A Kernel-Power-Density Based Algorithm for Channel Multipath Components Clustering

Ruisi He, *Member, IEEE*, Qingyong Li, *Member, IEEE*, Bo Ai, *Senior Member, IEEE*, Yang Li-Ao Geng, Andreas F. Molisch, *Fellow, IEEE*, Vinod Kristem, Zhangdui Zhong, *Senior Member, IEEE*, and Jian Yu

***Abstract*—Cluster based channel modeling has been an important trend in the development of channel model, as it has maintains accuracy while reducing complexity. Whereas a large number of channel measurements have shown that multipath components (MPCs) are distributed as groups, i.e., clusters, existing clustering algorithms have various drawbacks with respect to complexity, threshold choices, and/or assumptions about prior knowledge. In this paper, a Kernel-power-density (KPD) based algorithm is proposed for MPC clustering. It uses the Kernel density of MPCs to incorporate the modeled behavior of MPCs and takes into account the power of the MPCs. Furthermore, the KPD algorithm only considers the K nearest MPCs in the density estimation to better identify the local density variations of MPCs. A heuristic approach of cluster merging is used to improve the performance. Both simulation and channel measurements validate the KPD algorithm, and almost no performance degradation is found even with a large number of clusters and large cluster angular spread, which outperforming other algorithms. The KPD algorithm enables applications in multiple-input-multiple-output (MIMO) channels with no prior knowledge about the clusters, such as number and initial locations. It also has a fairly low computational complexity and can be used for cluster based channel modeling.**

***Index Terms*—Channel measurement and modeling, clustering analysis, Kernel density, machine learning, multipath component, wireless channel.**

I. Introduction

CHANNEL modeling has been an important research topic in wireless communications, as the design and performance evaluation of any wireless communication system is based on an accurate channel model. The main goal of channel modeling is to characterize the multipath components (MPCs) in different environments, with a consideration of the tradeoff between model accuracy and complexity.

To statistically model MPCs, there generally exists two modeling methodologies - non-clustered and clustered structure modelings. The non-clustered structure model characterizes the channel at a level of individual MPCs. A widely used model is the tapped delay line (TDL) model [1], which includes a number of taps at different delays and can be further extended to include angular domain. Each tap represents the superposition of a large number of MPCs and experiences small-scale fading and a multi-dimensional Poisson process can be used to characterize each tap. The non-clustered structure channel model has been used for a long time [2]–[4] and accepted by channel models such as the COST 207 model [5] and IEEE 802.11p channel model [6]. Another modeling methodology is the clustered structure modeling, where MPCs are grouped into clusters, and both the intra- and inter-cluster statistics are characterized for the parameters such as number, position, and delay and angular spreads. The use of clustered structures is mainly motivated by the fact that 3G, 4G, and next-generation systems have larger bandwidth as well as multiple-input-multiple-output (MIMO) arrays are increasing. With the high resolution of MPC on both delay and angle domains, a large body of MIMO measurements has shown that the MPCs are generally distributed in groups, i.e., clustered, in the real-world environments, see, e.g., [7] and references therein. Therefore, the clustered structure channel models are used to reflect this condition.

Compared with the non-clustered structure modeling, the channel models with clustered structure generally has the following advantages: i) the clustered structure fits well to measurement data in some environments and is widely adopted in some standards (as reported later); ii) clustering of MPCs helps to reduce the number of MIMO model parameters [8] and the cluster structure is more flexible when characterizing multi-link scenarios, e.g., by using the so-called twin clusters and common clusters [9] and by using the cluster visibility regions to model time-variant channels [10]; and iii) the clustered structure allows to separate intra-cluster and inter-cluster statistics. The intra-cluster statistics can usually be described very compactly, whereas the inter-cluster statistics might take on a more complicated form, but that is acceptable if the number of clusters is small, so that the number of parameters

Part of the work is presented at IEEE WCNC 2017. This work was supported by the National key research and development program under Grant 2016YFB1200102-04, the National Natural Science Foundation of China under Grant 61501020 and U1334202, the State Key Laboratory of Rail Traffic Control and Safety under Grant RCS2016ZJ005, China Postdoctoral Science Foundation under Grant 2016M591355, the National S&T Major Project 2016ZX03001021-003, the Fundamental Research Funds for the Central Universities under Grant 2016JBZ006, the Beijing Natural Science Foundation under Grant J160004, and the Shanghai Research Program under Grant 17511102900. The work of Andreas F. Molisch was supported by the National Science Foundation. (*Corresponding author: Ruisi He*.)

R. He, B. Ai, and Z. Zhong are with the State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, China. R. He is also with Beijing Key Laboratory of Traffic Data Analysis and Mining, Beijing Jiaotong University, Beijing 100044, China (e-mail: ruisi.he@bjtu.edu.cn, boai@bjtu.edu.cn, zhdzhong@bjtu.edu.cn).

Q. Li and Y. L. A. Geng are with the School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China (e-mail: liqy@bjtu.edu.cn, 15120392@bjtu.edu.cn).

A. F. Molisch and Vinod Kristem are with the Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089, USA (e-mail: molisch@usc.edu, kristem@usc.edu).

J. Yu is with the Beijing Key Lab of Traffic Data Analysis and Mining, Beijing Jiaotong University, Beijing 100044, China (e-mail: jianyu@bjtu.edu.cn).

is also inherently limited. However, it is noteworthy that the cluster based channel modeling may lead to reduced accuracy, as it neglects the impacts of some MPCs, especially those are far from the cluster center. Moreover, it requires extra efforts of MPC clustering and also introduces extra modeling errors due to MPC estimation and clustering inaccuracies [11]. The above problems can be mitigated by improving estimation and clustering algorithms. Therefore, a good clustering algorithm is required to enhance the cluster based channel modeling.

To our knowledge, the phenomenon of clustered MPCs has been observed for many years, as in [12]. In [13], a cluster-based channel model was proposed, where the MPCs are clustered in the delay domain based on measurements. In [14], a geometry-based stochastic channel model (GSCM) suitable for MIMO channels was introduced, where the concept of MPC cluster was extended to include both delay and angular domains. Over the past 20 years, the clustering of MPCs have been widely observed in many environments [15]–[17] and cluster based channel modeling has been widely adopted in many channel models, such as COST 259 [18], COST 2100 [10], 3GPP Spatial Channel Model [19], and WINNER [20] [1].

To develop the cluster based channel model, MPC clustering is the first step so that the clusters' number, positions, and delay and angular spreads can be parameterized. However, finding good clustering algorithms is very much an open and research-active topic. In the past, visual inspection [21] has been used to cluster MPCs for a long time. However, it has some limitations as follows: i) visual inspection tends to detect patterns even in completely random data, which usually leads to misleading results; ii) the objective clustering criteria is generally missing and the results depend on the operator; iii) it is too time-consuming for the clustering implementation with a large amount of data; and iv) it is difficult for the human eye to cluster high-dimensional data. Therefore, a carefully designed automatic clustering algorithm is required for channel modeling.

A. Related Work

Even though clustering analysis is a hot research topic in the field of machine learning [22], considerable effort has to be made to adapt the results to clustering of MPCs in wireless channels. Since the MPC has many attributes such as power, delay, angle, and each of the above attributes usually has an independent characteristic (i.e., MPCs exhibit different statistical characteristics in each domain), the main challenge of MPC clustering is how to incorporate the impacts of different attributes. Several algorithms are proposed to cluster MPCs when only the power and delay attributes are available. In [23], MPCs are clustered with the help of region competition algorithm [24] and the amplitude distribution of MPCs is incorporated into the algorithm by using the Kurtosis measure. In [25], a series of exponential curves is fitted to the measurements so that the root-mean-squared-error (RMSE) is minimized. In [26], a sparsity-based method is proposed to cluster MPCs, which exploits the feature that the power of the MPCs is exponentially decreasing with increasing delay. However, the above algorithms are only suitable when only the delay information of the MPCs is available.

Several algorithms are proposed to cluster MPCs when all the attributes (power, delay, angles) are considered. In [27], the K-Power-Means (KPM) algorithm is proposed, which is similar to the KMeans algorithm [28]. It considers the impact of MPC power in computing the cluster centers and uses MPC distance (MCD) [29] to quantify the similarity between MPCs. KPM has been widely used in the analysis of double-directional (MIMO) channels. In [30], the Fuzzy-c-means algorithm is used to cluster MPC and is found to outperform the KPM when using random initialization. In [16], the density-based spatial clustering for applications with noise (DBSCAN) algorithm [31] is applied to cluster local MPCs. In [32], a fixed inter-cluster void interval, which represents the minimum propagation time between likely reflection or scattering objects, is used to distinguish clusters on time domain. In [33], a hierarchical agglomerative clustering algorithm is used to search for clusters jointly in the delay-angle-space domain and the performance is validated by ray-tracing simulation.

B. Motivation

Despite the impressive progress made in automated clustering over the past 10 years, the existing works have several limitations:

- The attributes of MPCs are not well incorporated into the clustering algorithm. Unlike the synthetic samples in machine learning, the attributes of real-world MPCs are caused by the physical environments and thus have certain inherent characteristics. Such anticipated behaviors of MPCs should be incorporated into the clustering algorithm. For example, many measurements show that the angle distribution of MPC clusters can be usually modeled as a Laplacian distribution [34], however, this characteristic has not been well considered in the design of clustering algorithm.
- The number of clusters is usually required as prior information. Even though in [35] several validity indices are compared to select the best estimation of the number of clusters, it is found that none of the indices is able to always predict correctly the desired number of clusters. The final decision of clustering is usually derived by applying a fusion method to various cluster validation indices [36], [37]. Mostly, people still need to use visual inspection to ascertain the optimum number of clusters in the environment [38].
- Most clustering algorithms still require many user-specified parameters. For example, the KPM algorithm requires the cluster initialization (delay and angle), and usually the weight factors of delay and angle need to be adjusted to obtain a reasonable output, which is subjective; in DBSCAN, the neighborhood radius and the minimum number of points to form a dense region are required. As each of the parameters influences the algorithm in specific way, it is difficult to find a good

[1]Note, however, that 3GPP and WINNER use a somewhat different definition for clusters than the remainder of the literature.

initialization in real-world measurements. In general, an algorithm with fewer user-specified parameters and easier adjustment is needed for MPC clustering.

In our previous work of [39], we briefly introduce the idea of using density based method to cluster MPCs. In this paper, we extend our previous work and propose a novel clustering framework by using a density based method. We define a Kernel density to incorporate the modeled behavior of MPCs, and a power term is also included. Only the K nearest MPCs are considered in the density estimation to better identify the local density variations of MPCs. The proposed algorithm can be used in MIMO channels and requires no prior knowledge of the number of clusters. Both simulations and measurements are used in the validation and it is found that the proposed algorithm outperforms other algorithms. It is noteworthy that a complete algorithm validation includes: i) the clustering performance validation, which shows how well the MPCs are clustered; and ii) validation of the resulting cluster-based model accuracy, which shows that the recovered model from the improved clustering results leads to a lower modeling error. As the latter case is highly dependent on the channel modeling method and model error analysis, it involves selections of different modeling approaches and measurement data, and cannot be easily addressed in full generality; it is therefore considered to be outside the scope of this paper. Therefore, in this paper, the discussions are limited to the former case, i.e., the validation is conducted only on the level of MPC clustering performance. The investigation of the impact of clustering algorithm improvement on channel modeling accuracy could be a topic for future work.

C. Outline

This paper is organized as follows. Section II describes the radio channel model and the corresponding parameters used in the clustering. Section III shows the detailed framework of the proposed clustering algorithm and pseudocode is provided as well. Section IV presents some insights for the proposed algorithm. Section V validates the algorithm using both simulations and measurements. Finally, Section VI concludes the paper.

II. Channel Description

In any wireless channel, the signal can get from the transmitter (TX) to the receiver (RX) via a number of different paths [1]. MIMO channels can be modeled as double-directional [40], and are characterized by the double-directional impulse response, which contains the information of power α, delay τ, azimuth of departure (AOD) Ω_T, azimuth of arrival (AOA) Ω_R, elevation of departure (EOD) Θ_T, and elevation of arrival (EOA) Θ_R of the MPCs. As mentioned in Section I, MPCs tend to appear in clusters, i.e., the MPCs in each cluster have similar parameters of power, delay, and angle. For each snapshot, the double-directional channel impulse response h can thus be expressed as follows:

$$\begin{aligned} &h(t, \tau, \Omega_T, \Omega_R, \Theta_T, \Theta_R) = \\ &\quad \sum_{m=1}^{M} \{ \sum_{n=1}^{N_m} \sqrt{\alpha_{m,n}} e^{j\phi_{m,n}} \delta(\tau - \tau_m - \tau_{m,n}) \\ &\quad \times \delta(\Omega_T - \Omega_{T,m} - \Omega_{T,m,n}) \times \delta(\Omega_R - \Omega_{R,m} - \Omega_{R,m,n}) \\ &\quad \times \delta(\Theta_T - \Theta_{T,m} - \Theta_{T,m,n}) \times \delta(\Theta_R - \Theta_{R,m} - \Theta_{R,m,n}) \} \end{aligned} \tag{1}$$

where M is the number of cluster and N_m is the number of MPCs in the m-th cluster. $\alpha_{m,n}$ and $\phi_{m,n}$ are the power and phase of the n-th MPC in the m-th cluster, respectively. τ_m, $\Omega_{T,m}$, $\Omega_{R,m}$, $\Theta_{T,m}$, and $\Theta_{R,m}$ are the arrival time, AOD, AOA, EOD, and EOA of the m-th cluster, respectively. $\tau_{m,n}$, $\Omega_{T,m,n}$, $\Omega_{R,m,n}$, $\Theta_{T,m,n}$, and $\Theta_{R,m,n}$ are the excess delay, excess AOD, excess AOA, excess EOD, and excess EOA, of the n-th MPC in the m-th cluster, respectively, where excess delay is usually taken with respect to the first component in the cluster, while excess angles are taken with respect to the mean. $\delta(\cdot)$ is the Dirac delta function and t is time.

All the multipath parameters in (1) can be estimated by using high-resolution algorithm, such as MUSIC [41], CLEAN [42], SAGE [43], or RiMAX [44]. Therefore, the clustering analysis can be applied to the extracted MPCs. As noted in (1), we consider one data snapshot with T MPCs and the total number of clusters is M. Each extracted MPC is represented by a vector $\boldsymbol{x}$ containing its power α, delay τ, AOD Ω_T, AOA Ω_R, EOD Θ_T, and EOA Θ_R. The set of all the MPCs for one snapshot is Φ and the complete data set is denoted by $\Phi = \{x_1, \cdots, x_T\}$. In the following, we propose a clustering algorithm for MPC using the above parameters.

III. Clustering Algorithm

In this section, we first briefly summarize the KPM and DBSCAN algorithms, which are widely used in MPC clustering and will be compared later in this paper. Then we propose a Kernel-power-density (KPD) based algorithm for MPC clustering, which is found to outperform other algorithms.

A. KPM

The KPM algorithm [27] is based on the KMeans algorithm [28] and incorporates the impact of MPC powers. Fig. 1(a) briefly shows the idea of KPM and Kmeans. The main steps are as follows:

1) Initialize M cluster centroids $\mu_1, \mu_2, \ldots, \mu_M$ randomly, i.e., the M centroid positions are independently chosen as events of equal probability (without replacement) from the data set Φ.
2) Assign each MPC sample x to the reasonable cluster centroid μ_j: for each x, set

$$c^{(e)} := \arg\min_j \left\{ \alpha_x \cdot d_{\text{MPC}}(x, \mu_j^{(e)}) \right\} \tag{2}$$

where superscript (e) represents the e-th iteration. c represents the store indices of MPC clustering in the e-th iteration. d_{MPC} is the MCD defined in [29].

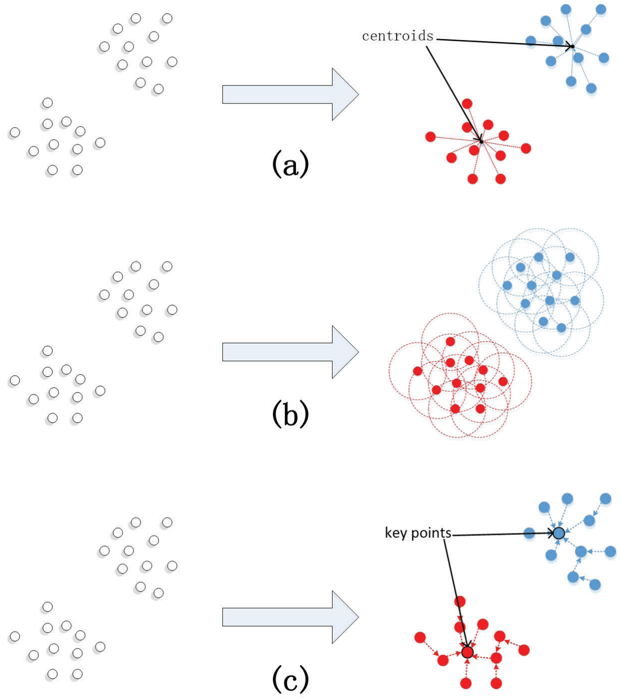

Fig. 1. Main idea illustrations of three clustering algorithms. The white dots indicate the raw MPCs, and the red and blue dots are the clustering results. (a) KPM. (b) DBSCAN. (c) KPD.

3) Update the cluster centroids: for each j, set

$$\mu_j^{(e+1)} := \frac{\sum\limits_{x\in\Phi} 1\left\{c^{(e)}=j\right\}\alpha_x \cdot x}{\sum\limits_{x\in\Phi} 1\left\{c^{(e)}=j\right\}\alpha_x} \tag{3}$$

4) Repeat steps 2 and 3 until convergence.

The KPM clustering is an unsupervised learning algorithm, and the initialization of clusters has to be known a priori. It is also found that the weight factors of delay and angle domains in d_{MPC} significantly affect the clustering results. Therefore, manual adjustments of algorithm parameters according to different data are usually required to improve the performance, which makes the KPM somehow subjective.

B. DBSCAN

The DBSCAN [31] is a density-based clustering algorithm, which is able to find arbitrarily shaped clusters and does not require to specify the number of clusters a priori. It is based on the notion of "density-reachable" and requires two parameters: neighborhood radius ε and threshold $minPts$ of the minimum number of points required to form a cluster. In DBSCAN, a point x is defined as a core point if at least $minPts$ points are within neighborhood radius ε of it, and those points are said to be directly reachable from x. A point x_1 is density-reachable from x_n if there is a chain of points $x_1, \cdots, x_n$ such that x_{p+1} is directly reachable from x_p, where $1 \leq p < p+1 \leq n$. Two points x_1 and x_n are density-connected if there is a point x_o such that both x_1 and x_n are density-reachable from x_o. Fig. 1(b) illustrates the main idea of DBSCAN algorithm and the main steps are:

1) Choose an arbitrary MPC sample x, if the neighborhood of a given radius ε contains sufficiently many points (more than $minPts$), a cluster j is started and the MPC sample x is labeled as visited.
2) Examine all the MPCs in the cluster j, if any MPC's ε-neighborhood has more than $minPts$ MPCs, all the MPCs are added into cluster j and are labeled as visited.
3) Continues steps 1 and 2 until all the density-connected MPCs are found for the cluster j.
4) Examine a new unvisited MPC and do steps 1-3, leading to the discovery of a further cluster.
5) Finally, label all the unvisited MPCs as noise samples.

As a non-partitional clustering algorithm, the DBSCAN can work well only if MPC clusters are largely separated from each other, otherwise, many clusters are erroneously merged due to the fact that they are density-reachable.

C. Proposed KPD Algorithm

To overcome the limitations of the current MPC clustering algorithms, we proposed the KPD algorithm, which is based on the recent work in [45]. The KPD is also a density-based clustering algorithm, and the main difference between KPD and DBSCAN is threefold: i) the KPD uses the Kernel density instead of using the number of samples; ii) the KPD uses the relative density (i.e., normalized within a local region) and a threshold is used to determine whether two clusters are connected; and iii) the impact of power is incorporated in the clustering. For the detailed analysis of algorithm robustness and extension (from machine learning view) can be found in [45], we present in the following steps of KPD from an MPC clustering point of view, including some necessary modifications to meet the requirement of MPC clustering:

1) *Calculating Density*: For each MPC sample, say x, calculate the density ρ using the K nearest MPCs as follows:

$$\begin{aligned}\rho_x = &\sum_{y\in K_x} \exp(\alpha_y) \times \exp\left(-\frac{|\tau_x-\tau_y|^2}{(\sigma_\tau)^2}\right) \\ &\times \exp\left(-\frac{|\Omega_{T,x}-\Omega_{T,y}|}{\sigma_{\Omega_T}}\right) \times \exp\left(-\frac{|\Omega_{R,x}-\Omega_{R,y}|}{\sigma_{\Omega_R}}\right) \\ &\times \exp\left(-\frac{|\Theta_{T,x}-\Theta_{T,y}|}{\sigma_{\Theta_T}}\right) \times \exp\left(-\frac{|\Theta_{R,x}-\Theta_{R,y}|}{\sigma_{\Theta_R}}\right)\end{aligned} \tag{4}$$

where y is an arbitrary MPC $y \neq x$. K_x is the set of the K nearest MPCs for the MPC x. $\sigma_{(\cdot)}$ is the standard deviation of the MPCs in the domain of $(\cdot)$. In (4), we use the Gaussian Kernel density for the delay domain as the physical channels do not favor a certain distribution of delay; we use the Laplacian Kernel density for the angular domain as it has been widely observed (both in the past and nowadays) that the angle of MPC follows the Laplacian distribution [15], [21], [34], [38], [46]–[49], and this assumption has also been widely used in standards [19], [50]. The term of $\exp(\alpha)$ in (4) shows that MPCs with strong power increase the density, which is intuitive as the weighting of dominant MPCs by power is quite natural. We heuristically use the exponential form of power to increase the power difference between MPCs to a reasonable level. We find that by including power into the Kernel density, cluster centroids are pulled to points with strong powers. It is noteworthy that the Kernel density in (4) can be adjusted as needed, e.g., if the elevation domain is not considered, the corresponding Kernel density functions of EOD and EOA can be removed. Fig. 2(b) shows an example plot of the estimated ρ using the measured data in [47], [48], where only AOD and AOA domains are considered.

2) *Calculating Relative Density*: For each MPC sample, calculate the relative density ρ^* using the K nearest MPCs' density, as follows:

$$\rho_x^* = \frac{\rho_x}{\max\limits_{y \in K_x \cup \{x\}} \rho_y} \quad . \tag{5}$$

Fig. 2(c) shows an example plot of the estimated ρ^*. By using the relative density, we normalize the density over different regions, which ensures that different clusters have similar level of density, so that it is able to identify the clusters with relatively weak power. It can be seen from (5) that $\rho^* \in (0, 1]$.

3) *Searching Key MPCs*: For each MPC x, if $\rho_x^* = 1$, label it as the key MPC $\hat{x}$. We thus obtain the set of key MPCs as follows:

$$\hat{\Phi} := \{x | x \in \Phi, \rho_x^* = 1\} \quad . \tag{6}$$

The key MPCs can be considered as the initial cluster centroids. As shown in Fig. 2(c), 8 key MPCs are identified.

4) *Clustering*: For each non-key MPC x, define its high-density-neighboring [51] MPC $\tilde{x}$ as:

$$\tilde{x} := \mathop{\arg\min}_{y \in \Phi, \rho_y > \rho_x} d(x, y) \tag{7}$$

where d represents the Euclidean distance. The high-density-neighboring MPC allows us to construct a directed graph $\zeta_1 = (\Phi, E_1)$ [52], where each vertex is a MPC and an arc exists from a non-key MPC x to its high-density-neighboring $\tilde{x}$. Namely,

$$E_1 = \{(x, \tilde{x}) | x \in \Phi \backslash \hat{\Phi}\} \tag{8}$$

where $(x, \tilde{x})$ denotes an arc (ordered pair) from x to $\tilde{x}$, and $\Phi \backslash \hat{\Phi}$ is the relative complement of set $\hat{\Phi}$ in set Φ. In ζ_1, starting from any non-key MPC and following the arcs, we will eventually reach a key MPC. Those MPCs which reach the same key MPC are grouped as one cluster. Fig. 1(c) shows the idea of KPD initial clustering.

5) *Cluster Merging*: For each MPC, we add edges [52] between it and its K nearest MPCs. Thus, we acquire a graph $\zeta_2 = (\Phi, E_2)$, where

$$E_2 = \{\{x, y\} | x \in \Phi, y \in K_x\} \tag{9}$$

and the unordered pair $\{x, y\}$ is an undirected edge between x and y. If there exists a path where any MPC has $\rho^* > \chi$ between two key MPCs in ζ_2, where χ is a density threshold, we merge the two key MPCs' clusters as one new cluster. Fig. 3 shows the details of merging using the key MPCs of 2 and 3 in Fig. 2(c) as an example. In Fig. 3(a), the red and blue circles are the K nearest MPCs for key MPCs 2 and 3, and the red and blue stars are the K nearest MPCs for the red and blue circles, respectively. It can be seen that for points a and b (as indicated by the arrows), the blue circles are the same points to the red stars, which means that key MPCs 2 and 3 can be connected to each other in ζ_2. Furthermore, it is found from Fig. 3(b) that almost all the MPCs in Fig. 3(a) have $\rho^* > \chi = 0.8$ (parameter selection will be discussed later in Section V), i.e., there is at least one path connecting the key MPCs 2 and 3, where any MPC in that path has $\rho^* > \chi$. Therefore, clusters of key MPCs 2 and 3 in Fig. 2(c) are merged. Similarly, clusters 4 and 5, 6 and 7 are merged respectively. We finally obtain the results in Fig. 2(d). Compared with the raw MPCs in Fig. 2(a), the resulting clusters in Fig. 2(d) look fairly convincing.

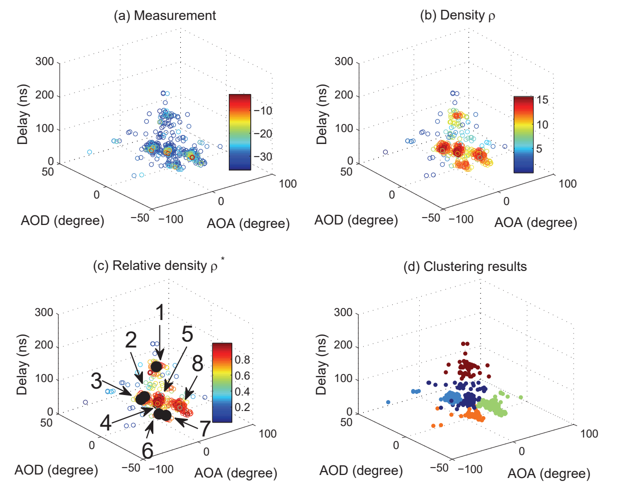

Fig. 2. Illustration of KPD clustering using the measured MPCs in [47], [48]. (a) Measured MPCs, where the color bar indicates power of MPC. (b) Plots of the estimated density ρ, where the color bar indicates the level of ρ. (c) Plots of the estimated density ρ^*, where the color bar indicates the level of ρ^*. The 8 solid black points are the key MPCs with $\rho^* = 1$. (d) Clustering results with the KPD algorithm, where the clusters are plotted with different colors.

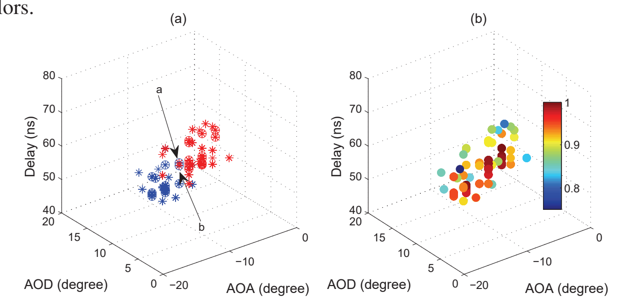

Fig. 3. Illustration of KPD clustering merging using the measured MPCs in [47], [48]. (a) K nearest MPCs for the key MPCs 2 and 3 in Fig. 2(c). The arrows indicate points a and b, which are shared in the K nearest MPCs of the key MPCs 2 and 3. (b) Plots of the estimated density ρ^* of the MPCs in (a), where the color bar indicates the level of ρ^*.

The details of the KPD algorithm are described by the pseudocode in Appendix, and the flowchart is shown in Fig. 4. The parameter selection is further discussed in Section V. Note that the proposed algorithm does not consider different channel polarizations, i.e., the power term in (1) and (4) is for a specific channel polarization. To extend the algorithm to

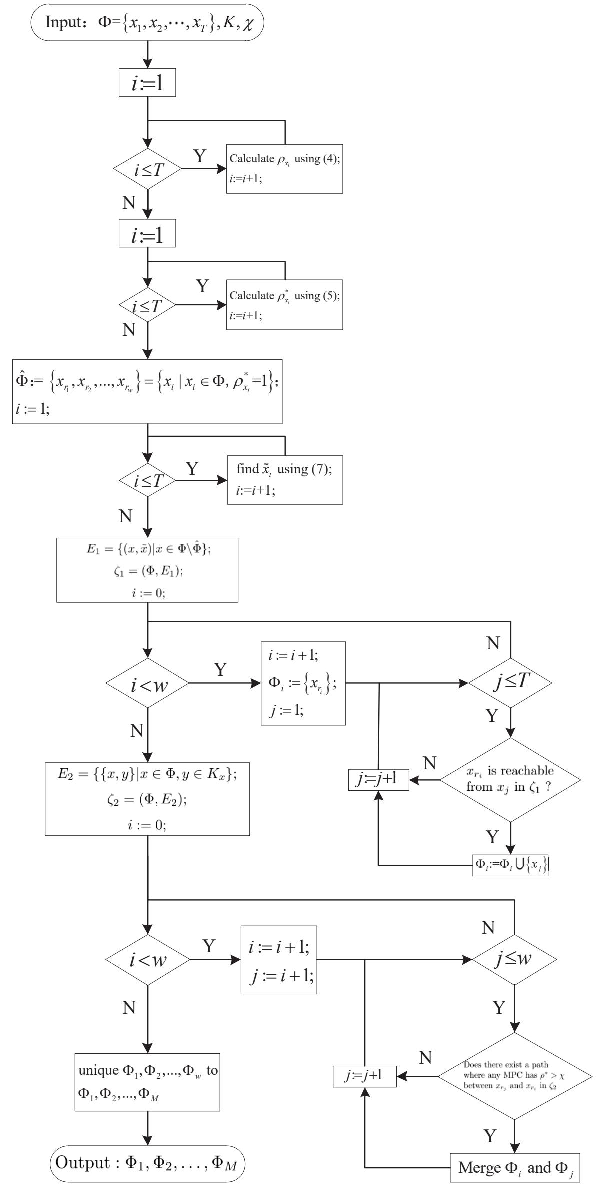

Fig. 4. Flowchart of the KPD algorithm implementation.

cover different polarizations, a polarimetric matrix needs to be applied to the power term as in [18], and all the formulations of the KPD algorithm need to be extended to incorporate this matrix. Such a modification is, however, not pursued any further in the present paper and is thus left as future work.

IV. Algorithm Analysis

A. Insight and Motivation

The steps in Section III.C generally involve three important features: Kernel density, K nearest MPCs and relative density, and cluster merging. Here we present some insights and motivations for the above steps. The reason of using the Kernel density is based on the idea in [53] that the variation of each data point can be modeled using a mathematical function that is called influence function. If the overall density of the data space is calculated as the sum of the influence functions of all data points, the mathematical form of the density function yields clustering with desired shape in a very compact mathematical form. For our MPC clustering, the variation of MPCs is usually modeled in a statistical way. We can thus use a mathematical function, namely the Kernel function, to incorporate the modeled behavior of MPCs, and the resulting Kernel density favors the clustering with desired shape. It is noteworthy that the Kernel function based MPC density in (4) is flexible: we can add the term of elevation angle accordingly if 3D MIMO measurements are used; we can also drop angular terms if angular information is not available. However, it is found that the estimated Kernel density level usually varies within a large range. We thus normalize it by using the relative density and only consider the K nearest MPCs. This approach ensures that the estimated density is sensitive to the local structure of the data, i.e., nearer neighbors contribute more, which is intuitive as the natural clusters are usually compact and separated. We find that this approach is necessary for a density-based MPC clustering algorithm, otherwise, a partitional clustering result are rarely obtained. Finally, it is necessary to merge clusters, as natural clusters have small-scale fading and intra-cluster power variation exists. Therefore, there are usually too many initial clusters according to the estimated key MPCs. We thus merge those clusters that are fairly close to each other. The step of cluster merging also guarantees that an acceptable performance is achieved even if an unsuitable K is used.

It must be noted that the proposed clustering algorithm assumes that MPCs have been correctly estimated from measurements and does not consider the impact of estimation errors of MPCs. Actually, in the natural environments all channel sounding signals typically come from the same source and may be reflected by the same scatterer, individual MPCs thus are correlated and cannot be assumed independent. This results in reduced accuracy of MPC estimation [1]. In addition, it is found in [11] that densely spaced MPCs can lead to large spreads in the estimated MPCs and such spreads could be erroneously considered by the clustering algorithm. It is possible to further improve clustering performance by considering the used MPC estimator to eliminate errors of estimation, which is thus another important goal of future work.

B. Algorithm Extension

Here we present some possible extensions for the proposed KPD algorithm. One of the key features of the algorithm is using the Laplacian Kernel density for angular domain. This assumption has been widely used [15], [21], [34], [49] and also validated by our own measurements [38], [47], [48], where the goodness of the fit for the Laplacian distribution is tested using Kolmogorov-Smirnov test and it passes at 5% significance level. However, in the above KPD framework the selection of Kernel density function is not fixed and can be flexible. In (4), other Kernel density such as Gaussian Kernel can also be applied to angular domain. In fact, angular characterization of 3D channels is a hot research topic [54], as the characteristics depend on the carrier frequency and propagation scenario, e.g., [55] suggests to use a uniform distribution for AOA at 28 GHz, and [56] suggests to use the Gaussian distribution for AOA and AOD at 73 GHz. However, angular channel characterization is not the scope of this paper. We suggest to use the Laplacian Kernel density based on the reported measurements and standards, but, the KPD framework allows to use other Kernel density functions for different applications.

V. Validation

This section presents the comparison and validation of different algorithms. For the KPM implementation, the initialization is done by using the *InitialGuess* step in [57]. The initial cluster number is set to range from 2 to 30, which covers the possible cases of cluster number and reduces the complexity reasonably. The Calinski-Harabasz and Davies-Bouldin indices [22] are used to determine the optimal cluster number. To ensure that KPM gives reasonable results, the parameter of *delay weighting factor* [29] is chosen by visual inspections for different experimental data (both simulated and measured). In this paper, the *delay weighting factor* is set to be 10, which ensures the optimal performance of KPM.

For the DBSCAN and KPD implementation, the data on delay and angular domains are firstly normalized by $\tau_i/\max(\tau)$, $(\Omega_i - \min(\Omega))\,/\,(\max(\Omega) - \min(\Omega))$, and $(\Theta_i - \min(\Theta))\,/\,(\max(\Theta) - \min(\Theta))$, which is found to increase the clustering accuracy. In DBSCAN, the neighborhood radius ε is set to be 0.2 and threshold $minPts$ is set to be the mean number of MPCs within the neighborhood radius in each snapshot, for both simulation and measurement validations. Again, those parameters are chosen by visual inspections of clustering results to ensure the optimal performance of algorithm.

A. Validation Measure

In this section, several measures are introduced to evaluate the clustering performance, i.e., measuring “goodness” of a clustering result comparing to other ones. Two widely used measurement criteria have been considered in this paper [58]: i) compactness, which means the member of each cluster should be as close to each other as possible; and ii) separation, which means the clusters themselves should be widely separated in data space. In the following, both simulated and measured channels are used for algorithm validation. For the simulated channels, the ground-truth partitions are available and the number and spread of clusters are synthesized and can be easily adjusted, which allows us to test algorithm performance under different conditions; whereas for the measured channels, the ground truth is not available and the data are more affected by noise and closer to real propagation environments. Therefore, we use two measures for different cases in this paper to better validate the KPD algorithm.

If the ground-truth partitions of MPCs are available, we use the F measure [59] to evaluate performance, which is a robust external quality measure that can be used to balance the precision and recall [60]. In order to facilitate the following discussions of validation measures, in this subsection we use “class” to represent the ground-truth partition and “cluster” to represent the partition determined by the algorithm. Given all the MPCs $\{x_1, \ldots, x_T\}$, let $L(x_i)$ and $C(x_i)$ denote the class label (the label of the class that x_i comes from) and the cluster label (the label of the cluster that x_i belongs to) of x_i, respectively. The correctness of the relation between x_i and x_j is given by

$$\text{Correctness}(x_i, x_j) = \begin{cases} 1 & \text{if } L(x_i) = L(x_j) \text{ and } C(x_i) = C(x_j) \\ 0 & \text{otherwise} \end{cases}. \tag{10}$$

The correctness term measures whether the clustering algorithm gives correct result. Only when x_i and x_j come from the same class ($L(x_i) = L(x_j)$) and also both belong to the same cluster ($C(x_i) = C(x_j)$), it is correct (i.e., the value equals to 1). Then, the terms of PB and RB [61] are defined as:

$$PB = \frac{1}{T}\sum_{i=1}^{T} \frac{\sum_{x_j : i\neq j, C(x_i)=C(x_j)} \text{Correctness}(x_i, x_j)}{|\{x_j | i \neq j, C(x_i) = C(x_j)\}|} \tag{11}$$

$$RB = \frac{1}{T}\sum_{i=1}^{T} \frac{\sum_{x_j : i\neq j, L(x_i)=L(x_j)} \text{Correctness}(x_i, x_j)}{|\{x_j | i \neq j, L(x_i) = L(x_j)\}|} \tag{12}$$

where $|\{\cdot\}|$ denotes element number of the set $\{\cdot\}$. In the ratio in (11), for a certain MPC x_i, the numerator counts how many other MPCs which belong to the same cluster (i.e., $i \neq j$, $C(x_i) = C(x_j)$) come from the same class (i.e. $L(x_i) = L(x_j)$), and the denominator ensures that the value of the ratio falls into the range of 0 to 1. This ratio is called the BCubed precision of x_i [59]. Thus PB in (11) is the average of BCubed precision for all the MPCs. Accordingly, the numerator of the ratio in (12) counts how many other MPCs that come from the same class (i.e., $i \neq j$, $L(x_i) = L(x_j)$) belong to the same cluster (i.e. $C(x_i) = C(x_j)$), and the denominator also ensures that the value of the ratio falls into the range of 0 to 1 as in (11). The ratio in (12) is called the BCubed recall of x_i [59]. Therefore RB in (12) is the average of BCubed recall for all the MPCs. To integrate them into a fair metric, the F measure is defined by

$$F = \frac{2 \cdot PB \cdot RB}{PB + RB}. \tag{13}$$

The value of the F measure ranges from 0 to 1, and a larger value indicates higher clustering quality.

If the ground-truth partitions of MPCs are not available, e.g., clustering for the real-world measurements, the F measure cannot be used because the class labels are unknown. In this case, we use the silhouette coefficient S [62], which provides a succinct graphical representation of how well each object lies within its cluster. Take any MPC x in the data set and denote by A the cluster to which it has been assigned, we have

$$a(x) := E[\{d(x, y)\,|x \in A, y \in A, x \neq y\}] \tag{14}$$

where $E[\cdot]$ denotes the sample mean value of the set $(\cdot)$. Consider any cluster B which is different from A, we have

$$d_S(x, B) := E[\{d(x, y)\,|x \in A, y \in B, A \neq B\}] \tag{15}$$

and then we have

$$b(x) = \min_{B \neq A} d_S(x, B). \tag{16}$$

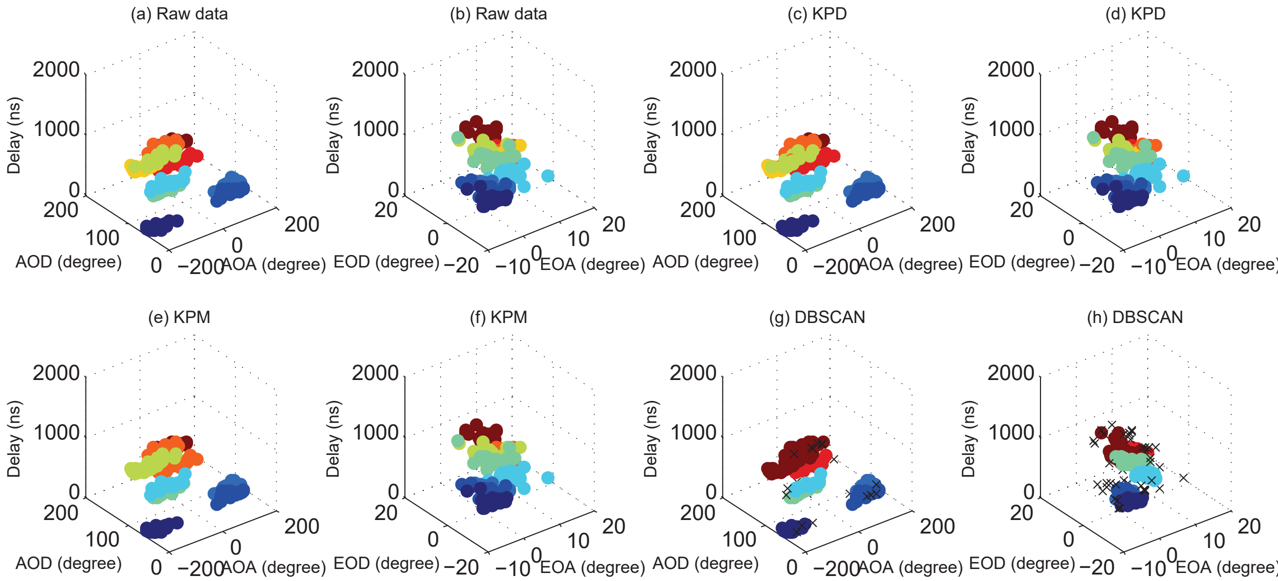

Fig. 5. Clustering algorithm validation with simulated 3D channels. (a) and (b) Simulated MPCs, where the raw clusters are plotted with different colors. (c) and (d) Clustering results with the proposed KPD algorithm. (e) and (f) Clustering results with the KPM algorithm. (g) and (h) Clustering results with the DBSCAN algorithm, where the black x is the noise point resulting from the DBSCAN.

The silhouette coefficient is thus defined as follows:

$$S = E\left[\frac{b(x) - a(x)}{\max\{a(x), b(x)\}}\right]_{x\in\Phi}. \tag{17}$$

From the above definition it is clear that S ranges from -1 to 1. The silhouette coefficient is a measure of how tightly grouped all the data in the cluster are, and a larger value indicates higher clustering quality. The silhouette coefficient only depends on the actual partition of the objects, and not on the clustering algorithm that was used to obtain it. Therefore, it can be used to compare the output of different clustering algorithms, which is especially useful when the ground truth is not available [63]. Note that the silhouette coefficient can also be used when the ground truth is available.

B. Simulation Validation

In this subsection, the KPD algorithm is validated by using a simulated channel, where the ground truth is available. The 3GPP 3D MIMO channel model [50] is used to generate the synthetic MPCs. We consider the Urban Micro scenario with line-of-sight (LOS) and the carrier frequency is 2 GHz. The number of clusters can be adjusted and MPC number in each cluster is fixed to 20. Fig. 5 shows an example plot of the raw clusters in the simulated 3D channel and the clustering results by using different algorithms. We generated 10 clusters with different power and delay/angle positions. It can be seen that the KPM leads to wrong clustering decisions for the MPCs with 100 to 200 degrees AOD and -200 to 0 degrees AOA, and the DBSCAN leads to a wrong cluster number; whereas the KPD has almost 100% correct identification as shown in Figs. 5(c) and (d).

Furthermore, we test the performance of the algorithm under different "cluster conditions". We consider two cluster conditions: cluster number and cluster angular spread. Intuitively, a channel with large cluster number and angular spread would have reduced clustering performance. Both the F measure and silhouette coefficient are used for performance comparison.

We first test the impact of the cluster number on the clustering accuracy. We still use the 3GPP 3D MIMO channel model to generate MPCs, and different cluster numbers are used in the simulation, ranging from 3 to 24, which generally covers all the conditions according to the literature. 300 random channels are simulated for each cluster number case. Fig. 6 shows the comparison among three clustering algorithms using the F measure. It is observed that the proposed KPD algorithm, having the highest value of the F measure, shows the best performance, and the value of the F measure decreases only slightly for larger cluster numbers. The KPM and DBSCAN algorithms show good performance only for a small number of clusters, and their values of the F measure decease strongly with increasing cluster number. Fig. 7 shows the performance comparison using the silhouette coefficient. It is also found that the KPD algorithm has the best performance.

Then we test the impact of cluster angular spread on the clustering accuracy. In the simulation, the number of clusters is fixed and different spreads are introduced by adding white Gaussian noise with variances of $\{1°, 2°, \ldots, 30°\}$ to MPC angles [64]. 300 random channels are simulated for each cluster angular spread. Fig. 8 shows the impact of cluster angular spread on the F measure and silhouette coefficient, where 10 and 20 clusters are simulated respectively. It is found from Figs. 8(a) and (c) that the F measure generally decreases with the increasing cluster angular spread. The KPD algorithm shows best performance for arbitrary cluster sizes, especially when cluster number is high, the performances of KPM and DBSCAN show significant degradations whereas KPD still has good performance. This can be explained by the use of the Laplacian Kernel density, as the simulation model assumes

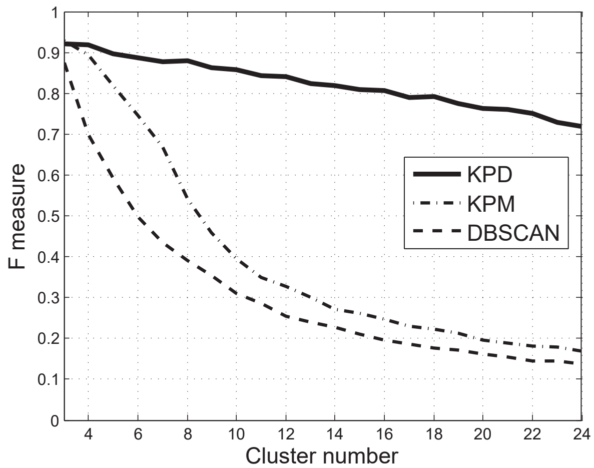

Fig. 6. Impact of cluster number on the F measure.

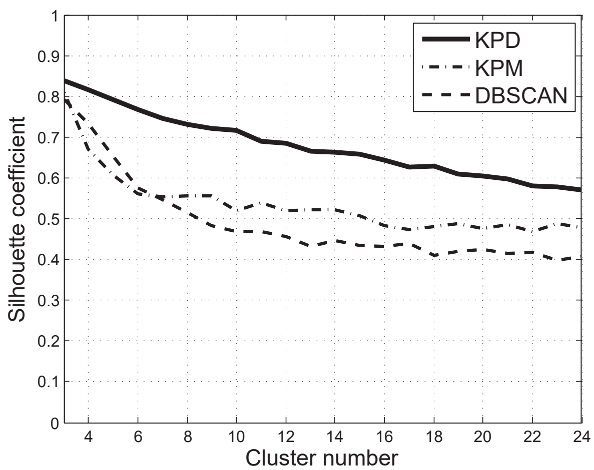

Fig. 7. Impact of cluster number on the silhouette coefficient.

a Laplacian angular distribution for MPCs. The curves of silhouette coefficient in Figs. 8(b) and (d) also show that KPD has the best performance, however, when the additive angular spread is large, the silhouette coefficient of KPD is similar, or only slightly higher than KPM and DBSCAN.

Finally, we test the algorithms when the channel is sparse and there are only a few MPCs in each cluster. In such a case, the statistical behavior of MPCs in each cluster is not obvious anymore. We consider the cases that there are 10 clusters and only 5 MPCs in each cluster. As there is no standard model that considers this case, we synthesize the MPCs as follows: each cluster center is generated randomly in delay and 3D angle domains using the uniform distribution; for each cluster, the power and angles are generated by using the model and parameters of Urban Micro scenario in [50], though the distribution fit is poor due to the small number of MPCs. Fig. 9 shows an example plot of the simulated channels and clustering results using different algorithms. For simplicity we only plot AOA and EOA domains. It is found that the KPD has 100% correct identification, and the KPM leads to wrong clustering decisions for the MPCs with 0 to 100 degrees AOA and 0 to 100 degrees EOA. The DBSCAN still performs the worst. Furthermore, 300 random channels are simulated with the same cluster condition. Fig. 10 shows

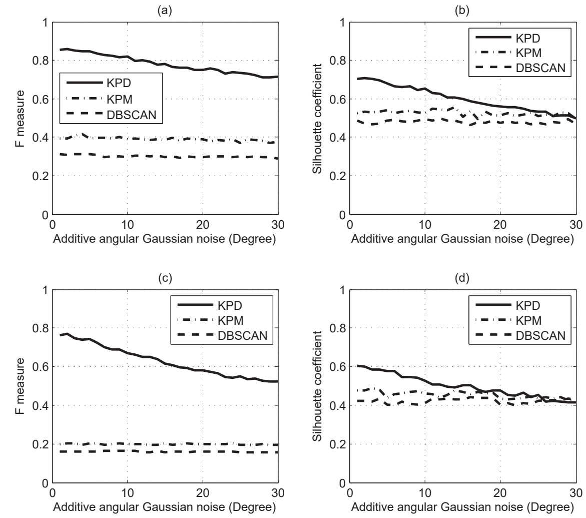

Fig. 8. Impact of cluster angular spread on the F measure and silhouette coefficient. In (a) and (b), 10 clusters are simulated. In (c) and (d), 20 clusters are simulated.

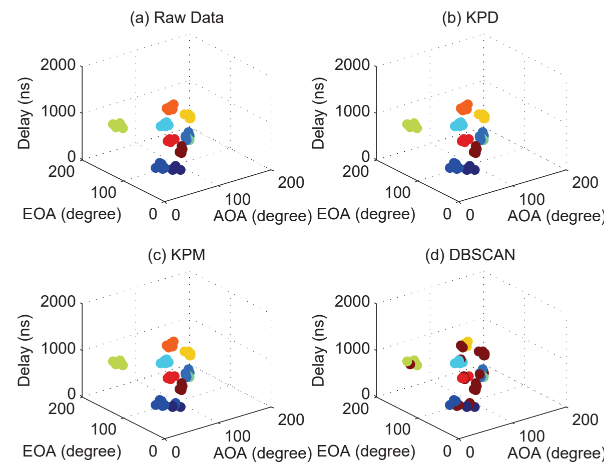

Fig. 9. Clustering algorithm validation with simulated channels with 10 clusters, and each cluster has 5 MPCs. (a) Simulated MPCs, where the raw clusters are plotted with different colors. (b) Clustering results with the proposed KPD algorithm. (c) Clustering results with the KPM algorithm. (d) Clustering results with the DBSCAN algorithm.

the cumulative distribution function (CDF) comparisons of the F measure and silhouette coefficient for the clustering results of 300 simulated channels. It can be seen that the KPD has the best performance.

C. Measurement Validation

In this subsection, the KPD algorithm is validated with channel measurements. UWB data measured at the University of Southern California (USC), USA are used in the validation. For space reasons, we only give a brief summary; details of the measurement campaign can be found in [47], [48].

The measurements were performed in an indoor warehouse facility at USC, including both LOS and non-LOS (NLOS) propagation scenarios. A HP8720ET vector network analyzer (VNA) was used to measure the complex transfer function of the channel, in the 2-8 GHz frequency range. Two vertically polarized omni-directional antennas were used with a height of

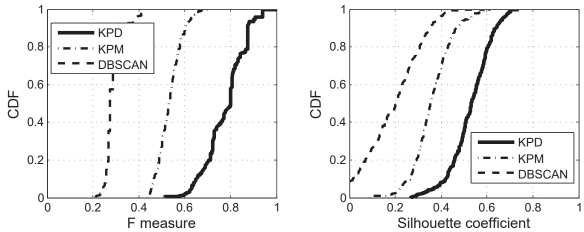

Fig. 10. CDF plot of the estimated F measure and silhouette coefficients with different clustering algorithms, based on 300 random channel simulations where 10 clusters are simulated and each cluster has 5 MPCs.

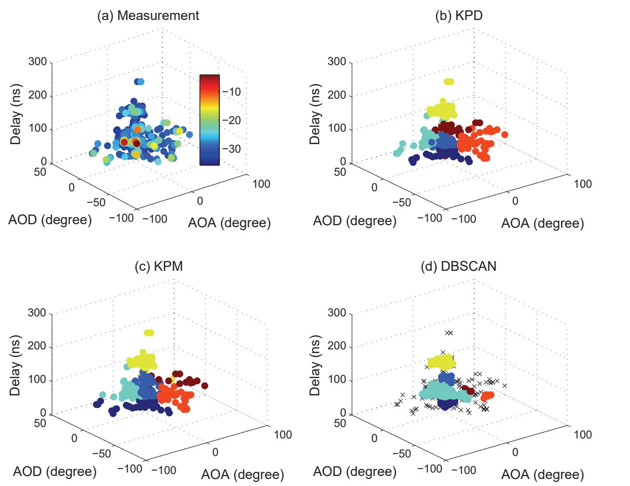

Fig. 11. Clustering algorithm validation with measured channels. (a) Measured MPCs at USC indoor environments, where the color bar indicates the power of MPC. (b) Clustering results with the proposed KPD algorithm. (c) Clustering results with the KPM algorithm. (d) Clustering results with the DBSCAN algorithm, where the black x is the noise point resulting from the DBSCAN.

1.78 m. At each position, 64 realizations of the local channel were measured by moving TX and RX over 8 positions with a 50 mm step interval respectively, i.e., a 8x8 MIMO channel was measured. The high-resolution CLEAN algorithm [42] was used to extract the delay and direction information of the MPCs. The resulting MPCs have a number between 400 and 1000 in each snapshot, and can be used to validate our proposed algorithm in the following. It is noteworthy that when applying KPM to the measured data, the Calinski-Harabasz and Davies-Bouldin indices are found to be very sensitive to the outliers in the data and cannot give reasonable results. We thus use visual inspection to determine the number of clusters for each measured snapshot and pick the result that gives the visually most compact clusters. More details of the process can be found in [48].

Fig. 11(a) shows the raw MPCs extracted from a sample LOS measurement, which are color-coded with their power. Visual inspection gives the impression of a large number of crowded MPCs in space, which is a usual case for the real-world measurements. We only consider AOD and AOA domains as we used a virtual uniform linear array in measurements and the elevation of MPC cannot be extracted. After applying different clustering algorithms, we find that all the three algorithms lead to 6 clusters, though the DBSCAN labels many MPCs as noise points as they are border points. As

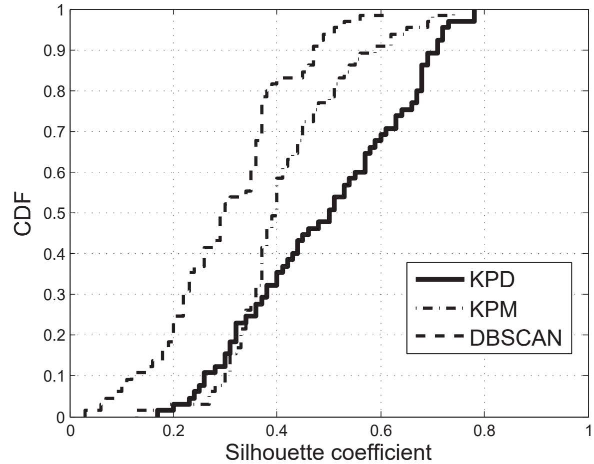

Fig. 12. CDF plot of the estimated silhouette coefficients with different clustering algorithms.

shown in Fig. 11(b), the KPD resulting partition into 6 clusters realizes a reasonable trade-off between cluster compactness and separation, where cluster centers are attracted by strong powers. The parts of light blue and brown clusters in Fig. 11(b) are grouped together in Fig. 11(c).

It is noteworthy that the above comparison is somewhat heuristic, as it is difficult to find the ground truth in the real-world measurements. Therefore we use the silhouette coefficient to evaluate the clustering quality of measurements. The estimated silhouette coefficients in Fig. 11 for KPD, KPM, and DBSCAN are found to be 0.45, 0.36, and 0.31, respectively. It shows that the KPD algorithm outperforms the other algorithms for this data set. To further validate the algorithms, all the measurements in [47] are used and the CDF comparison of the silhouette coefficient S with different algorithms is shown in Fig. 12. It is found that KPD generally has the highest value of S, whereas the DBSCAN performs the worst. It is also observed that for some cases the KPM shows similar performance with KPD. Those cases are mainly NLOS scenarios with large cluster number and angular spread. This phenomenon can also be validated by the results in Figs. 8(b) and (d). The comparison shows that the proposed KPD leads to compact and clearly separated clusters.

D. Parameter Selection

In the KPD algorithm, two parameters are required: K and χ. K determines how many MPCs are used to calculate density and to yield ζ_2. A small K increases the sensitivity of local density variation to the clustering results, i.e., reduces the size of local region. We use $K = \sqrt{T/2}$ as suggested in [45] and a heuristic argument is as follows: in general, each cluster has $\sqrt{2T}$ points [61], whereas our algorithm requires that any two MPCs in each cluster are connected in ζ_2 so that the cluster is compact. However, a $K = \sqrt{2T}$ usually fails to yield such compactness [45] (i.e., any two MPCs in each cluster may not be connected in ζ_2), therefore, we use $K = (\sqrt{2T})/4 = \sqrt{T/2}$ as a heuristic approach to reduce the size of local region and to ensure the compactness of clustering.

χ determines whether two clusters can be merged. A large χ leads to a large number of clusters. [45] proposes a fast

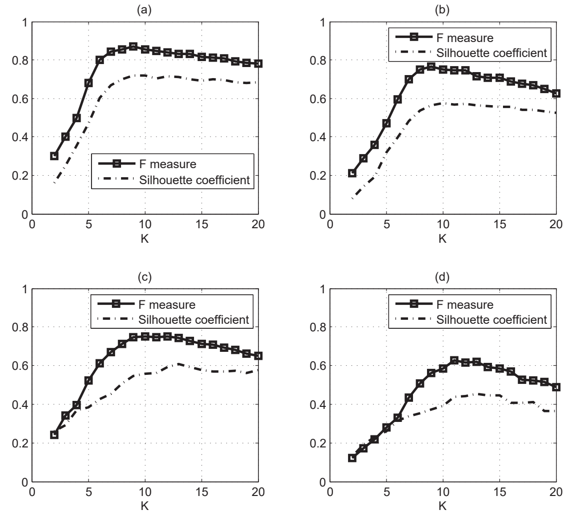

Fig. 13. Impact of K on the F measure and silhouette coefficient based on random simulations. (a) 10 clusters, additive Gaussian noise 1°. (b) 10 clusters, additive Gaussian noise 20°. (c) 20 clusters, additive Gaussian noise 1°. (d) 20 clusters, additive Gaussian noise 20°.

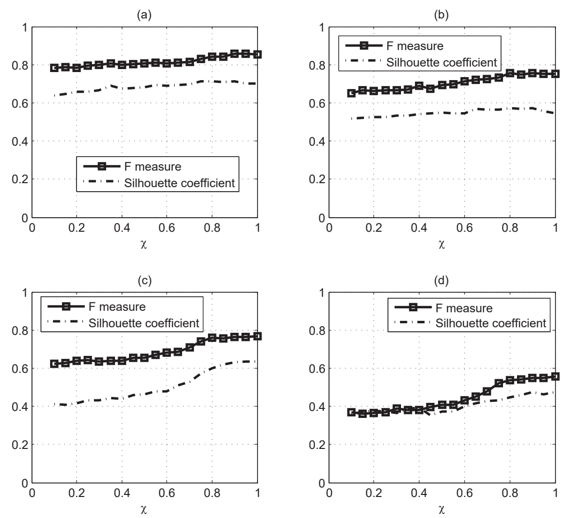

Fig. 14. Impact of χ on the F measure and silhouette coefficient based on random simulations. (a) 10 clusters, additive Gaussian noise 1°. (b) 10 clusters, additive Gaussian noise 20°. (c) 20 clusters, additive Gaussian noise 1°. (d) 20 clusters, additive Gaussian noise 20°.

approach to determine χ. For simplicity, we suggest to set $\chi = 0.8$, which is found to have a reasonable performance as reported in Section V.B and Section V.C. A heuristic argument for $\chi = 0.8$ is that a large value of χ ensures that the clusters are separated from each other.

Finally, we experimentally investigate the sensitivities of K and χ to the clustering quality under different channel conditions. We still use the 3GPP 3D MIMO channel model and simulate 300 random channels for different conditions, so that both the F measure and silhouette coefficient can be used for evaluation. Fig. 13 shows the impact of K on the F measure and silhouette coefficient, with different channel conditions. It is observed that the curves of F measure and silhouette coefficient generally are first increasing, and then decreasing with K. This is because a small K fails to reflect the density in a local region and a large K smoothes density and erroneously drops local variations. In the simulation of Figs. 13(a) and (b), $\sqrt{T/2} = 10$, which corresponds to high values of F measure and silhouette coefficient; in Figs. 13(c) and (d), $\sqrt{T/2} = 14$, which also corresponds to high values of F measure and silhouette coefficient. Therefore, Fig. 13 validates that $K = \sqrt{T/2}$ has fairly good performance. We thus suggest $K = \sqrt{T/2}$ for KPD clustering of MPCs. Fig. 14 shows the impact of χ on the F measure and silhouette coefficient, with different channel conditions. It can be seen that the curves of F measure and silhouette coefficient generally increases with χ. This is because a large χ reduces the erroneous cluster merging. It is also found that the F measure and silhouette coefficient are fairly steady when $\chi > 0.8$. Therefore, we suggest to use $\chi = 0.8$ for KPD clustering of MPCs.

VI. Conclusion

In this paper, a KPD based algorithm is proposed for MPC automatical clustering. The main features are: i) it uses the Kernel density to incorporate the modeled behavior of MPCs into the clustering algorithm, which is also flexible for implementation; ii) it uses the relative density and only considers the K nearest MPCs in the density estimation, which is able to better identify the local density variations of MPCs; iii) it uses a heuristic approach to merge clusters, which improves the clustering performance; iv) the algorithm provides a trustworthy clustering result with a small number of user input, and almost no performance degradation occurs even with a large number of clusters and large cluster angular spread, which outperforms other algorithms; and v) the algorithm has a fairly low computational complexity. Finally, both synthetic and measured MIMO channel data validate the proposed KPD algorithm. The results in this paper can be used to cluster real-world measurement data, and used for the cluster based channel modeling for 4G/5G communications. Modification of the algorithm to account for polarimetric data, and the investigation of the impact of clustering algorithm improvement on channel modeling performance, which further validates the proposed algorithm, could be topics for future work.

Appendix

The pseudocode of KPD algorithm is presented as follows:
Algorithm KPD

1) **Input** $\Phi = \{x_1, x_2, \cdots, x_T\}$, K, and χ
   % Calculating density
2) **for** $i = 1 : T$
3) calculate ρ_{x_i} using (4)
4) **end**
   % Calculating relative density
5) **for** $i = 1 : T$
6) calculate $\rho^*_{x_i}$ using (5)
7) **end**
   % Searching key MPCs
8) $\hat{\Phi} := \{x_{r_1}, x_{r_2}, \cdots, x_{r_w}\} = \left\{x_i \,|x_i \in \Phi \,, \rho^*_{x_i} = 1\right\}$
   % Clustering

for $i = 1 : T$
    find $\tilde{x}$ using (7)
end
$E_1 = \{(x, \tilde{x}) | x \in \Phi \backslash \hat{\Phi}\}$
$\zeta_1 = (\Phi, E_1)$
for $i = 1 : w$
    $\Phi_i := \{x_{r_i}\}$
    for $j = 1 : T$
        if x_{r_i} is reachable from x_j in ζ_1
            $\Phi_i := \Phi_i \cup \{x_j\}$
        end
    end
end
% Clustering merging
$E_2 = \{\{x, y\} | x \in \Phi, y \in K_x\}$
$\zeta_2 = (\Phi, E_2)$
for $i = 1 : w$
    for $j = i + 1 : w$
        if if there exists a path where any MPC has $\rho^* > \chi$ between x_{r_j} and x_{r_i} in ζ_2
            merge Φ_i and Φ_j
        end
    end
end
%Remove redundancy terms
unique $\Phi_1, \Phi_2, \ldots, \Phi_w$ to $\Phi_1, \Phi_2, \ldots, \Phi_M$
return $\Phi_1, \Phi_2, \ldots, \Phi_M$

Acknowledgement

The authors thank the anonymous reviewers for their thorough reading and constructive comments, which greatly helped to improve the paper.

References

[1] A. F. Molisch, *Wireless communications 2nd ed.* Wiley, 2010.
[2] W. Wang, T. Jost, U. Fiebig, and W. Koch, "Time-variant channel modeling with application to mobile radio based positioning," in *Proc. IEEE GLOBECOM'12*, 2012, pp. 5038–5043.
[3] C.-C. Chong, C.-M. Tan, D. I. Laurenson, S. McLaughlin, M. A. Beach, and A. R. Nix, "A novel wideband dynamic directional indoor channel model based on a Markov process," *IEEE Transactions on Wireless Communication*, vol. 4, no. 4, pp. 1539–1552, 2005.
[4] R. He, O. Renaudin, V.-M. Kolmonen, K. Haneda, Z. Zhong, B. Ai, and C. Oestges, "A dynamic wideband directional channel model for vehicle-to-vehicle communications," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7870–7882, 2015.
[5] M. Failli, *Digital Land Mobile Radio Communications - COST 207*. EC, 1989.
[6] G. Acosta-Marum and M. A. Ingram, "Six time-and frequency-selective empirical channel models for vehicular wireless LANs," *IEEE Vehicular Technology Magazine*, vol. 2, no. 4, pp. 4–11, 2007.
[7] A. F. Molisch and F. Tufvesson, "Propagation channel models for next-generation wireless communications systems," *IEICE Transactions on Communications*, vol. 97, no. 10, pp. 2022–2034, 2014.
[8] E. Bonek, "MIMO propagation channel modeling," in *IEEE Proc. EuCAP'13*, 2013, pp. 2488–2492.
[9] H. Hofstetter, A. F. Molisch, and N. Czink, "A twin-cluster MIMO channel model," in *IEEE Proc. EuCAP'06*, 2006, pp. 1–8.
[10] L. Liu, C. Oestges, J. Poutanen, K. Haneda, P. Vainikainen, F. Quitin, F. Tufvesson, and P. Doncker, "The COST 2100 MIMO channel model," *IEEE Wireless Commun.*, vol. 19, no. 6, pp. 92–99, 2012.
[11] M. Bengtsson and B. Volcker, "On the estimation of azimuth distributions and azimuth spectra," in *IEEE Proc. VTC'01*, 2001, pp. 1612–1615.
[12] H. Suzuki, "A statistical model for urban radio propogation," *IEEE Transactions on communications*, vol. 25, no. 7, pp. 673–680, 1977.
[13] A. A. Saleh and R. Valenzuela, "A statistical model for indoor multipath propagation," *IEEE Journal on Selected Areas in Communications*, vol. 5, no. 2, pp. 128–137, 1987.
[14] J. Fuhl, A. F. Molisch, and E. Bonek, "Unified channel model for mobile radio systems with smart antennas," in *IEE Proceedings-Radar, Sonar and Navigation*, vol. 145, no. 1. IET, 1998, pp. 32–41.
[15] C. Gustafson, K. Haneda, S. Wyne, and F. Tufvesson, "On mm-wave multipath clustering and channel modeling," *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 3, pp. 1445–1455, 2014.
[16] M. Gan, Z. Xu, C. F. Mecklenbrauker, and T. Zemen, "Cluster lifetime characterization for vehicular communication channels," in *Proc. EuCAP'15*, 2015, pp. 1–5.
[17] T. Santos, J. Karedal, P. Almers, F. Tufvesson, and A. F. Molisch, "Modeling the ultra-wideband outdoor channel: Measurements and parameter extraction method," *IEEE Transactions on Wireless Communications*, vol. 9, no. 1, pp. 282–290, 2010.
[18] A. F. Molisch, H. Asplund, R. Heddergott, M. Steinbauer, and T. Zwick, "The COST259 directional channel model - part I: Overview and methodology," *IEEE Transactions on Wireless Communications*, vol. 5, no. 12, pp. 3421–3433, 2006.
[19] 3GPP TSG RAN, "Spatial channel model for multiple input multiple output (MIMO) simulations," 3GPP, Tech. Rep., pp. 1-41, 2004.
[20] J. Meinilä, P. Kyösti, T. Jämsä, and L. Hentilä, "WINNER II channel models," *Radio Technologies and Concepts for IMT-Advanced*, pp. 39–92, 2009.
[21] R.-M. Cramer, R. A. Scholtz, and M. Z. Win, "Evaluation of an ultra-wide-band propagation channel," *IEEE Transactions on Antennas and Propagation*, vol. 50, no. 5, pp. 561–570, 2002.
[22] R. Xu, D. Wunsch *et al.*, "Survey of clustering algorithms," *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 645–678, 2005.
[23] C. Gentile, "Using the Kurtosis measure to identify clusters in wireless channel impulse responses," *IEEE Transactions on Antennas and Propagation*, vol. 61, no. 6, pp. 3392–3395, 2013.
[24] S. C. Zhu and A. Yuille, "Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 9, pp. 884–900, 1996.
[25] J. Chuang, S. Bashir, and D. G. Michelson, "Automated identification of clusters in UWB channel impulse responses," in *Proc. IEEE CCECE'07*, 2007, pp. 761–764.
[26] R. He, W. Chen, B. Ai, A. F. Molisch, W. Wang, Z. Zhong, J. Yu, and S. Sangodoyin, "On the clustering of radio channel impulse responses using sparsity-based methods," *IEEE Transactions Antennas and Propagation*, vol. 64, no. 6, pp. 2465–2474, 2016.
[27] N. Czink, P. Cera, J. Salo, E. Bonek, J.-P. Nuutinen, and J. Ylitalo, "A framework for automatic clustering of parametric MIMO channel data including path powers," in *Proc. IEEE VTC'06*, 2006, pp. 1–5.
[28] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proc. BSMSP'67*, 1967, pp. 281–297.
[29] M. Steinbauer, H. Ozcelik, H. Hofstetter, C. F. Mecklenbrauker, and E. Bonek, "How to quantify multipath separation," *IEICE Transactions on Electronics*, vol. 85, no. 3, pp. 552–557, 2002.
[30] C. Schneider, M. Bauer, M. Narandžić, W. Kotterman, and R. S. Thomä, "Clustering of MIMO channel parameters-performance comparison," in *Proc. IEEE VTC'09*, 2009, pp. 1–5.
[31] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise." in *Proc. ACM KDD'96*, 1996, pp. 226–231.
[32] M. K. Samimi and T. S. Rappaport, "3-D statistical channel model for millimeter-wave outdoor mobile broadband communications," in *Proc. IEEE ICC'15*, 2015, pp. 2430–2436.
[33] 3GPP TSG RAN WG1 R1-163115, "A hierarchical agglomerative clustering algorithm for channel modelling," 3GPP, Tech. Rep., pp. 1-6, 2016.
[34] Q. H. Spencer, B. D. Jeffs, M. A. Jensen, and A. L. Swindlehurst, "Modeling the statistical time and angle of arrival characteristics of an indoor multipath channel," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 347–360, 2000.
[35] S. Mota, F. Perez-Fontan, and A. Rocha, "Estimation of the number of clusters in multipath radio channel data sets," *IEEE Transactions on Antennas and Propagation*, vol. 61, no. 5, pp. 2879–2883, 2013.
[36] S. Mota, M. O. Garcia, A. Rocha, and F. Pérez-Fontán, "Clustering of the multipath radio channel parameters," in *Proc. EUCAP'11*, 2011, pp. 3232–3236.
[37] C. Schneider, M. Ibraheam, S. Häfner, M. Käske, M. Hein, and R. S. Thomä, "On the reliability of multipath cluster estimation in realistic channel data sets," in *Proc. EuCAP'14*, 2014, pp. 449–453.

[38] S. Sangodoyin, V. Kristem, C. U. Bas, M. Kaske, J. Lee, C. Schneider, G. Sommerkorn, J. Zhang, R. Thomä, and A. F. Molisch, "Cluster-based analysis of 3D MIMO channel measurement in an urban environment," in *Proc. IEEE MILCOM'15*, 2015, pp. 744–749.

[39] R. He, Q. Li, B. Ai, Y. L. A. Geng, A. F. Molisch, V. Kristem, Z. Zhong, and J. Yu, "An automatic clustering algorithm for multipath components based on Kernel-Power-Density," in *Proc. IEEE WCNC'17*, 2017, pp. 1–6.

[40] M. Steinbauer, A. F. Molisch, and E. Bonek, "The double-directional radio channel," *IEEE Antennas and Propagation Magazine*, vol. 43, no. 4, pp. 51–63, 2001.

[41] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[42] J. Högbom, "Aperture synthesis with a non-regular distribution of interferometer baselines," *Astronomy and Astrophysics Supplement Series*, vol. 15, p. 417, 1974.

[43] B. H. Fleury, M. Tschudin, R. Heddergott, D. Dahlhaus, and K. I. Pedersen, "Channel parameter estimation in mobile radio environments using the SAGE algorithm," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 3, pp. 434–450, 1999.

[44] A. Richter, "Estimation of radio channel parameters: models and algorithms," Ph.D. dissertation, Technischen Universitat Ilmenau, Ilmenau, Germany, Dec. 2005.

[45] Y. L. A. Geng, Q. Li, R. Zheng, F. Zhuang, and R. He, "RECOME: a new density-based clustering algorithm using relative KNN kernel density," in *Proc. IEEE International Conference on Data Mining*, submitted, 2016.

[46] D. S. Baum, J. Hansen, and J. Salo, "An interim channel model for beyond-3G systems: extending the 3GPP spatial channel model (SCM)," in *Proc. IEEE VTC'05*, 2005, pp. 3132–3136.

[47] S. Sangodoyin, R. He, A. Molisch, V. Kristem, and F. Tufvesson, "Ultra-wideband MIMO channel measurements and modeling in a warehouse environment," in *Proc. IEEE ICC'15*, 2015, pp. 1–6.

[48] S. Sangodoyin, V. Kristem, A. F. Molisch, R. He, F. Tufvesson, and H. Behairy, "Statistical modeling of ultrawideband MIMO propagation channel in a warehouse environment," *IEEE Transactions on Antennas and Propagation*, vol. 64, no. 9, pp. 4049–4063, 2016.

[49] S. Hur, S. Baek, B. Kim, Y. Chang, A. F. Molisch, T. S. Rappaport, K. Haneda, and J. Park, "Proposal on millimeter-wave channel modeling for 5G cellular system," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 3, pp. 454–469, 2016.

[50] 3GPP TR RAN, "Study on 3D channel model for LTE," 3GPP, Tech. Rep., pp. 1-42, 2015.

[51] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.

[52] J. Bondy and U. Murty, *Graph theory (graduate texts in mathematics)*. Springer New York, 2008.

[53] A. Hinneburg and D. A. Keim, "An efficient approach to clustering in large multimedia databases with noise," in *Proc. IEE KDDM'98*, 1998, pp. 58–65.

[54] A. F. Molisch, *3D Propagation Channels: Modeling and Measurements, Signal Processing for 5G: Algorithms and Implementations,*. Wiley, 2016.

[55] M. Samimi, K. Wang, Y. Azar, G. N. Wong, R. Mayzus, H. Zhao, J. K. Schulz, S. Sun, F. Gutierrez, and T. S. Rappaport, "28 GHz angle of arrival and angle of departure analysis for outdoor cellular communications using steerable beam antennas in New York City," in *Proc. IEEE VTC'13*, 2013, pp. 1–6.

[56] T. A. Thomas, H. C. Nguyen, G. R. MacCartney, and T. S. Rappaport, "3D mmWave channel model proposal," in *Proc. IEEE VTC'14*, 2014, pp. 1–6.

[57] N. Czink, R. Tian, S. Wyne, F. Tufvesson, J.-P. Nuutinen, J. Ylitalo, E. Bonek, and A. F. Molisch, "Tracking time-variant cluster parameters in MIMO channel measurements," in *Proc. IEEE CHINACOM'07*, 2007, pp. 1147–1151.

[58] F. Kovács, C. Legány, and A. Babos, "Cluster validity measurement techniques," in *Proc. ISHRCI'05*, 2005, pp. 1–11.

[59] E. Amigó, J. Gonzalo, J. Artiles, and F. Verdejo, "A comparison of extrinsic clustering evaluation metrics based on formal constraints," *Information retrieval*, vol. 12, no. 4, pp. 461–486, 2009.

[60] M. Steinbach, G. Karypis, V. Kumar *et al.*, "A comparison of document clustering techniques," in *Proc. KDD workshop on Text Mining'00*, 2000, pp. 525–526.

[61] J. Han, M. Kamber, and J. Pei, *Data mining: concepts and techniques*. Elsevier, 2011.

[62] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.

[63] L. Kaufman and P. J. Rousseeuw, *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons, 2009.

[64] N. Czink, P. Cera, J. Salo, E. Bonek, J.-P. Nuutinen, and J. Ylitalo, "Automatic clustering of MIMO channel parameters using the multipath component distance measure," *WPMC'05*, 2005.

Ruisi He (S'11-M'13) received the B.E. and Ph.D. degrees from Beijing Jiaotong University, Beijing, China, in 2009 and 2015, respectively.

Since 2015, Dr. He has been an Associate Professor with the State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing, China. Dr. He has been a Visiting Scholar in Georgia Institute of Technology, USA, University of Southern California, USA, and Université Catholique de Louvain, Belgium. His research interests include measurement and modeling of wireless channels, machine learning and clustering analysis in communications, vehicular and high-speed railway communications, 5G massive MIMO and high frequency communication techniques. He has authored/co-authored 2 books, 3 book chapters, more than 100 journal and conference papers, as well as several patents.

Dr. He is an editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and a Lead Guest Editor of the IEEE ACCESS. He serves as the Early Career Representative (ECR) of Commission C, International Union of Radio Science (URSI). He has been a Technical Program Committee (TPC) chair and member for many conferences and workshops. He received the Second Prize of the Natural Science Award for Scientific Research Achievements of the Ministry of Education in China in 2016, the Best Ph.D. Thesis Award of Chinese Institute of Electronics in 2016, the URSI Young Scientist Award in 2015, and three Best Paper Awards in conferences. He is a member of the COST.

Qingyong Li (M'12) received the B.Sc. degree in computer science and technology from Wuhan University, Wuhan, China, in 2001 and the Ph.D. degree in computer science and technology from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2006. He is currently a Professor with the Beijing Jiaotong University, Beijing. His research interests include machine learning and computer vision.

Bo Ai (M'00-SM'10) received the M.S. and Ph.D. degrees from Xidian University, Xi'an, China, in 2002 and 2004, respectively.

He was with Tsinghua University, Beijing, China, where he was an Excellent Postdoctoral Research Fellow in 2007. He is currently a Professor and an Advisor of Ph.D. candidates with Beijing Jiaotong University, Beijing, where he is also the Deputy Director of the State Key Laboratory of Rail Traffic Control and Safety. He is also currently with the Engineering College, Armed Police Force, Xi'an. He has authored or coauthored six books and 140 scientific research papers, and holds 26 invention patents in his research areas. His interests include the research and applications of orthogonal frequency-division multiplexing techniques, high-power amplifier linearization techniques, radio propagation and channel modeling, global systems for mobile communications for railway systems, and long-term evolution for railway systems.

Dr. Ai is a Fellow of The Institution of Engineering and Technology. He was as a Cochair or a Session/Track Chair for many international conferences such as the 9th International Heavy Haul Conference (2009); the 2011 IEEE International Conference on Intelligent Rail Transportation; HSRCom2011; the 2012 IEEE International Symposium on Consumer Electronics; the 2013 International Conference on Wireless, Mobile and Multimedia; IEEE Green HetNet 2013; and the IEEE 78th Vehicular Technology Conference (2014). He is an Associate Editor of IEEE TRANSACTIONS ON CONSUMER ELECTRONICS and an Editorial Committee Member of the Wireless Personal Communications journal. He has received many awards such as the Qiushi Outstanding Youth Award by HongKong Qiushi Foundation, the New Century Talents by the Chinese Ministry of Education, the Zhan Tianyou Railway Science and Technology Award by the Chinese Ministry of Railways, and the Science and Technology New Star by the Beijing Municipal Science and Technology Commission.

Yang Li-Ao Geng received the Bachelor degree in information management and information system from Zhengzhou University, Zhengzhou, China, in 2014. He is currently working toward the Ph.D. degree at the School of Computer Science and Technology, Beijing Jiaotong University, Beijing, China. His research interests include clustering, unsupervised learning, and machine learning.

Andreas F. Molisch (S'89-M'95-SM'00-F'05) received the Dipl. Ing., Ph.D., and habilitation degrees from the Technical University of Vienna, Vienna, Austria, in 1990, 1994, and 1999, respectively. He subsequently was with AT&T (Bell) Laboratories Research (USA); Lund University, Lund, Sweden, and Mitsubishi Electric Research Labs (USA). He is now a Professor of Electrical Engineering and Director of the Communication Sciences Institute at the University of Southern California, Los Angeles.

His current research interests are the measurement and modeling of mobile radio channels, ultra-wideband communications and localization, cooperative communications, multiple-input-multiple-output systems, wireless systems for healthcare, and novel cellular architectures. He has authored, coauthored, or edited four books (among them the textbook Wireless Communications, Wiley-IEEE Press), 19 book chapters, more than 200 journal papers, some 300 conference papers, as well as more than 80 patents and 70 standards contributions.

Dr. Molisch has been an Editor of a number of journals and special issues, General Chair, Technical Program Committee Chair, or Symposium Chair of multiple international conferences, as well as Chairman of various international standardization groups. He is a Fellow of the National Academy of Inventors, Fellow of the AAAS, Fellow of the IEEE, Fellow of the IET, an IEEE Distinguished Lecturer, and a member of the Austrian Academy of Sciences. He has received numerous awards, among them the Donald Fink Prize of the IEEE, and the Eric Sumner Award of the IEEE.

Vinod Kristem received his Bachelor of Technology degree in Electronics and Communications Engineering from the National Institute of Technology (NIT), Warangal in 2007, Master of Engineering degree in Telecommunications from the Dept. of Electrical Communication Engineering, Indian Institute of Science, Bangalore, India in 2009 and Ph.D. degree in Electrical Engineering from the University of Southern California, Los Angeles in 2017. From 2009–2011, he was with Beceem Communications Pvt. Ltd., Bangalore, India (acquired by Broadcom Corp.), where he worked on channel estimation and physical layer measurements for WiMAX and LTE. He is currently working as a research scientist at Intel Corporation. His current research interests include the multi-antenna systems, Ultrawideband systems, channel measurements and modeling.

Zhangdui Zhong received the B.E. and M.S. degrees from Beijing Jiaotong University, Beijing, China, in 1983 and 1988, respectively. He is a Professor and Advisor of Ph.D. candidates with Beijing Jiaotong University, Beijing, China. He is currently a Chief Scientist of State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, a Director of the Innovative Research Team of Ministry of Education, Beijing, and a Chief Scientist of Ministry of Railways, Beijing. He is an Executive Council Member of Radio Association of China, Beijing, and a Deputy Director of Radio Association, Beijing. He was a Director of the School of Computer and Information Technology, Beijing Jiaotong University. His interests include wireless communications for railways, control theory and techniques for railways, and GSM-R systems. His research has been widely used in railway engineering, such as Qinghai-Xizang railway, Datong-Qinhuangdao Heavy Haul railway, and many high-speed railway lines in China.

He has authored or coauthored seven books, five invention patents, and over 200 scientific research papers in his research area. He received the MaoYiSheng Scientific Award of China, ZhanTianYou Railway Honorary Award of China, and Top 10 Science/Technology Achievements Award of Chinese Universities.

Jian Yu received the B.S., M.S., and Ph.D. degree in mathematics from Peking University of China in 1991, 1994, and 2000, respectively. He is currently a professor in the School of Computer and Information Technology of Beijing Jiaotong University, the Chairman of the Department of Computer Science, and Director of Beijing Key Lab of Traffic Data Analysis and Mining. He has published the book "Machine LearningFrom Axioms to Algorithms". His research interests include machine learning, data mining, and natural language processing.