting* (CC), a novel framework that maps slowly varying CSI components of transmitters (e.g., UEs) into a low-dimensional channel chart that preserves the local geometry of the transmitters' true spatial locations. We show that by collecting and processing large amounts of high-dimensional CSI, one can accurately learn such channel charts in an unsupervised fashion, i.e., without access to location information. Our key contributions are summarized as follows:

- We propose a novel framework, which we call *channel charting*, that maps CSI acquired from UEs in a cell into a low-dimensional map that captures the local geometry of the true UEs' location in space. Channel charting is unsupervised, i.e., does *not* require any information from the true UEs' location, e.g., obtained from global navigation satellite systems (GNSSs).
- We describe how suitable CSI features can be extracted from channel measurements. More specifically, we identify that taking the absolute value of the raw second moment (R2M) in the angular (or beam-space) domain delivers features that exhibit high trustworthiness and continuity.
- We show analytically (in Example 2), that the R2M captures large-scale fading components of wireless channels which is key to enabling the concept of channel charting.
- We develop three new channel charting algorithms by extending existing manifold learning and dimensionality reduction techniques and adapting them to the tasks of channel charting. We emphasize that we are not simply providing a survey of existing methods but rather adapting and modifying them for our framework. Furthermore, the Sammon's mapping plus (SM+) method which includes side information on the UEs movement and the corresponding numerical algorithm developed in Section V-B3 has not been proposed in the literature—not even for other, unrelated machine learning applications that use manifold learning or dimensionality reduction.
- We provide a range of numerical simulation results for three distinct channel models to demonstrate the efficacy of our channel charting framework. More specifically, our results show that channel charting is feasible in line-of-sight (LoS) and non-LoS scenarios, and performs surprisingly well at relatively low signal-to-noise-ratio (SNR).

We envision a range of possible future applications for channel charting in cognitive tasks that rely on CSI and UE movement relative to the BSs, including semantic localization [26], cell search, hand-over and multi-connectivity [9], [12]–[14], [27]–[29], link adaptation, user clustering, beam finding, etc. However, being the first paper on the subject, this study is intended to (i) introduce and validate the fundamental concept of CC and (ii) compare a wide range of possible algorithms to maximize the quality of the learned channel charts.

### B. Relevant Prior Art

To the best of the authors' knowledge, direct charting of the radio geometry of the UEs has not been addressed in the open literature. All existing approaches are related to localizing UEs in the true spatial geometry. Alternatively, look-up tables based

on supervised fingerprinting have been used to identify use-case specific states of the channel. Conventional methods to localize UEs in spatial geometry are mainly based on triangulation or trilateration methods which use fixed geometrical models to map a low-level descriptor of the channel, such as the time-of-flight (ToF), angle-of-arrival (AoA), and/or received signal strength (RSS) to a location in spatial geometry [15], [16]. Localization in a mMIMO system based on ToF and AoA measurements has been addressed recently in [17]. However, to provide a chart in radio geometry, such methods would have to be complemented with a map from spatial to radio geometry.

A digital map is essentially a spatial geometry map that associates radio geometry features (e.g., RSS) with a given spatial location. Such maps have been created either by prediction models (e.g., by network planning tools) or by carrying out dedicated measurement campaigns [15], i.e., either based on analytical models, or in a fully supervised manner.

Similarly, for channel *fingerprinting* [9], [18]–[22], a coarse grained channel map is generated in a measurement campaign [18]–[22] or by directly classifying RSS measurement features by a network event, such as the vicinity of a small cell [9]. More refined fingerprinting has been proposed in [23], where mMIMO channel states are fingerprinted for positioning purposes. In [24], state-of-the-art deep learning methods are used for this purpose. Existing fingerprinting methods are, however, *fully supervised*. This implies that changes in the physical channel (e.g., new buildings) would require a completely new measurement campaign. Furthermore, the method in [24] requires training of the channel at wavelength scales in space. In contrast, CC is unsupervised, which avoids costly measurement campaigns, and requires orders-of-magnitude less dense spatial sampling.

In channel charting, we are primarily interested in preserving the local neighborhood structure of the spatial geometry when charting the radio geometry. For this, we shall use and extend tools from manifold learning [30], [31] and dimensionality reduction [32]. Multidimensional scaling (MDS) [30] and Sammon's mapping [31] attempt to embed a high-dimensional manifold into a low-dimensional space. We will show how CSI can be transformed into suitable channel features that enable an unsupervised extraction of accurate channel charts using such manifold learning and dimensionality reduction tools.

### C. Paper Outline

The rest of the paper is organized as follows. Section II introduces the principles of CC. Section III details the used quality measures. Section IV discusses suitable channel features that enable accurate CC. Section V proposes three different CC algorithms. Section VI shows CC results for a range of channel scenarios. We conclude in Section VII.

### D. Notation

Lowercase and uppercase boldface letters stand for column vectors and matrices, respectively. For the matrix  $\mathbf{A}$ , the Hermitian is  $\mathbf{A}^H$  and the  $k$ th row and  $\ell$ th column entry is  $A_{k,\ell}$  or  $[\mathbf{A}]_{k,\ell}$ . For the vector  $\mathbf{a}$ , the  $k$ th entry is  $a_k$ . The Euclidean norm of  $\mathbf{a}$  and the Frobenius norm of  $\mathbf{A}$  are denoted by  $\|\mathbf{a}\|_2$

and  $\|\mathbf{A}\|_F$ , respectively. The  $M \times N$  all-zeros and all-ones matrix is  $\mathbf{0}_{M \times N}$  and  $\mathbf{1}_{M \times N}$ , respectively, and the  $M \times M$  identity is  $\mathbf{I}_M$ . The collection of  $K$  vectors  $\mathbf{a}_k$ ,  $k = 1, \dots, K$ , is denoted by  $\{\mathbf{a}_k\}_{k=1}^K$ . The real and imaginary parts of the vector  $\mathbf{a}$  are denoted by  $\Re(\mathbf{a})$  and  $\Im(\mathbf{a})$ , respectively.

## II. THE PRINCIPLES OF CHANNEL CHARTING

We now introduce the core ideas of CC. We first discuss the main objective and then detail the operating principles as well as the underlying assumptions.

### A. Main Objective

The main objective of CC is to *learn* a low-dimensional embedding, the so-called *channel chart*, from a large amount of high-dimensional CSI of transmitters (e.g., mobile or fixed UEs) at different spatial locations over time. This channel chart *locally* preserves the original spatial geometry, i.e., transmitters that are nearby in real space will be placed nearby in the low-dimensional channel chart and vice versa. CC will learn whether two transmitters are close to each other by forming a *dissimilarity measure* [33] between CSI features of these transmitters. Based on this, CC generates the low-dimensional channel chart in an *unsupervised* fashion from CSI only and without assumptions on the physical channel, i.e., without the aid of information from GNSS, such as the global positioning system (GPS), triangulation/trilateration techniques, or fingerprinting-based localization methods [15], [16]. This important property enables CC to extract geometry information about the transmitters' in a completely passive manner, opening up a broad range of novel applications.

**Example 1.** Figure 1 demonstrates the key concepts of CC: (a) shows the considered scenario. A massive MIMO BS with a uniform linear array (ULA) of  $B = 32$  antennas receives data from  $N = 2048$  UE locations. We simulate a narrowband, line-of-sight (LoS) channel at a signal-to-noise ratio (SNR) of 0 dB (see Section VI for more details). (b) illustrates the relation between carefully-designed channel features (obtained solely from CSI) and UE locations. The scatter plot consists of points representing pairs of transmitters. For each pair, there is a point, with  $x$ -value being the pairwise spatial distance and  $y$ -value the pairwise feature dissimilarity. The used CSI-features and dissimilarities are discussed in Section IV. The channel features are designed to ensure that the pairwise feature dissimilarity is approximately lower-bounded by the pairwise spatial distance (when divided by a suitable reference distance). Thus, UEs that are far apart in space will have dissimilar channel features. (c) shows the resulting chart of one of our unsupervised CC algorithms. We observe that the local geometric features of the original spatial geometry are well-preserved. In fact, we recover the "VIP" curve (which are UEs positioned in space to form a contiguous curve) in the channel chart.

### B. Operating Principles of Channel Charting

Figure 2 provides a high-level overview of the CC framework. Consider, for the sake of simplicity, a single-antenna transmitter (Tx) that is either static or moves in real space. We denote its

spatial locations at discrete time instants  $n = 1, \dots, N$  by the set  $\{\mathbf{x}_n\}_{n=1}^N$  with  $\mathbf{x}_n \in \mathbb{R}^D$ , where  $D$  is the dimensionality of the *spatial geometry* (for example the three dimensions representing the UE's  $x$ ,  $y$ , and  $z$  coordinates in real space). At each time instant  $n$ , the Tx sends data  $\mathbf{s}_n$  (e.g., pilots or information symbols), which is received at a multi-antenna receiver (Rx) with  $B$  antennas; this could be a mMIMO BS [4]–[6]. The received data is modeled as  $\mathbf{y}_n = H(\mathbf{s}_n) + \mathbf{n}_n$ , where the function  $H(\cdot)$  represents the wireless channel between the transmitter and receiver, and the vector  $\mathbf{n}_n$  models noise.

1) *Channel Function:* In what follows, we are not interested in the transmitted data but rather in the associated CSI. Concretely, the Rx uses the received data  $\mathbf{y}_n$  to extract CSI denoted by the vector  $\mathbf{h}_n \in \mathbb{C}^M$ , where  $M$  denotes the dimensionality of the acquired CSI from all antennas, frequencies, and/or delays. The generated CSI typically describes angle-of-arrival, power delay profile, Doppler shift, RSS, signal phase, or simply first and second moments (e.g., mean and covariance) of the received data; typically, we have  $M \gg D$ . We denote the mapping from spatial location  $\mathbf{x}_n$  to CSI  $\mathbf{h}_n$  with the following channel function:

$$\mathcal{H} : \mathbb{R}^D \rightarrow \mathbb{C}^M,$$

where  $\mathbb{C}^M$  refers to the *radio geometry*. Clearly, the CSI represented by  $\mathbf{h}_n$  mainly depends on the Tx's spatial location  $\mathbf{x}_n$ , but also on moving objects within the cell, as well as on noise and interference. Throughout this paper, we make the following key assumption:

**Assumption 1.** We assume that the statistical properties of the multi-antenna channel vary relatively slowly across space, on a length-scale related to the macroscopic distances between scatterers in the channel, not on the small fading length-scale of wavelengths. We furthermore assume the channel function  $\mathcal{H}$  to be static<sup>1</sup>.

Large-scale effects of channels are considered to be created by reflection, diffraction, and scattering of the physical environment, whereas small-scale effects are caused by multipath propagation and related destructive/constructive addition of signal components [34]. To motivate Assumption 1, we consider the following example, which demonstrates that the statistical moments of interest for this paper (see Section IV) indeed capture large-scale effects of the wireless channel.

**Example 2.** The channel between a single Tx and a  $B$ -antenna Rx is modeled with a set of rays and we assume  $N_s$  scatterers. We consider a NLoS scenario for which all rays are in the far field, so that they can be modeled by plane waves. The distance from Tx  $t$  to scatterer  $s$  is  $d_{ts}$ , and the distance from scatterer  $s$  to Rx-antenna  $r$  is  $d_{sr}$ . The attenuation between two points  $x$  and  $y$  is modeled by a function of the distance,  $a_{xy} = a(d_{xy})$ , which absorbs the relevant scatterer cross sections, antenna gains, etc. The distance dependence is typically a power law, and changes in  $a(d)$  happen on length scales much larger than the wavelength  $\lambda$ ; for conventional ray-tracing,  $a(d) \sim d^{-2}$ , corresponding to free-space path loss [35]. In addition, each

<sup>1</sup>An extension to time-varying channels is part of ongoing research.

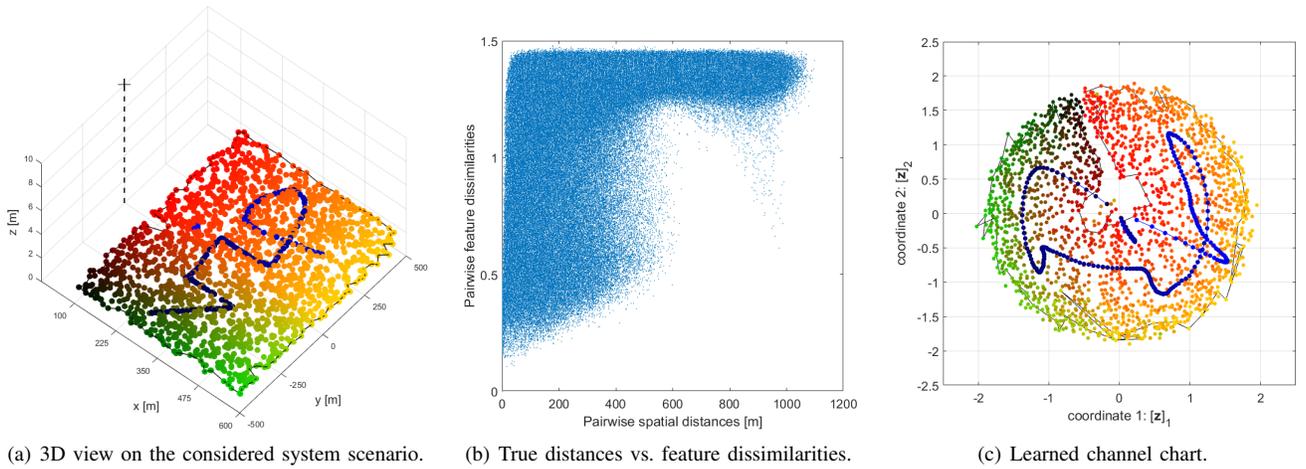


Fig. 1. Illustration of channel charting: (a) Massive MIMO BS with a 32-antenna uniform linear array at location  $(x, y, z) = (0, 0, 10)$  meters measures CSI from 2048 distinct gradient colored points in space. (b) Scatter plot with points representing pairs of transmitters. For each pair, a point with pairwise spatial distance vs. pairwise feature dissimilarity, constructed from CSI. The dissimilarity is lower-bounded by the pairwise spatial distance. (c) Channel chart obtained from channel features in an unsupervised manner. The blue points that form the “VIP” curve illustrate the properties of the channel chart: local geometry is well-preserved (shown by the color gradients) and one can identify “VIP” in the channel chart.

scatterer  $s$  is modeled by a phase shift  $\phi_s$ , related to the dielectric properties of the scatterer [36], [37]; these are assigned i.i.d. random variables for each scatterer. The channel between  $t$  and  $r$  can thus be modeled as

$$h_{t,r} = \sum_{s=1}^{N_s} a_{ts} a_{sr} \exp\left(j\left(\frac{2\pi}{\lambda}(d_{ts} + d_{sr}) + \phi_s\right)\right).$$

When the number of scatterers  $N_s \rightarrow \infty$ , the channel becomes Rayleigh fading. This is a characteristic of the distribution of the absolute value of the channel coefficients, when considered a random variable, where randomness is according to the location of the transmitter within a small scale neighborhood of a few wavelengths. Long term channel characteristics are averaged over this neighborhood. For a mean of a MIMO channel, as a large-scale channel feature that describes the statistics of small scale fading, the pertinent characteristics are thus the mean absolute value of the channel at each antenna  $r$ , and the mean relative phase difference between antennas. For the means, following [35], and averaging over a small scale neighborhood of a few wavelengths, one finds that the wavelength ( $\lambda$ ) dependence vanishes. For the angular difference, a similar argument leads to the observation that they are large-scale effects of the channel.

Concretely, evaluating the raw  $2^{\text{nd}}$  moment of the channel from Tx  $t$  to Rx antennas  $r, r'$  yields

$$\begin{aligned} \left[\mathbb{E}_\phi[\mathbf{h}_t \mathbf{h}_t^H]\right]_{r,r'} &= \sum_{s=1}^{N_s} \sum_{s'=1}^{N_s} \mathbb{E}_\phi \left[ a_{ts} a_{t s'} e^{j\frac{2\pi}{\lambda}(d_{ts} - d_{t s'})} \right. \\ &\quad \left. \times a_{s r} a_{s' r'} e^{j\frac{2\pi}{\lambda}(d_{s r} - d_{s' r'}) + j(\phi_s - \phi_{s'})} \right] \\ &= \sum_{s=1}^{N_s} a_{ts}^2 a_{s r} a_{s r'} e^{j\frac{2\pi}{\lambda}(d_{s r} - d_{s r'})}, \end{aligned}$$

where for clarity, we have considered the expectation over the random phases  $\phi$  only, assuming that the distances are fixed. In the limit, this expression changes only slowly

with the distances  $d_{ts}$  through the attenuation function  $a_{ts}$ . Now consider the (raw) covariance matrix estimated for two transmitters  $t$  and  $t'$ . If  $a_{ts} \approx a_{t's}$  for all scatterers  $s$ , then the covariance matrices  $\mathbf{R}_t$  and  $\mathbf{R}_{t'}$  are approximately the same. The covariance matrices differ only at length scales where the change in the distances between the transmitter and the scatterers is significant—changes in the channel covariance is a large-scale fading effect, driven by the quenched random process that creates the scatterers in the environment.

2) *Channel Charting*: By relying on Assumption 1, we are ready to detail the CC procedure. CC starts by distilling the CSI  $\mathbf{h}_n$  into suitable *channel features*  $\mathbf{f}_n \in \mathbb{R}^{M'}$  that capture large-scale properties of the wireless channel; here,  $M'$  denotes the feature dimension and, typically, we have  $M' \gg D$ . See Section IV for the details on how to design channel features. We denote the feature extraction stage by the function

$$\mathcal{F} : \mathbb{C}^M \rightarrow \mathbb{C}^{M'}.$$

Feature extraction mainly serves three purposes: (i) extracting large-scale fading properties from CSI, (ii) distilling CSI into useful information for the subsequent CC pipeline, and (iii) reducing the vast amount of channel data. CC then proceeds by using the set of  $N$  collected features  $\{\mathbf{f}_n\}_{n=1}^N$  to learn the so-called *forward charting function* (with possible side information; see Section V-B) in an unsupervised manner. We denote the forward charting function to be learned by

$$\mathcal{C} : \mathbb{C}^{M'} \rightarrow \mathbb{R}^{D'},$$

which maps each channel feature  $\mathbf{f}_n$  to a point  $\mathbf{z}_n \in \mathbb{R}^{D'}$  in the low-dimensional *channel chart*; typically, we have  $D' \approx D$ . The objective for learning  $\mathcal{C}$  is as follows:

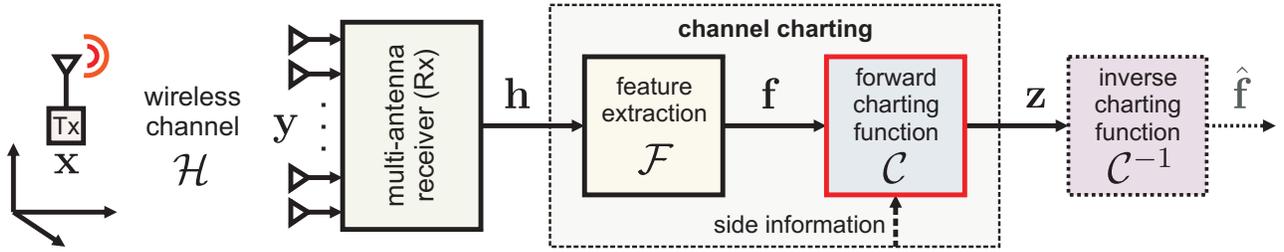


Fig. 2. Channel charting (CC) overview. Mobile transmitters (Tx) at spatial location  $\mathbf{x}$  are sending information to a multi-antenna receiver (Rx) over the wireless channel  $\mathcal{H}$ . Channel charting first uses channel-state information (CSI)  $\mathbf{h}$  to extract channel features  $\mathbf{f}$ , which are then processed by a channel charting algorithm to learn a forward charting function  $\mathcal{C}$  that generates an embedding in spatial geometry  $\mathbf{z}$  that preserves local geometry in an unsupervised manner.

The forward charting function  $\mathcal{C}$  should preserve local geometry between neighboring data points, i.e., it should satisfy the following condition:

$$d_z(\mathbf{z}, \mathbf{z}') \approx d_x(\mathbf{x}, \mathbf{x}').$$

Here,  $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^D$  are two points in real space within a certain neighborhood, and  $\mathbf{z}, \mathbf{z}' \in \mathbb{R}^{D'}$  are the corresponding vectors in the learned channel chart. The functions  $d_x(\mathbf{x}, \mathbf{x}')$  and  $d_z(\mathbf{z}, \mathbf{z}')$  are suitably defined measures of distance (or, more generally, dissimilarity) and the neighborhood size depends on the physical channel.

The goal of CC is to generate a channel chart  $\{\mathbf{z}_n\}_{n=1}^N$  satisfying the distance property above for  $\mathbf{x}$  and  $\mathbf{x}'$  in a neighborhood as large as possible. We would like to learn this channel chart solely from the set of  $N$  channel features  $\{\mathbf{f}_n\}_{n=1}^N$  in an unsupervised manner, i.e., without using the true spatial locations  $\{\mathbf{x}_n\}_{n=1}^N$  of the UEs.

**Remark 1.** The assumption that the channel features  $\{\mathbf{f}_n\}_{n=1}^N$  were obtained from a single transmitter (e.g., UE) is not important. In fact, we are merely interested in collecting  $N$  channel features from as many locations in spatial geometry as possible. The fact that certain subsets of channel features stem from a single UE can be used as potential side information, which improves the geometric relationships in the learned channel chart; see Section V-B for a concrete example.

### C. Involved Geometries and Usage of CC

Figure 3 provides a summary of the geometries involved in CC. The transmitters are located in spatial geometry denoted by  $\mathbb{R}^D$  (e.g., representing their coordinates). The physical wireless channel  $\mathcal{H}$  maps data (pilots and information) into CSI in radio geometry space denoted by  $\mathbb{C}^M$ . This non-linear mapping into radio geometry obfuscates the spatial relationships between transmitters. The purpose of feature extraction is to find a representation from which spatial geometry is easily recovered. CC then learns—in an unsupervised manner—the forward charting function  $\mathcal{C}$  that maps the channel features into low-dimensional points in the channel chart  $\mathbb{R}^{D'}$  such that neighboring transmitters (in real-world coordinates) will be neighboring points in the channel chart, i.e., CC preserves the

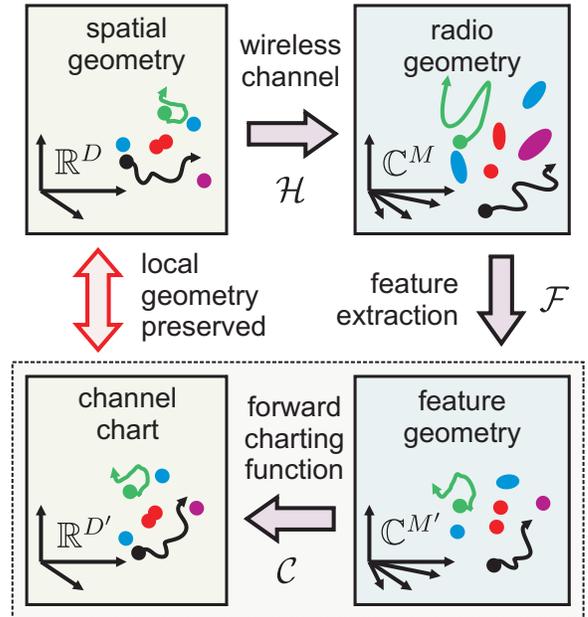


Fig. 3. Summary of the geometries involved in CC. Transmitters (Tx) are located in spatial geometry  $\mathbb{R}^D$  and a receiver (Rx) extracts channel-state information (CSI) in radio geometry  $\mathbb{C}^M$ . Feature extraction distills useful information into feature geometry  $\mathbb{C}^{M'}$ , which is then used to learn the forward charting function that maps the features into a low-dimensional channel map in  $\mathbb{R}^{D'}$  that preserves the local geometry of the original spatial locations  $\mathbb{R}^D$ .

local geometry. Note that in some application scenarios one may be interested in the inverse charting function  $\mathcal{C}^{-1}$  that maps channel charts information back into feature geometry.<sup>2</sup>

**Example 3.** An example of how CC could be used in practice is as follows. A mobile UE is served by a cellular network, and is connected to a particular BS. Conventionally, cell hand-over is executed based on RSS measurements performed at the UE. The UE continually monitors synchronization signals transmitted by all BSs in the network, and sends the measurement results to the BS. Handover is then reactively performed, according to these measurements. In a location-based mobility management scenario [40], to decrease signaling and UE measurements, the network proactively performs hand-over based on spatial

<sup>2</sup>For example, with  $\mathcal{C}^{-1}$ , the amount of multipoint CSI required for multipoint transmission [38] and interference alignment [39] can be reduced.

*localization of the UE. The user is first localized by fusing ToF and AoA measurements of multiple BSs. Based on the UE location, environment specific information is used to calculate the best cell. In a CC-based approach to cell hand-over, the BS would have a chart of the radio features in the cell served by it, labeled by locations where handover events have occurred. From uplink pilots transmitted by the UE, it may localize the UE in the radio geometry, and execute handover when the CC indicates a point where handovers happen. Note that in CC, the decision to execute handover is based on measurements at a single BS; network wide fusion is not required. In contrast, the location-based method discussed in the literature [40] applies both network-wide fusion for spatial localization, and side information related to propagation condition between a BS, and a UE at a given spatial location. Furthermore, by tracking and predicting a UE's movement in the channel chart, one can even anticipate cell hand-over events before they happen.*

#### D. Do We Have Sufficient CSI for Channel Charting?

To extract accurate channel charts in an unsupervised manner, we require high-dimensional CSI that is from as many distinct transmit locations as possible and acquired at multiple BS antennas over large bandwidths and at fast rates. Fortunately, virtually all modern wireless systems already generate high-dimensional CSI data at extremely fast rates.

**Example 4.** *A BS for 3GPP long-term evolution (LTE) [41] measures up to 100 MIMO channels each millisecond, leading to more than  $10^{10}$  complex-valued numbers per day for a  $2 \times 4$  MIMO channel. A similar amount of data is collected by active user equipments (UEs), which signal up to 226 bits of CSI to the BS every 2 ms [25]. Currently, most of that data is discarded immediately after use (e.g., for data detection or precoding), with a limited amount kept in order to track the average received signal strength (RSS) of the UEs.*

For CC, the idea is to collect and process the acquired CSI to learn channel charts. The total dimensionality  $M$  of each CSI vector is determined by the number of receiver antennas  $B$  times the number of subcarriers (or delays)  $W$ . As we will show in Section IV, we intentionally “lift” the CSI vectors into a higher dimensional space, effectively squaring the total feature dimension. We collect channel features from  $N$  distinct transmitter locations, which further amplifies the amount of data available for channel charting. Hence, the total number of channel features used for CC can easily be in the billions.

**Example 5.** *Consider a wideband massive MIMO receiver with  $B = 32$  BS antennas and  $W = 128$  subcarriers, which results in  $M = BW = 2^{12}$  dimensional CSI vectors. If we lift each CSI vector into an  $M' = M^2$  dimensional space, we have features with  $M' = 2^{24}$  dimensions. By collecting channel features from  $N = 2,048$  distinct spatial locations, we have a total dimension of  $2^{35}$ , which is a dataset containing more than 34 billion complex-valued channel feature coefficients.*

Note that these numbers are conservative. Fifth-generation (5G) wireless networks likely have many more BS antennas and subcarriers, and receive data from a large number of UEs.

This torrent of channel features is a blessing and a curse at the same time. Clearly, the proposed CC methods will have sufficient data to learn from. However, the vast amount of CSI poses severe challenges for storage and processing. Channel feature extraction must reduce the size of this data, and charting algorithms must scale appropriately. We will discuss suitable features in Section IV and computationally efficient CC algorithms in Section V.

### III. QUALITY MEASURES FOR CHANNEL FEATURES AND CHANNEL CHARTS

To characterize the usefulness of channel features and the quality of the generated channel charts, we need a measure of how well the channel features or points in the channel chart preserve the spatial geometry of the true transmitter locations—suitable features would preserve the local geometry for a neighborhood as large as possible. To assess the channel charting quality, we borrow two metrics typically used to measure the quality of dimensionality reduction methods, namely *continuity* (CT) and *trustworthiness* (TW) [42]–[44].

We next explain both of these quality measures in the context of two abstract sets of data points with cardinality  $N$ , i.e.,  $\{\mathbf{u}_n\}_{n=1}^N$  from an *original space* and  $\{\mathbf{v}_n\}_{n=1}^N$  from a *representation* of the original space; the point  $\mathbf{v}_n$  is said to represent  $\mathbf{u}_n$ . In the CC context, the original space would be the spatial geometry and the representation space can either be the feature geometry or the channel chart (see Figure 3), depending on whether we want to measure the quality of the channel features or of the learned channel chart.

In what follows, we define the  $K$ -neighborhood of a point  $\mathbf{u}$  as the set containing its  $K$  nearest neighbors in terms of the chosen distance (or dissimilarity) function  $d_u(\mathbf{u}, \mathbf{u}')$ . The neighborhood of  $\mathbf{v}$  is defined analogously using  $d_v(\mathbf{v}, \mathbf{v}')$ .

#### A. Continuity (CT)

Neighbors in the original space can be far away (dissimilar) in the representation space. In such situations, we say that the representation space does not preserve the *continuity* of the original point set. To measure such situations, we first define the *point-wise continuity* for  $K$  neighbors of the data point  $\mathbf{u}_i$ . Let  $\mathcal{V}_K(\mathbf{u}_i)$  be the  $K$ -neighborhood of point  $\mathbf{u}_i$  in the original space (but not necessarily in the representation space). Also, let  $\hat{r}(i, j)$  be the ranking of point  $\mathbf{v}_j$  among the neighbors of point  $\mathbf{v}_i$ , ranked according to their similarity to  $\mathbf{v}_i$ . For example,  $\hat{r}(i, j) = k$  indicates that point  $\mathbf{v}_j$  is the  $k$ th most similar point to  $\mathbf{v}_i$ . Then, the point-wise continuity of the representation  $\mathbf{v}_i$  of the point  $\mathbf{u}_i$  is defined as

$$\text{CT}_i(K) = 1 - \frac{2}{K(2N - 3K - 1)} \sum_{j \in \mathcal{V}_K(\mathbf{u}_i)} (\hat{r}(i, j) - K).$$

The (global) *continuity* between a point set  $\{\mathbf{u}_n\}_{n=1}^N$  and its representation  $\{\mathbf{v}_n\}_{n=1}^N$  is simply the average over all the point-wise continuity values, i.e.,  $\text{CT}(K) = \frac{1}{N} \sum_{i=1}^N \text{CT}_i(K)$  [42]. Both the point-wise and global continuity measures range between zero and one. If continuity is low (e.g., 0.5 or smaller), then points that are similar in the original space are dissimilar in the representation space. When continuity is large (close to 1), the representation mapping is neighbor preserving.

## B. Trustworthiness (TW)

Continuity measures whether neighbors in the original space are preserved in the representation space. However, it may be that the representation mapping introduces *new* neighbor relations that were absent in the original space. Trustworthiness measures how well the feature mapping avoids introducing these kinds of false relationships. Analogous to the point-wise continuity, we first define the point-wise *trustworthiness* for a  $K$ -neighborhood of point  $\mathbf{v}_i$ . Let  $\mathcal{U}_K(\mathbf{v}_i)$  be the set of “false neighbors” that are in the  $K$ -neighborhood of  $\mathbf{v}_i$ , but not of  $\mathbf{u}_i$  in the original space. Also, let  $r(i, j)$  be the ranking of point  $\mathbf{u}_j$  in the neighborhood of point  $\mathbf{u}_i$ , ranked according to their similarity to  $\mathbf{u}_i$ . The point-wise trustworthiness of the representation of point  $\mathbf{u}_i$  is then

$$\text{TW}_i(K) = 1 - \frac{2}{K(2N - 3K - 1)} \sum_{j \in \mathcal{U}_K(\mathbf{v}_i)} (r(i, j) - K).$$

The (global) *trustworthiness* between a point set  $\{\mathbf{u}_n\}_{n=1}^N$  and its representation  $\{\mathbf{v}_n\}_{n=1}^N$  is simply the average over all the point-wise trustworthiness values, i.e.,  $\text{TW}(K) = \frac{1}{N} \sum_{i=1}^N \text{TW}_i(K)$  [42]. Both the point-wise and global trustworthiness range between zero and one. Low trustworthiness values represent situations in which most data points that seem to be similar in representation space are actually dissimilar in the original space. If the trustworthiness lies close to one, then data points that are close in representation space are also similar (close) in original space.

**Remark 2.** *Since we are interested in preserving local geometry, we set  $K$  to 5% of the total number of points  $N$ , i.e.,  $K = 0.05N$ . Note that this is a common choice in the dimensionality-reduction literature [42].*

## C. Uses of CT and TW for Channel Charting

We will use the CT and TW measures for two purposes. First, we will use both measures to assess the quality of channel features  $\{\mathbf{f}_n\}_{n=1}^N$ . For this purpose, we measure CT and TW between the spatial geometry and the feature geometry (see Figure 3). See Section IV-D for a detailed analysis of channel features that preserve the CT and TW and, hence, are suitable for CC. Second, we will use these measures to assess the quality of the learned channel charts  $\{\mathbf{z}_n\}_{n=1}^N$ . For this purpose, we measure CT and TW between the spatial geometry and the channel chart. See Section VI for a comparison of the CC algorithms proposed in this paper.

## IV. CHANNEL FEATURES

We now focus on the feature extraction stage. Concretely, we show that computing the raw 2<sup>nd</sup> moment of CSI, feature scaling, and transforming the result in the angular domain yields channel features that accurately represent large-scale fading properties of wireless channels.

### A. Features from CSI via Moments

To limit the search for suitable channel features, we focus on Frobenius (or Euclidean) distance as dissimilarity measure

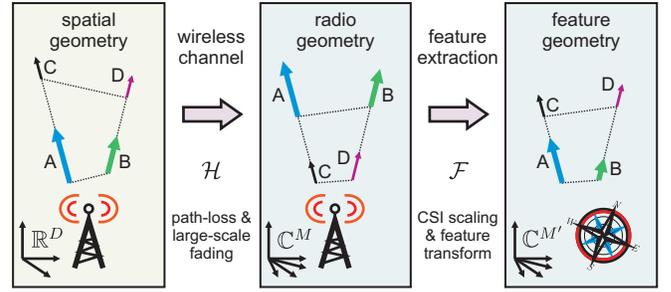


Fig. 4. Illustration of the importance of CSI scaling during feature extraction. The solid lines show the dissimilarity between the UEs A and B, as well as C and D in the various geometries. The dotted lines indicate the UEs located on the same incident rays, i.e., A and C, as well as D and B. In radio geometry, the acquired CSI misrepresents the true Tx distance due to path-loss. Concretely, UEs far away in spatial geometry appear similar in radio geometry and vice versa. To compensate for this distortion effect, we perform CSI scaling that unwraps radio geometry into feature geometry that better represents an Euclidean space.

on pairs of features, i.e., we use  $d_f(\mathbf{F}, \mathbf{F}') = \|\mathbf{F} - \mathbf{F}'\|_F$ , where (by abuse of notation) we allow the features to be matrices. To generate suitable channel features, we focus on a second order statistical moment of the received CSI. Let  $\mathbf{h}_t \in \mathbb{C}^M$  be a vector containing CSI acquired (e.g., during the training phase) at time instant  $t$ . We compute the *raw 2<sup>nd</sup> moment* (R2M) of dimension  $M^2$  as follows:  $\bar{\mathbf{H}} = \mathbb{E}[\mathbf{h}\mathbf{h}^H]$ . Here, expectation is over noise, interference, and potential variations in CSI caused by small-scale motion during short time (i.e., well-below the coherence time of the channel). It is important to note that computing the outer product leads to a representation of CSI that is agnostic to any global phase rotation that may stem from small-scale fading. In practice, we compute  $\bar{\mathbf{H}} = \frac{1}{T} \sum_{t=1}^T \mathbf{h}_t \mathbf{h}_t^H$  for a small number (e.g., ten or less) of time instants  $T$ . We can then use  $\bar{\mathbf{H}}$  to extract the necessary channel features in two steps: (i) CSI scaling and (ii) feature transform. Both of these steps are detailed next.

### B. Step 1: CSI Scaling

One of the most critical aspects in the design of good features for CC is to realize that CSI in radio geometry is a poor representation of spatial geometry. Figure 4 illustrates this aspect. Assume that the two Txes A and B are close to the Rx, and the Txes C and D are further away. Due to path-loss, the CSI measurements  $\mathbf{H}_C$  and  $\mathbf{H}_D$  of Txes C and D appear weaker (i.e., have small Frobenius norm) than those of the Txes nearby,  $\mathbf{H}_A$  and  $\mathbf{H}_B$ . If we now directly compare the Frobenius distance between C and D, their distance appears to be smaller than that between A and B (because they have small norm), even though they should be further apart. To compensate for this phenomenon, we “unwrap” the CSI so that it is more compatible with spatial geometry (see Figure 4). This approach is called *CSI scaling* and explained next.

Consider a transmitter that is separated  $d$  meters from a uniform linear array (ULA) with  $B$  antennas. Assume a narrowband LoS channel without scatterers and a 2-dimensional plane wave model (PWM). For this scenario, each entry  $b$  of

the normed<sup>3</sup> CSI vector  $\mathbf{h} \in \mathbb{C}^B$  is given by [45]

$$h_b = d^{-\rho} \exp\left(-j \frac{2\pi}{\lambda} \Delta r (b-1) \cos(\phi)\right) \quad (1)$$

for  $b = 1, \dots, B$ , where  $\rho > 0$  is the path-loss exponent,  $\Delta r$  is the antenna spacing, and  $\phi$  is the incident angle of the Tx to the Rx. Let  $\tilde{\mathbf{H}} = \mathbf{h}\mathbf{h}^H$  be the associated R2M. As in Figure 4, assume two Txs **A** and **C** with the same incident angle  $\phi$  but with distances  $d_A$  and  $d_C$  to the receiver. Our goal is now to scale the CSI matrices so that the Frobenius distance  $d_h(\tilde{\mathbf{H}}_A, \tilde{\mathbf{H}}_C) = \|\tilde{\mathbf{H}}_A - \tilde{\mathbf{H}}_C\|_F$  of the scaled moments  $\tilde{\mathbf{H}}_A$  and  $\tilde{\mathbf{H}}_C$  is exactly their true distance. For the above LoS scenario, we have the following result.

**Lemma 1.** *Consider the LoS channel model in (1). Assume two UEs **A** and **C** with the same incident angle  $\phi$ , with distances  $d_A$  and  $d_C$  to the BS. By scaling the R2M of both UEs as*

$$\tilde{\mathbf{H}} = \frac{B^{\beta-1}}{\|\tilde{\mathbf{H}}\|_F^\beta} \tilde{\mathbf{H}} \quad \text{with} \quad \beta = 1 + 1/(2\sigma), \quad (2)$$

the distance  $d_h(\tilde{\mathbf{H}}_A, \tilde{\mathbf{H}}_C) = \|\tilde{\mathbf{H}}_A - \tilde{\mathbf{H}}_C\|_F$  of the scaled moments  $\tilde{\mathbf{H}}_A$  and  $\tilde{\mathbf{H}}_C$  is exactly their true distance

$$d_h(\tilde{\mathbf{H}}_A, \tilde{\mathbf{H}}_C) = |d_A - d_C| \quad (3)$$

if the parameter  $\sigma \in (0, \infty]$  matches the path-loss exponent  $\rho$ .

*Proof.* The proof follows immediately from the requirement in (3) and the fact that both users **A** and **C** are associated with the same channel vectors  $\mathbf{h}$  given by the LoS model in (1) that only differ in terms of the path loss.  $\square$

Since  $\beta \geq 1$ , CSI from transmitters far away is amplified and nearby CSI is attenuated. In words, feature scaling as in (2) unwraps the radio geometry as illustrated in Figure 4.

**Remark 3.** *As the path-loss exponent  $\rho > 0$  is often unknown in practice, we can use the parameter  $\sigma$  in (2) as a tuning parameter. As shown in Section VI,  $1 \leq \sigma \leq 16$  yields excellent CC quality (in terms of TW and CT) for various scenarios. Furthermore, as seen from (2), the extreme case of  $\sigma \rightarrow \infty$  ignores the magnitude of CSI altogether; this is, for example, useful in multi-user systems that deploy transmit-power control or in scenarios in which shadowing effects are dominating.*

### C. Step 2: Feature Transform

We are now ready to transform the scaled CSI moments  $\tilde{\mathbf{H}}$  into channel features. Since we focus on the Frobenius distance as dissimilarity, a straightforward choice of a channel feature is to set the feature directly to the scaled CSI moments  $\mathbf{F} = \tilde{\mathbf{H}}$ ; we denote this feature by “ $\mathbb{C}\{\cdot\}$ ”. However, as we will show in Section IV-D, applying certain nonlinear transforms to the scaled CSI moments can significantly improve the feature quality. In particular, we also consider taking the entry-wise real part (denoted by “ $\Re\{\cdot\}$ ”), imaginary part (denoted by “ $\Im\{\cdot\}$ ”), angle (denoted by “ $\angle(\cdot)$ ”), or absolute value (denoted by “ $|\cdot|$ ”) of the scaled CSI moments. We furthermore say that all these channel features were taken in the *antenna domain* (denoted

TABLE I  
KEY PARAMETERS OF THE QUADRIGA NLOS CHANNEL MODEL [47]

Parameter	Setting
Scenario	BERLIN_UMa_NLOS
Carrier frequency	$f_c = 2.0$ GHz
Channel bandwidth	$BW = 312.5$ KHz
Number of BS antennas	$B = 32$
Antenna array	ULA with $\Delta r = \lambda/2$

by “Ant.”). We also consider the case in which we take the scaled CSI vectors and transform then into the *angular domain* (denoted by “Ang.”) followed by one of the nonlinearities mentioned above. For the scaled R2M, denoted by  $\tilde{\mathbf{H}}$ , we compute  $\mathbf{D}\tilde{\mathbf{H}}\mathbf{D}^H$ , where  $\mathbf{D}$  is the  $M \times M$  discrete Fourier transform matrix that satisfies  $\mathbf{D}^H\mathbf{D} = \mathbf{I}_M$ . This approach transforms the scaled CSI moments from the antenna domain into the angular (or beamspace) domain, which represents the incident angles of the Tx and potential scatterers to the array in a concise way [46]. We then either use this feature directly or apply one of the above mentioned nonlinearities.

### D. Feature Analysis and Comparison

We now evaluate the effectiveness of the channel features discussed above. We first detail the simulation parameters, and then evaluate the associated CT and TW measured between spatial geometry and radio geometry.

1) *Simulation Setup:* We consider a scenario as depicted in Figure 1(a) with a narrowband non-LoS (NLoS) channel generated from the Quadriga channel model [47]; the key parameters are summarized in Table I. We record CSI of  $N = 2048$  randomly selected (with the exception of the “VIP” curve, which have been placed to form a contiguous curve) spatial locations within a square area of  $1000 \text{ m} \times 500 \text{ m}$ ; the median distance between nearest neighbors is approximately 7.86 meters, i.e., we sample CSI in space at roughly 53 wavelengths. We acquire CSI at an SNR of 0 dB, average over  $T = 10$  time instants, and set  $\sigma = 16$ .

2) *Feature Comparison:* Table II summarizes the global TW and CT for a range of channel features with a neighborhood of  $K = 0.05N$ ; the numbers in the parentheses indicate the standard deviation over the point-wise TW and CT measures. We see that the absolute value of R2M in the angular domain yields high TW and CT values. Other features, such as the absolute value of the R2M in the *antenna domain* perform poorly. In summary, we observe that—given appropriate channel features—even challenging NLoS channel scenarios at low SNR exhibit surprisingly high TW and CT. This observation supports the validity of Assumption 1 and paves the way for the CC methods proposed next.

**Remark 4.** *We conducted the same experiments for a “vanilla” LoS (V-LoS) channel as in (1) as well as a Quadriga-based LoS (Q-LoS) channel, and we arrived at the same conclusions. We emphasize that absolute value of the R2M in the angular domain turned out to be the most robust channel feature for all considered channel models and scenarios.*

<sup>3</sup>The vector’s  $\mathbf{h}$  phase is rotated so that  $h_1$  is real and positive.

TABLE II  
COMPARISON OF CHANNEL FEATURES EXTRACTED FROM THE RAW 2<sup>ND</sup> MOMENT (R2M)  
IN TERMS OF GLOBAL TRUSTWORTHINESS (TW) AND CONTINUITY (CT) FOR  $K = 0.05N$

Domain		$\mathbb{C}\{\cdot\}$	$\mathbb{R}\{\cdot\}$	$\mathfrak{I}\{\cdot\}$	$\angle(\cdot)$	$ \cdot $
Antenna	TW	0.76 ( $\pm 0.11$ )	0.62 ( $\pm 0.12$ )	0.70 ( $\pm 0.09$ )	0.67 ( $\pm 0.09$ )	0.54 ( $\pm 0.07$ )
	CT	0.76 ( $\pm 0.07$ )	0.71 ( $\pm 0.07$ )	0.69 ( $\pm 0.08$ )	0.63 ( $\pm 0.08$ )	0.56 ( $\pm 0.09$ )
Angular	TW	see TW above	0.76 ( $\pm 0.12$ )	0.56 ( $\pm 0.08$ )	0.55 ( $\pm 0.07$ )	<b>0.81</b> ( $\pm 0.13$ )
	CT	see CT above	0.74 ( $\pm 0.07$ )	0.52 ( $\pm 0.06$ )	0.53 ( $\pm 0.09$ )	<b>0.84</b> ( $\pm 0.09$ )

## V. CHANNEL CHARTING ALGORITHMS

We now introduce three distinct CC algorithms with varying complexity, flexibility, and accuracy. We propose principal component analysis (PCA), Sammon's mapping (and a variation thereof), and autoencoders in the context of CC. For each method, we briefly discuss the pros and cons. Corresponding channel chart results are shown in Section VI.

### A. Principal Component Analysis

As a baseline charting algorithm, we perform PCA [48], [49] on a centered version of the channel features. PCA is among the most popular *linear* and *parametric* methods for dimensionality reduction and maps a high-dimensional point set (the channel features) into a low-dimensional point set (the channel chart) in an unsupervised manner. The specific method we use for channel charting is detailed next.

1) *Algorithm*: We collect all  $N$  channel features, vectorize them, and concatenate them in the  $M' \times N$  matrix  $\mathbf{F} = [\mathbf{f}_1, \dots, \mathbf{f}_N]$ . We then normalize each row of  $\mathbf{F}$  to have zero empirical mean; we call the resulting matrix  $\bar{\mathbf{F}}$ . We then compute an eigenvalue decomposition on the empirical covariance matrix of the centered channel features so that  $\bar{\mathbf{F}}^H \bar{\mathbf{F}} = \mathbf{U} \mathbf{\Sigma} \mathbf{U}^H$ . Here, the  $N \times N$  matrix  $\mathbf{U}$  is unitary, i.e.,  $\mathbf{U}^H \mathbf{U} = \mathbf{I}_N$ , and  $\mathbf{\Sigma}$  is a diagonal matrix with the  $N$  eigenvalues on the main diagonal sorted in descending order of their value (assuming all eigenvalues are real-valued), i.e.,  $\mathbf{\Sigma} = \text{diag}(\sigma_1, \dots, \sigma_N)$  so that  $\sigma_k \geq \sigma_\ell$  for  $1 \leq k < \ell \leq N$ . Finally, we compute the  $D' \times N$  matrix containing the low-dimensional points in the channel chart  $\mathbf{Z} = [\mathbf{z}_1, \dots, \mathbf{z}_N]$ . Let  $\mathbf{u}_d$  denote the  $d$ th column of  $\mathbf{U}$ . Then, the channel chart obtained via PCA is given by

$$\mathbf{Z}_{\text{PCA}} = [\sqrt{\sigma_1} \mathbf{u}_1, \dots, \sqrt{\sigma_{D'}} \mathbf{u}_{D'}]^H. \quad (4)$$

2) *Pros and Cons*: PCA is straightforward to implement and can be carried out in a computationally efficient manner using power iterations [50], [51]. However, as shown in Section VI, PCA performs worse in terms of TW and CT than the nonlinear CC methods proposed in the next two subsections.

### B. Sammon's Mapping

Sammon's mapping (SM) [31] is a classical nonlinear method that maps a high-dimensional point set into a point set of lower dimensionality with the goal of retaining small pairwise distances between both point sets—exactly what we wished for in Section II-B. We next describe SM for CC in detail,

explain an efficient algorithm to compute the channel chart, and propose a modified version that takes into account side information (called SM+ in what follows).

1) *SM Basics*: First, we compute a pairwise distance matrix  $\mathbf{D}$  of all channel features

$$D_{n,\ell} = d_f(\mathbf{F}_n, \mathbf{F}_\ell), \quad n = 1, \dots, N, \quad \ell = 1, \dots, N,$$

where we use the Frobenius distance (see Section IV-A). SM tries to find a low-dimensional channel chart, i.e., a point set  $\{\mathbf{z}_n\}_{n=1}^N$ , that results from the following optimization problem:

$$(SM) \quad \begin{cases} \text{minimize} & \sum_{\substack{n=2, \dots, N \\ \ell=1, \dots, n-1}} D_{n,\ell}^{-1} (D_{n,\ell} - \|\mathbf{z}_n - \mathbf{z}_\ell\|_2)^2 \\ \text{subject to} & \sum_{n=1, \dots, N} \mathbf{z}_n = \mathbf{0}_{D' \times 1}, \end{cases}$$

where we omit pairs of points for which  $D_{n,\ell} = 0$ . The objective function of SM promotes channel charts for which the Euclidean distance of pairs of nearby points in  $\mathbb{R}^{D'}$  agrees with the feature distance. Points for which  $D_{n,\ell}^{-1}$  is small (i.e., points that are dissimilar in feature geometry) are discounted; this ensures that SM retains small pairwise distances between both point sets. Since the objective function is invariant to global translations, we use a constraint that enforces the channel chart to be centered in each of the coordinates in  $\mathbb{R}^{D'}$ .

2) *Forward-Backward Splitting for SM*: The problem (SM) is non-convex and typically solved using quasi-Newton methods [52]. We next detail an efficient first-order method that enables us to include side information that is available for CC. We use an accelerated forward-backward splitting (FBS) procedure [53], [54] that solves a class of convex optimization problems of the following general form:

$$\text{minimize}_{\mathbf{Z} \in \mathbb{R}^{D' \times N}} f(\mathbf{Z}) + g(\mathbf{Z}),$$

where the function  $f(\mathbf{Z}) = \sum_{n=1}^K f_n(\mathbf{z}_n)$  should be convex and smooth and  $g$  should be convex, but must not be smooth or bounded. FBS mainly consists of the simple iteration

$$\mathbf{Z}^{(t+1)} = \text{prox}_g \left( \mathbf{Z}^{(t)} - \tau^{(t)} \nabla f(\mathbf{Z}^{(k)}), \tau^{(t)} \right)$$

for  $t = 1, \dots, T_{\text{max}}$  or until convergence. Here,  $\nabla f(\mathbf{Z})$  is the gradient of the smooth function  $f$ , and the proximal operator for the nonsmooth function  $g$  is defined as [55]

$$\text{prox}_g(\mathbf{Z}, \tau) = \arg \min_{\mathbf{V}} \left\{ \tau g(\mathbf{V}) + \frac{1}{2} \|\mathbf{V} - \mathbf{Z}\|_F^2 \right\}.$$

The sequence  $\{\tau^{(t)} > 0\}$  contains carefully selected step-size parameters that ensure convergence of FBS.

For CC, the matrix  $\mathbf{Z} = [\mathbf{z}_1, \dots, \mathbf{z}_N]$  contains all points in the channel chart. The function  $f$  is chosen to be

$$f(\mathbf{Z}) = \sum_{\substack{n=2, \dots, N \\ \ell=1, \dots, n-1}} D_{n,\ell}^{-1} (D_{n,\ell} - \|\mathbf{z}_n - \mathbf{z}_\ell\|_2)^2, \quad (5)$$

and the  $n$ th column of the gradient of  $f$  is

$$[\nabla f(\mathbf{Z})]_n = 2 \sum_{\substack{\ell=1, \dots, n \\ \ell \neq n}} D_{n,\ell}^{-1} (D_{n,\ell} - \|\mathbf{z}_n - \mathbf{z}_\ell\|_2) \frac{\mathbf{z}_n - \mathbf{z}_\ell}{\|\mathbf{z}_n - \mathbf{z}_\ell\|_2}.$$

The centering constraint in (SM) is enforced by choosing

$$g(\mathbf{Z}) = \chi \left( \sum_{n=1}^N \mathbf{z}_n \right),$$

where the ‘‘characteristic function’’  $\chi$  is zero when its argument  $\sum_{n=1}^N \mathbf{z}_n$  is zero, and infinity otherwise. The proximal operator of this characteristic function is simply a re-projection onto the centering constraint given by

$$\text{prox}_g(\mathbf{Z}, \tau) = \mathbf{Z} - \frac{1}{K} \mathbf{Z} \mathbf{1}_{N \times 1} \mathbf{1}_{N \times 1}^T.$$

**Remark 5.** *Since the function  $f$  is nonconvex, FBS is not guaranteed to find a global minimizer. We will demonstrate in Section VI that FBS with a suitable initialization and step-size criterion yields excellent CC results in a computationally efficient manner. Concretely, we initialize FBS with the solution from PCA  $\mathbf{Z}^{(1)} = \mathbf{Z}_{\text{PCA}}$  as detailed in Section V-A and we deploy the adaptive step-size procedure proposed in [54].*

3) *SM with Side-Information:* We now provide an example of how CC can be improved with side information. Note that the methods in this section remain unsupervised as they do not require information about the transmitter’s spatial locations.

In practice, one often collects many CSI vectors from a single transmitter (e.g., a UE). In this case, the channel features for a given transmitter  $u$  form a time series  $\{\mathbf{f}_n\}_{n \in \mathcal{N}_u}$ , where  $\mathcal{N}_u$  contains the temporally ordered channel feature indices associated with UE  $u$ . Since transmitters move with finite velocity, we know that temporally adjacent CSI vectors from the same UE should lie close together in the channel chart. To exploit this information, we include a squared  $\ell_2$ -norm penalty in the objective function that keeps temporally adjacent points in  $\mathcal{N}_u$  nearby in the channel chart. Concretely, for each transmitter  $u$ , we add

$$f_u(\mathbf{Z}) = \alpha_u \sum_{n \in \mathcal{N}_u} \|\mathbf{z}_n - \mathbf{z}_{n+1}\|_2^2$$

to the objective of (SM), where the parameter  $\alpha_u > 0$  determines the spatial smoothness of transmitter  $u$  in the channel chart. The  $n$ th row of the gradient of this penalty can be computed effectively and is given by

$$[\nabla f_u(\mathbf{Z})]_n = 2\alpha_u((\mathbf{z}_n - \mathbf{z}_{n+1}) + (\mathbf{z}_n - \mathbf{z}_{n-1}))$$

for  $n \in \mathcal{N}_u$ . In what follows, we refer to the resulting CC algorithm as Sammon’s mapping *plus* (SM+).

4) *Pros and Cons:* The main advantages of SM/SM+ are that (i) they directly implement the desirables for CC summarized in Section II-B, which results in excellent TW and CT (see Section VI for results), and (ii) temporal side information is easily included. The drawbacks are that (i) they are nonparametric, which would require an out-of-sample extension procedure as proposed in, e.g., [56], if new points need to be mapped without relearning the channel chart, and (ii) the complexity is substantially higher than that of PCA.

### C. Autoencoder

Autoencoders (AEs) [57] are single- or multi-layer (deep) artificial neural networks that are commonly used for unsupervised dimensionality reduction tasks [32] and have shown to yield excellent performance on numerous real-world datasets [58]. We now detail how AEs can be used for CC.

1) *Autoencoders for CC:* The basic idea of an AE is to learn two functions, an encoder  $\mathcal{C} : \mathbb{R}^{M'} \rightarrow \mathbb{R}^{D'}$  and a decoder  $\mathcal{C}^{-1} : \mathbb{R}^{D'} \rightarrow \mathbb{R}^{M'}$ , with  $M' > D'$ , so that the average approximation error

$$E = \frac{1}{N} \sum_{n=1}^N \|\mathbf{f}_n - \mathcal{C}^{-1}(\mathcal{C}(\mathbf{f}_n))\|_2^2 \quad (6)$$

for a set of vectors  $\{\mathbf{f}_n\}_{n=1}^N$  is minimal. Since the codomain (outputs) of the encoder  $\mathcal{C}$  is typically of lower dimension than the domain (inputs), we have that  $\mathbf{f}_n \approx \mathcal{C}^{-1}(\mathcal{C}(\mathbf{f}_n))$ , but this is not a perfect equality. The hope is that the AE implements a low dimensional representation  $\mathbf{z}_n = \mathcal{C}(\mathbf{f}_n)$  that captures the essential components of the inputs  $\mathbf{f}_n$ .

We now describe how AEs can be used for CC. First, it is important to realize that the encoder  $\mathcal{C}$  directly corresponds to the forward charting function with  $\mathbf{f}_n$  being the inputs; the decoder  $\mathcal{C}^{-1}$  corresponds to the inverse charting function. Second, we will use multi-layer (or deep) AEs [57] to learn the two functions  $\mathcal{C}$  and  $\mathcal{C}^{-1}$  in an unsupervised manner.

**Example 6.** *Consider a simple (shallow) AE whose encoder and decoder consist of a single layer, the inputs are the channel features, and the outputs of the decoder correspond to the points in the channel chart. Each layer first multiplies the inputs with a matrix (containing the weights) and adds a bias term; a (nonlinear) activation function (also known as neuron) is then applied element-wise to generate the outputs. Mathematically, such a shallow AE is described as follows:*

$$\mathcal{C} : \mathbf{z} = f_{\text{enc}}(\mathbf{W}_{\text{enc}} \mathbf{f} + \mathbf{b}_{\text{enc}}) \quad (7)$$

$$\mathcal{C}^{-1} : \hat{\mathbf{f}} = f_{\text{dec}}(\mathbf{W}_{\text{dec}} \mathbf{z} + \mathbf{b}_{\text{dec}}). \quad (8)$$

Here, the forward charting function  $\mathcal{C}$  (the encoder) first computes a matrix-vector product between the weight matrix  $\mathbf{W}_{\text{enc}} \in \mathbb{R}^{D' \times M'}$  and the vectorized channel feature  $\mathbf{f}$  (the inputs), followed by adding a bias vector  $\mathbf{b}_{\text{enc}} \in \mathbb{R}^{D'}$ . The result of this operation is then passed through a nonlinear activation function  $f_{\text{enc}}$  that operates element-wise on the entries of the argument. The inverse charting function  $\mathcal{C}^{-1}$  (the decoder) uses another weight matrix  $\mathbf{W}_{\text{dec}} \in \mathbb{R}^{M' \times D'}$ , bias vector  $\mathbf{b}_{\text{dec}} \in \mathbb{R}^{M'}$ , and activation function  $f_{\text{dec}}$  to map the input  $\mathbf{z} \in \mathbb{R}^{D'}$  to the channel feature geometry in  $\mathbb{R}^{M'}$ .

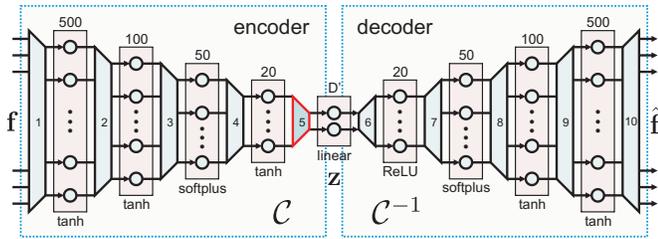


Fig. 5. Structure of the deep autoencoder used for CC. The entire artificial neural network consists of 10 layers; circles correspond to activation functions, trapezoids correspond to the weights and biases; the bottom text indicates the activation function type and the top text the output dimension of each layer.

In practice, one often resorts to multi-layer (or so-called deep) AEs [57] instead of the shallow network discussed in Example 6, as they often yield superior performance for many dimensionality-reduction tasks [32]. For such deep AEs, one simply cascades the inputs and outputs of multiple single-layer networks as in (7) and (8). The key design parameters of such deep AEs are the number of layers  $L$  (per encoder and decoder), the dimensions of the weight matrices and bias vectors on each layer, and the activation function types for each layer—all these parameters are fixed at design time. During the CC procedure, one jointly learns the entries of the weight matrices  $\{\mathbf{W}_{\text{enc}}^{(l)}, \mathbf{W}_{\text{dec}}^{(l)}\}$  and bias vectors  $\{\mathbf{b}_{\text{enc}}^{(l)}, \mathbf{b}_{\text{dec}}^{(l)}\}$ , where  $l = 1, \dots, L$  denotes the layer index, solely from the set of channel features  $\{\mathbf{f}_n\}_{n=1}^N$  so that the approximation error in (6) is minimal. Learning is typically accomplished by a procedure known as back-propagation [57], which is computationally efficient and scales favorably to large datasets.

2) *Implementation Details:* We use a deep AE as illustrated in Figure 5. We carefully selected the number of layers and their dimensionality, as well as the involved activation functions. The encoder and decoder both consist of  $L = 5$  layers.

The inputs of the encoder  $\mathcal{C}$  (the forward charting function) are the  $M'$ -dimensional channel features  $\{\mathbf{f}_n\}_{n=1}^N$ , the outputs correspond to points in the  $D'$  dimensional channel chart  $\{\mathbf{z}_n\}_{n=1}^N$ . For each layer  $l$ , the linear operation with the weights  $\mathbf{W}_{\text{enc}}^{(l)}$  and bias  $\mathbf{b}_{\text{enc}}^{(l)}$  are represented by the trapezoids in Figure 5. For the layers  $l = \{1, 2, 4\}$ , we set the activation to the hyperbolic tangent function  $f_{\text{enc}}^{(l)}(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$ . For layer three, we use the softplus function  $f_{\text{enc}}^{(3)}(x) = \log(1 + \exp(x))$ . For layer five, we use the identity  $f_{\text{enc}}^{(5)}(x) = x$ . The number of neurons for each layer are as follows:  $R^{(1)} = 500$ ,  $R^{(2)} = 100$ ,  $R^{(3)} = 50$ ,  $R^{(4)} = 20$ , and  $R^{(5)} = D'$ .

The inputs of the decoder  $\mathcal{C}^{-1}$  (the inverse charting function) are the points in the channel chart  $\{\mathbf{z}_n\}_{n=1}^N$  of dimension  $D'$ , and the outputs correspond to estimates of the  $M'$ -dimensional channel features  $\{\hat{\mathbf{f}}_n\}_{n=1}^N$ . As shown in Figure 5, the decoder is essentially a mirrored version of the encoder, having the same number of neurons per layer (but in reverse order). The only difference is the activation function on the sixth layer, where we use the rectified linear unit (ReLU) defined as  $f_{\text{dec}}^{(6)}(x) = \max\{x, 0\}$  instead of a hyperbolic tangent.

To reduce the approximation error of our AE and to obtain better TW and CT values, the weights in layer  $l = 5$  have been regularized. We include a squared Frobenius-norm regularizer

on the entries of  $\mathbf{W}_{\text{enc}}^{(5)}$  (also known as weight decay) by using the following average approximation error:

$$E = \frac{1}{2N} \sum_{n=1}^N \|\mathbf{f}_n - \mathcal{C}^{-1}(\mathcal{C}(\mathbf{f}_n))\|_2^2 + \frac{\beta}{2} \|\mathbf{W}_{\text{enc}}^{(5)}\|_F^2,$$

where the parameter  $\beta > 0$  was tuned for best performance. For learning of the AE, we use Tensorflow [59].

3) *Pros and Cons:* The key advantages of AE-based CC compared to PCA, SM, and SM+ are as follows: (i) AEs directly yield a parametric mapping of the forward and inverse channel charting function and (ii) they can be trained efficiently, even for very large datasets. The key drawback is the fact that identifying good network topologies, activation functions, and learning-rate parameters for AEs is notoriously difficult and involves tedious and time-consuming trial-and-error efforts by the user [60].

## VI. RESULTS

We are finally ready to provide results for CC for various channel models and the methods discussed above.

### A. Simulation Settings

Each channel chart shown next is generated for the system scenario depicted in Figure 1(a). We record CSI of  $N = 2048$  randomly placed (with the exception of the 234 points representing the “VIP” curve) spatial locations within a square area of  $1000 \text{ m} \times 500 \text{ m}$ ; the median sampling distance, measured in the spatial domain and between nearest neighbors, is approximately 53 wavelengths. We acquire CSI at an SNR of 0 dB, average noise over  $T = 10$  samples, and set  $\rho = 16$ . We compare results for a “vanilla” LoS channel (V-LoS) as in (1) at a carrier frequency of 2 GHz with  $\lambda/2$  antenna spacing, and for Quadriga LoS (Q-LoS) and Quadriga NLoS (Q-NLoS) channels (see Table I for the model parameters). Since the analysis in Section IV-D revealed that the feature configuration  $\{\text{R2M, Ant., } |\cdot|\}$  yields the most robust results with respect to CT and TW for all the above channel models (see Remark 4), we will generate channel charts solely for this channel feature. For each channel chart, we provide the global CT and TW values measured between spatial geometry and the channel chart for  $K = 0.05N$  nearest neighbors. In contrast to Figure 1, which has been tuned for visual appearance, the channel charts shown next are optimized for best TW and CT values.

### B. Channel Charts

Figure 6 shows learned channel charts for PCA, SM, SM+, and AE. For these CC algorithms and the three channel models, we obtain CT values between 0.91 and 0.94. This means that the neighborhood of a point in spatial geometry is strongly preserved in the channel charts, i.e., most points nearby in the spatial geometry space are nearby in the channel charts. The TW values are also high, ranging between 0.84 and 0.89; this indicates that most neighbors of a point in the channel charts are also neighbors in spatial geometry. We can also visually inspect the obtained results, e.g., by comparing the color gradient in Figure 6 with that of the scenario in Figure 1(a) or that of the

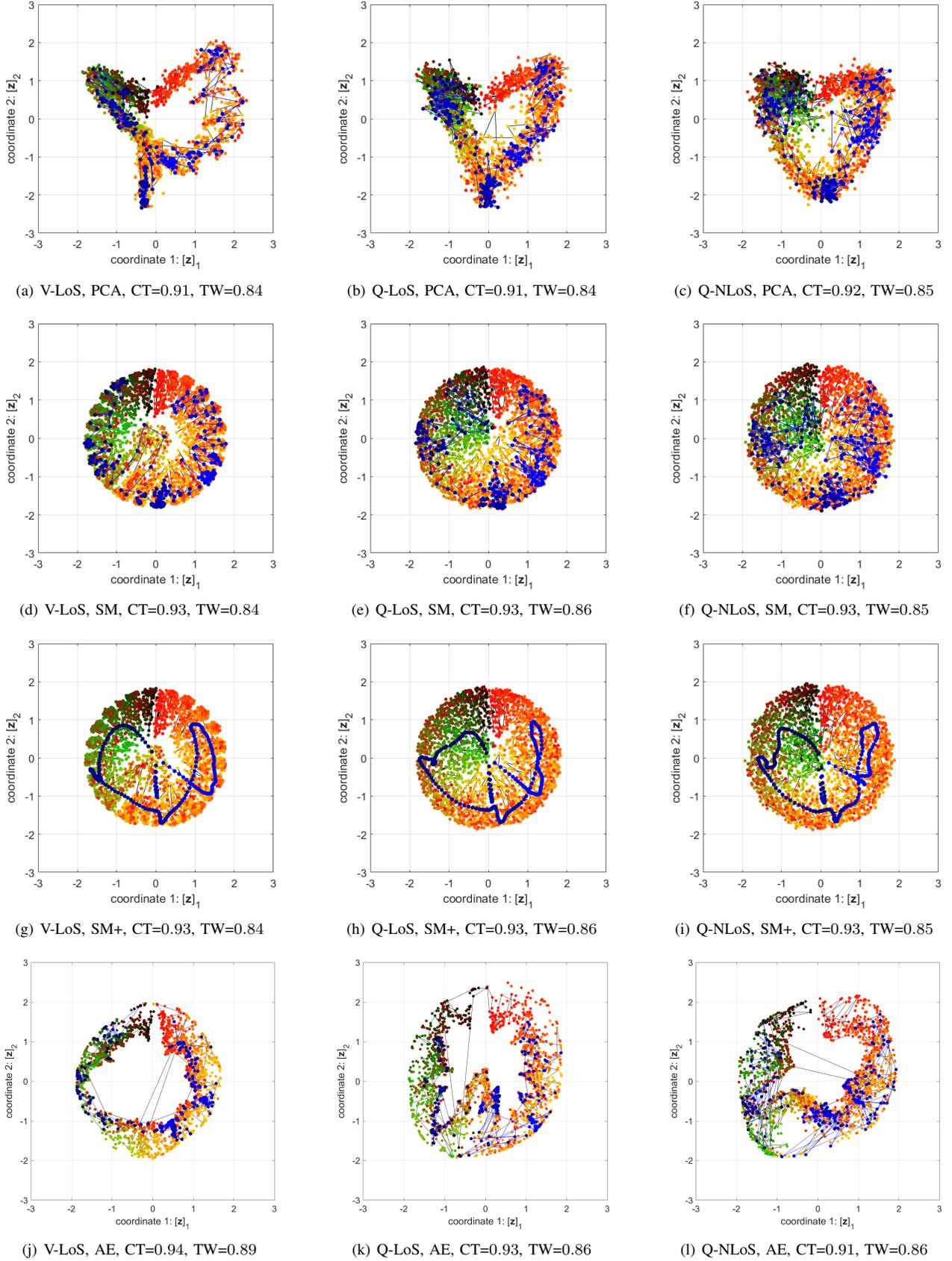


Fig. 6. Comparison of  $D' = 2$  dimensional channel charts for different channel models and CC algorithms. We compare “vanilla” LoS (V-LoS), Quadriga LoS (Q-LoS), and Quadriga non-LoS (Q-NLoS), with principal component analysis (PCA), Sammon’s mapping (SM), Sammon’s mapping with temporal continuity (SM+), and autoencoder (AE). We see that AE, SM, and SM+ achieve the highest CT and TW, whereas SM+ delivers the most visually pleasing results.

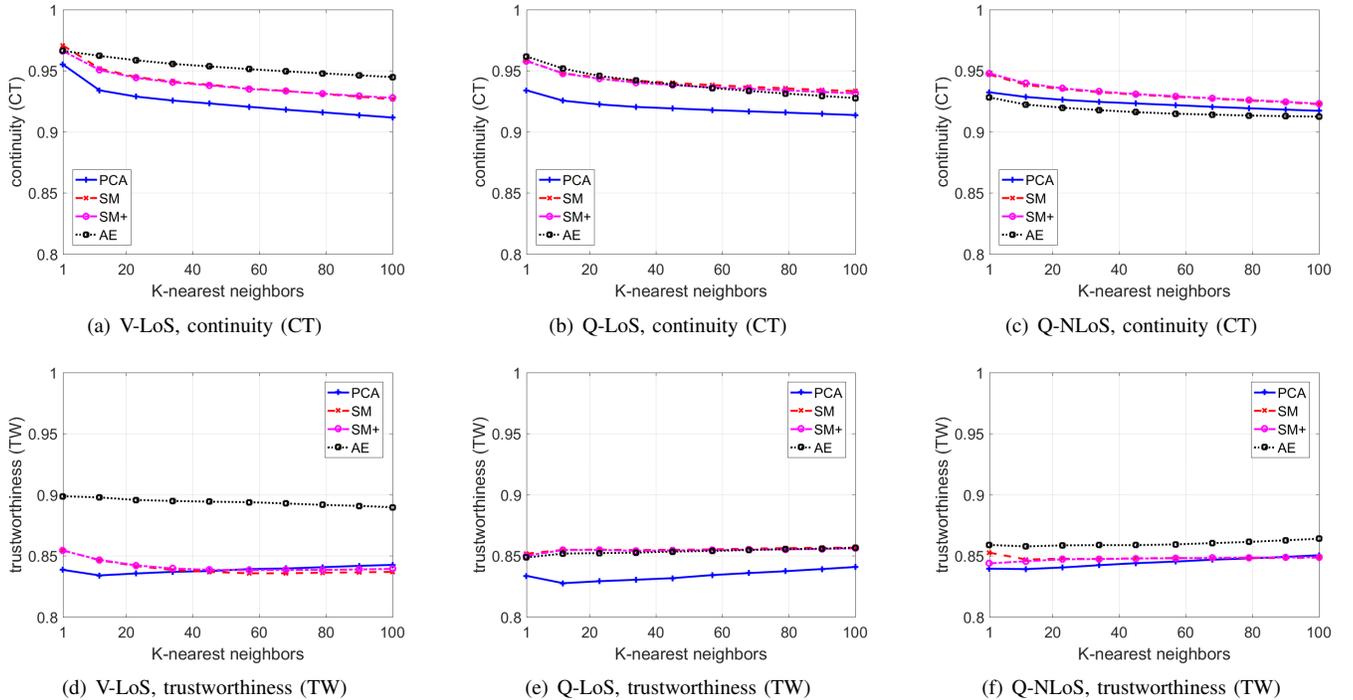


Fig. 7. Comparison of continuity (CT) and trustworthiness (TW) for various channel models and CC algorithms. We observe that autoencoders (AEs) outperform the other algorithms in terms of TW, while Sammon’s mapping (SM) and its extension (SM+) perform only slightly worse. In terms of CT, however, AEs only work well for simple LoS channels, whereas SM and SM+ perform better for channels generated from the Quadriga model. PCA yields surprisingly good results across the board and performs close to that of SM and SM+ in terms of CT for more challenging channel scenarios (Q-LoS and Q-NLoS).

“VIP” curve in spatial geometry and in the channel chart. To facilitate such a visual comparison, we have rotated and scaled all channel charts (note that this does not affect CT and TW).

The first row, Figures 6(a,b,c), shows the results for PCA. Quite surprisingly, PCA yields high CT and TW values for all channel models, and also provides a visually accurate embedding of the spatial geometry. This behavior is due to the fact that we use channel features that well-represent spatial geometry. The second row, Figures 6(d,e,f), shows the results for SM. SM yields superior CT and TW values than PCA and provides excellent preservation of the color gradients, especially for the two LoS scenarios. The third row, Figures 6(g,h,i), shows the results for SM+ in which we include spatial constraints obtained via temporal side-information. While the CT and TW values are nearly the same as that of SM, SM+ provides extremely well-preserved embeddings of the channel geometry, even for the challenging Q-NLoS scenario. The last row, Figures 6(j,k,l), shows the results for the AE. The AE yields high CT and TW values, comparable to those of SM/SM+, but slightly lower CT for Q-NLoS. In addition, the channel charts are less visually pleasing than those of SM+, but demonstrate excellent preservation of local spatial geometry.

### C. CT and TW Measures

To gain additional insight into the quality of the learned channel charts, Figure 7 shows the CT and TW values for different neighborhood sizes, i.e.,  $K$  ranges from 1 to 100. We see that, for the simplistic V-LoS channel, the AE provides the best performance, both in terms of CT and TW; SM and SM+

perform slightly worse, as does PCA. For the more realistic Q-LoS scenario that takes into account multi-path propagation, the performance of the AE drops significantly, while even PCA performs better. SM and SM+ have, once more, similar performance but perform better than the other two methods. For the most challenging scenario, the Quadriga non-LoS channel (Q-NLoS), SM and SM+ perform best, followed by PCA. Evidently, the AE struggles in achieving high CT. We address this issue to the fact that we train the AE only on the  $N = 2048$  points and the fact that we could spend another week in tuning the neural net architecture and learning rates.

## VII. CONCLUSIONS

We have proposed *channel charting* (CC), a novel unsupervised framework to learn a map between channel-state information (CSI) acquired at a single base-station (BS) and the relative transmitter (e.g., user equipment) locations. Our method relies on the extraction of suitable features from large amounts of high-dimensional CSI acquired at a massive MIMO BS, followed by CC algorithms that borrow ideas from dimensionality reduction and manifold learning. We have developed four distinct CC algorithms with varying complexity, flexibility, and accuracy that produce charts that preserve the local geometry of the transmitter locations for a range of realistic channel models. Since channel charting is unsupervised, i.e., does not require knowledge of the true user locations, the proposed method finds use in numerous applications relevant to 5G networks, including (but not limited to) rate adaptation, network planning, user scheduling, hand-

over, cell search, user tracking, user grouping for device-to-device (D2D) communication, beam prediction for mmWave or terahertz systems, and other cognitive tasks that rely on CSI and the relative UE movement to the BS.

There are many avenues for future work. A mathematical analysis of the proposed feature extraction and CC algorithm stages that provides insight into what aspects are relevant for the learning of accurate channel charts is a challenging open research question. Improved channel features that are particularly resilient to shadowing and more advanced CC algorithms, such as methods relying on metric learning or convolutional neural networks that take into account side information, have the potential to yield even better continuity and trustworthiness. Finally, an extension to semi-supervised methods, time-varying channels, and multi-user scenarios is part of ongoing work.

## REFERENCES

- [1] S. Medjkouh, E. Gönültaş, T. Goldstein, O. Tirkkonen, and C. Studer, "Unsupervised charting of wireless channels," in *IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018.
- [2] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. Soong, and J. Zhang, "What will 5G be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.
- [3] A. Osseiran, J. Monserrat, and P. Marsch, eds., *5G Mobile and Wireless Communications Technology*. Cambridge Univ. Press, 2016.
- [4] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Trans. Wireless Commun.*, vol. 9, no. 11, pp. 3590–3600, Nov. 2010.
- [5] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, O. Edfors, and F. Tufvesson, "Scaling up MIMO: Opportunities and challenges with very large arrays," *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 40–60, Jan. 2013.
- [6] E. Larsson, O. Edfors, F. Tufvesson, and T. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, Feb. 2014.
- [7] T. S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. N. Wong, J. K. Schulz, M. Samimi, and F. Gutierrez, "Millimeter wave mobile communications for 5G cellular: It will work!" *IEEE Access*, vol. 1, pp. 335–349, May 2013.
- [8] X. Ge, S. Tu, G. Mao, C. X. Wang, and T. Han, "5G ultra-dense cellular networks," *IEEE Wireless Commun.*, vol. 23, no. 1, pp. 72–79, Feb. 2016.
- [9] A. Prasad, O. Tirkkonen, P. Lunden, O. N. C. Yilmaz, L. Dalsgaard, and C. Wijting, "Energy-efficient inter-frequency small cell discovery techniques for LTE-advanced heterogeneous network deployments," *IEEE Commun. Mag.*, vol. 51, no. 5, pp. 72–81, May 2013.
- [10] H. Zhao, R. Mayzus, S. Sun, M. Samimi, J. K. Schulz, Y. Azar, K. Wang, G. N. Wong, F. Gutierrez, and T. S. Rappaport, "28 GHz millimeter wave cellular communication measurements for reflection and penetration loss in and around buildings in New York City," in *IEEE Int. Conf. on Commun. (ICC)*, Jun. 2013, pp. 5163–5167.
- [11] M. R. Akdeniz, Y. Liu, M. K. Samimi, S. Sun, S. Rangan, T. S. Rappaport, and E. Erkip, "Millimeter wave channel modeling and cellular capacity evaluation," *IEEE J. on Sel. Areas Commun.*, vol. 32, no. 6, pp. 1164–1179, Jun. 2014.
- [12] F. Ahmed, J. Deng, and O. Tirkkonen, "Self-organizing networks for 5G: Directional cell search in mmW networks," in *IEEE Int. Symp. Personal, Indoor, and Mobile Radio Commun. (PIMRC)*, Sep. 2016, pp. 1–5.
- [13] K. Chandra, R. V. Prasad, B. Quang, and I. G. M. M. Niemegeers, "Cogcell: cognitive interplay between 60 GHz picocells and 2.4/5 GHz hotspots in the 5G era," *IEEE Commun. Mag.*, vol. 53, no. 7, pp. 118–125, Jul. 2015.
- [14] J. Deng, O. Tirkkonen, R. Freij-Hollanti, T. Chen, and N. Nikaein, "Resource allocation and interference management for opportunistic relaying in integrated mmWave/sub-6 GHz 5G networks," *IEEE Commun. Mag.*, vol. 55, no. 6, pp. 94–101, Jun. 2017.
- [15] F. Gustafsson and F. Gunnarsson, "Mobile positioning using wireless networks," *IEEE Signal Process. Mag.*, pp. 41–53, July 2005.
- [16] S. Kumar, S. Gil, D. Katabi, and D. Rus, "Accurate indoor localization with zero start-up cost," in *Proc. 20th Annu. Int. Conf. Mobile Comput.*, Sep. 2014, pp. 483–494.
- [17] N. Garcia, H. Wymeersch, E. G. Larsson, A. M. Haimovich, and M. Coulon, "Direct localization for massive MIMO," *IEEE Trans. Sig. Proc.*, vol. 65, no. 10, pp. 2475–2487, July 2017.
- [18] M. Ibrahim and M. Youssef, "Cellsense: An accurate energy-efficient GSM positioning system," *IEEE Trans. Veh. Technol.*, vol. 61, no. 1, pp. 286–296, Jan. 2012.
- [19] Y. Chapre, A. Ignjatovic, A. Seneviratne, and S. Jha, "CSI-MIMO: An efficient Wi-Fi fingerprinting using channel state information with MIMO," *Pervasive and Mobile Comput.*, vol. 23, pp. 89–103, Oct. 2015.
- [20] X. Wang, L. Gao, S. Mao, and S. Pandey, "DeepFi: Deep learning for indoor fingerprinting using channel state information," in *IEEE Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2015, pp. 1666–1671.
- [21] L. Gao, "Channel state information fingerprinting based indoor localization: a deep learning approach," Auburn University M.S. Thesis, Aug. 2015.
- [22] X. Wang, L. Gao, and S. Mao, "CSI phase fingerprinting for indoor localization with a deep learning approach," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1113–1123, Dec. 2016.
- [23] V. Savic and E. G. Larsson, "Fingerprinting-based positioning in distributed massive MIMO systems," in *IEEE Veh. Technol. Conf.*, Sep. 2015, pp. 1–5.
- [24] J. Vieira, E. Leitinger, M. Sarajlic, X. Li, and F. Tufvesson, "Deep convolutional neural networks for massive MIMO fingerprint-based positioning," in *IEEE 28th Annu. Int. Symp. Personal, Indoor, and Mobile Radio Commun. (PIMRC)*, Oct. 2017.
- [25] E. Dahlman, S. Parkvall, and J. Sköld, *4G: LTE/LTE-Advanced for Mobile Broadband*. Elsevier, Mar. 2011.
- [26] S. Pradhan, "Semantic location," *Personal and Ubiquitous Computing*, vol. 4, no. 4, pp. 213–216, Aug. 2000.
- [27] J. G. Andrews, "Seven ways that HetNets are a cellular paradigm shift," *IEEE Commun. Mag.*, vol. 51, no. 3, pp. 136–144, Mar. 2013.
- [28] Y. Kishiyama, A. Benjebbour, T. Nakamura, and H. Ishii, "Future steps of LTE-A: evolution toward integration of local area and wide area systems," *IEEE Wireless Commun.*, vol. 20, no. 1, pp. 12–18, Feb. 2013.
- [29] F. B. Tesema, A. Awada, I. Viering, M. Simsek, and G. Fettweis, "Evaluation of context-aware mobility robustness optimization and multi-connectivity in intra-frequency 5G ultra dense networks," *IEEE Wireless Commun. Lett.*, vol. 5, no. 6, pp. 608–611, Dec. 2016.
- [30] J. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, vol. 29, no. 1, pp. 1–27, Mar. 1964.
- [31] J. W. Sammon, "A nonlinear mapping for data structure analysis," *IEEE Trans. Comput.*, vol. 100, no. 5, pp. 401–409, May 1969.
- [32] L. Van Der Maaten, E. Postma, and J. Van den Herik, "Dimensionality reduction: a comparative review," Tilburg University, TiCC-TR 2009-005, 2009.
- [33] A. A. Goshtasby, "Similarity and dissimilarity measures," in *Image registration*. Springer, 2012, pp. 7–66.
- [34] T. S. Rappaport, *Wireless Communications: Principles and Practice*. Prentice Hall, 1996.
- [35] A. Goldsmith, *Wireless Communications*. Cambridge Univ. Press, 2005.
- [36] W. C. Y. Lee, "Effects on correlation between two mobile radio base-station antennas," *IEEE Trans. Vehic. Technol.*, vol. 22, no. 4, pp. 130–140, Nov. 1973.
- [37] D.-S. Shiu, G. J. Foschini, M. J. Gans, and J. M. Kahn, "Fading correlation and its effect on the capacity of multielement antenna systems," *IEEE Trans. Commun.*, vol. 48, no. 3, pp. 502–513, Mar. 2000.
- [38] A. Lozano, R. W. Heath, and J. G. Andrews, "Fundamental limits of cooperation," *IEEE Trans. Inf. Theory*, vol. 59, no. 9, pp. 5213–5226, Sep. 2013.
- [39] K. Gomadam, V. R. Cadambe, and S. A. Jafar, "A distributed numerical approach to interference alignment and applications to wireless interference networks," *IEEE Trans. Inf. Theory*, vol. 57, no. 6, pp. 3309–3322, Jun. 2011.
- [40] M. Koivisto, A. Hakkarainen, M. Costa, P. Kela, K. Leppanen, and M. Valkama, "High-efficiency device positioning and location-aware communications in dense 5G networks," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 188–195, July 2017.
- [41] 3GPP, "Evolved universal terrestrial radio access (E-UTRA); physical layer procedures (release 10)," Apr. 2011, TS 36.213 ver. 10.1.0 Rel.10.
- [42] J. Venna and S. Kaski, "Neighborhood preservation in nonlinear projection methods: An experimental study," in *Int. Conf. on Artificial Neural Networks*. Springer, 2001, pp. 485–491.
- [43] S. Kaski, J. Nikkilä, M. Oja, J. Venna, P. Törönen, and E. Castrén, "Trustworthiness and metrics in visualizing similarity of gene expression," *BMC Bioinformatics*, vol. 4, no. 1, p. 48, Oct. 2003.

- [44] Á. Vathy-Fogarassy and J. Abonyi, *Graph-based clustering and data visualization algorithms*. Springer, 2013.
- [45] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge Univ. Press, 2005.
- [46] J. Brady, N. Behdad, and A. M. Sayeed, “Beamspace MIMO for millimeter-wave communications: System architecture, modeling, analysis, and measurements,” *IEEE Trans. Antennas Propag.*, vol. 61, no. 7, pp. 3814–3827, July 2013.
- [47] 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, Mar. 2014.
- [48] K. Pearson, “On lines and planes of closest fit to systems of points in space,” *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, 1901.
- [49] H. Hotelling, “Analysis of a complex of statistical variables into principal components,” *J. of Edu. Psychology*, vol. 24, no. 6, p. 417, 1933.
- [50] R. Mises and H. Pollaczek-Geiringer, “Praktische Verfahren der Gleichungsauflösung,” *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 9, no. 1, pp. 58–77, 1929.
- [51] G. H. Golub and C. F. van Loan, *Matrix Computations*, 3rd ed. The Johns Hopkins Univ. Press, 1996.
- [52] J. Nocedal and S. J. Wright, *Numerical optimization*, 2nd ed. Springer, 2006.
- [53] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm for linear inverse problems,” *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, Jan. 2009.
- [54] T. Goldstein, C. Studer, and R. G. Baraniuk, “A field guide to forward-backward splitting with a FASTA implementation,” *arXiv preprint: 1411.3406*, Nov. 2014. [Online]. Available: <http://arxiv.org/abs/1411.3406>
- [55] N. Parikh and S. Boyd, “Proximal algorithms,” *Foundations and Trends Optimization*, vol. 1, no. 3, pp. 123–231, Jan. 2013.
- [56] H. Li, L. Teng, W. Chen, and I.-F. Shen, “Supervised learning on local tangent space,” *Lect. Notes Comp. Science*, no. 3496, pp. 546–551, 2005.
- [57] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT Press, 2016.
- [58] W. Wang, Y. Huang, Y. Wang, and L. Wang, “Generalized autoencoder: A neural network framework for dimensionality reduction,” in *2014 IEEE Conf. on Comput. Vision and Pattern Recognition*, Jun. 2014, pp. 496–503.
- [59] Abadi, Martín et al., “TensorFlow: Large-scale machine learning on heterogeneous systems,” 2015, software available from tensorflow.org. [Online]. Available: <https://www.tensorflow.org/>
- [60] D. Hunter, H. Yu, M. S. Pukish III, J. Kolbusz, and B. M. Wilamowski, “Selection of proper neural network sizes and architectures — A comparative study,” *IEEE Trans. Ind. Informat.*, vol. 8, no. 2, pp. 228–240, May 2012.