

Research Article

Hybrid DE-EM Algorithm for Gaussian Mixture Model-Based Wireless Channel Multipath Clustering

Yupeng Li,¹ Jianhua Zhang ,² Ruisi He ,³ Lei Tian,² and Hewen Wei⁴

¹Key Lab of Universal Wireless Communications, Beijing University of Posts and Telecommunications, Beijing, China

²State Key Lab of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China

³State Key Lab of Rail Traffic and Safety, Beijing Jiaotong University, Beijing, China

⁴National Key Laboratory of Science and Technology on Blind Signal Processing, Chengdu, Sichuan, China

Correspondence should be addressed to Jianhua Zhang; jhzhang@bupt.edu.cn

Received 12 October 2019; Accepted 9 December 2019; Published 27 December 2019

Academic Editor: Francisco Falcone

Copyright © 2019 Yupeng Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, the Gaussian mixture model (GMM) is introduced to the channel multipath clustering. In the GMM field, the expectation-maximization (EM) algorithm is usually utilized to estimate the model parameters. However, the EM widely converges into local optimization. To address this issue, a hybrid differential evolution (DE) and EM (DE-EM) algorithms are proposed in this paper. To be specific, the DE is employed to initialize the GMM parameters. Then, the parameters are estimated with the EM algorithm. Thanks to the global searching ability of DE, the proposed hybrid DE-EM algorithm is more likely to obtain the global optimization. Simulations demonstrate that our proposed DE-EM clustering algorithm can significantly improve the clustering performance.

1. Introduction

The channel model is of great significance for the wireless communication system simulations and technology evaluations [1–3]. Meanwhile, the channel model standards, such as the 3GPP TR 36.873 [4] and 3GPP TR 38.900 [5], are always cluster-based. In [6], clustering is a necessary signal processing procedure to find the cluster-nuclei. Thus, the clustering plays an important role in the channel modeling, and a clustering method which can grasp the channel characteristics will significantly improve the precision of the channel model. Additionally, channel clustering also has significant impacts on estimating the channel capacity [7]. Therefore, a well-designed clustering method for channel modeling is necessary.

In three-dimensional (3D) multiple-input multiple-output (MIMO) channels, adjacent multipaths have similar parameters and always present correlations with each other. Therefore, the cluster of channel multipaths is defined as a set of multipath components (MPCs) with similar parameters consisting of the elevation angle of departure (EOD),

the azimuth angle of departure (AOD), the elevation angle of arrival (EOA), the azimuth angle of arrival (AOA), the delay (τ) [8], etc.

There are many clustering algorithms [8] used for MPC clustering. In the K means field, a new initialization method is proposed in [9]. Hu et al. [10] embed the K means algorithm into an improved differential evolution (DE) method to improve the global searching ability. In [11, 12], DE combined with modified K means is proposed to improve the clustering precision. However, all the above-mentioned clustering algorithms do not consider sufficient statistical information of the MPCs. To this end, we resort to the Gaussian mixture model (GMM) [13, 14], which delivers not only the mean information of channel MPCs but also their distributions. The GMM is constituted by taking a weight summation of several Gaussian density functions, where each function models the channel MPCs within the same cluster and each weight represents the corresponding prior probability of the cluster. Consequently, in this paper, the multipath parameters can be seen as generated from several Gaussian distributions. Moreover, it has been proved

that the GMM can approximate continuous distributions with high accuracy [15]. In this article, we employ the GMM to implement channel MPC clustering.

The expectation-maximization (EM) algorithm is a usually used tool to estimate the GMM parameter set [16]. It maximizes the log-likelihood value of observed dataset, given the model via iterating between maximizing the log-likelihood value (called as M-step) and calculating the log-likelihood expectation (called as E-step), whereas the EM algorithm is a deterministic optimization technique, which can reach merely a local optimization depending on the initialization heavily. The initialization will affect not only the estimation accuracy but also convergence speed. Numerous endeavors have been tried to optimize the initialization of EM [17]. A large number of local models are firstly identified in [18], and it selects the most separated ones. In [19], the TRUST-TECH technique is used to estimate adjacent local maximum values of the log-likelihood. Then, Nasios and Bors [20] adopted a hierarchical maximum initialization to initialize the hyperparameters. Constantinopoulos and Likas [21] reduce the influence of initialization with a deterministically incremental estimation algorithm. To mitigate the local convergence characteristics of EM, a split-and-merge EM method is proposed in [22]. However, the above-mentioned modifications can hardly avoid being trapped into local optimizations. A maximum likelihood (ML) estimator would solve the global search problem, while its computing complexity may be much higher.

Alternatively, compared with EM, the DE [23] possesses relative stronger global searching ability. The DE searches the space by virtue of generating a population of individuals, and each individual has its own fitness value based on a cost function. The highest fitness value in the population improves gradually with the iteration [24]. The population-based stochastic search of DE enables a more thorough exploration of the searching space. As the DE algorithm embraces the advantage of simple structure, robustness, and efficiency, it has been considered as one of the most popular evolutionary algorithms [25, 26].

To get a preferable parameter set of the GMM, we embed the EM algorithm in the DE framework and propose a hybrid DE-EM algorithm. The DE algorithm is employed to initialize the EM parameters and enhance the global searching ability of EM. The EM algorithm is utilized to estimate the GMM parameter set. To be specific, the stochastic searching of DE enables searching the space more thoroughly than conventional EM, which enables DE-EM jump out of local optimization. Therefore, through the iteration of DE, the parameter estimation values draw near the global optimization region. Moreover, the EM algorithm can accelerate the convergence when the current solution draws near the global optimal region, i.e., the EM guides the individuals towards the promising locations. This means that the DE combining EM framework can take the advantages from both algorithms. Experiments have been carried out with synthetic datasets and indoor channel measurement data. In both the visual and quantitative aspects, the DE-EM shows improved performance compared with the EM.

The rest of the work is organized as follows. In Section 2, the GMM and its EM solution are presented. In Section 3, we present the proposed DE-EM algorithms. Simulations are presented to verify the favorable searching ability of the DE-EM algorithm in Section 4. In Section 5, the paper is concluded.

We use the following notations in this paper: the bold symbols are reserved for vectors and matrices. \mathbf{B}^T is the transpose of \mathbf{B} .

2. The GMM and Its Solution with the EM

In this article, the GMM is employed to model the channel MPCs. The channel MPCs are represented by a group of d -dimensional vectors $\mathbf{X} = \{\mathbf{x}_i; i = 1, 2, \dots, N\}$, where N represents the number of samples and $\mathbf{x}_i \in C^d$ represents the i th MPC parameters of the channel. In this work, the MPC parameters we use are $[\phi_{Rx}, \phi_{Tx}, \theta_{Rx}, \theta_{Tx}, \tau]$, where $\phi_{Rx}, \phi_{Tx}, \theta_{Rx}, \theta_{Tx}, \tau$ are, respectively, the EOA, EOD, AOA, AOD, and delay of the MPC. In accordance with GMM, each \mathbf{x}_i is drawn independently from one of the L Gaussian components, and each of the components fits the channel MPCs as follows:

$$p(\mathbf{x}_i; \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l) = \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}_l|} \times \exp\left(\frac{(\mathbf{x}_i - \boldsymbol{\mu}_l)^T \boldsymbol{\Sigma}_l^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_l)}{-2}\right), \quad (1)$$

where $l = 1, 2, \dots, L$, L is the number of components, $\boldsymbol{\mu}_l$ is the mean, and $\boldsymbol{\Sigma}_l$ is the covariance matrix.

For the GMM, the log-likelihood value of the channel MPCs can be calculated as follows [13, 14]:

$$L(X; \lambda_l, \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l) = \sum_{i=1}^N \log \sum_{l=1}^L \lambda_l p(\mathbf{x}_i; \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l), \quad (2)$$

where the mixing coefficient $\lambda_l \in [0, 1]$ represents the prior probability of MPCs derived from cluster l and satisfies $\sum_{l=1}^L \lambda_l = 1$. The ML estimation of the GMM parameters can be obtained through the EM algorithm [27], where the iteration is implemented between posterior probability computing with (3) and parameters updating with (4)–(6) [28], as follows:

$$\gamma_{i,l} = \frac{\lambda_l p(\mathbf{x}_i | \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)}{\sum_{l=1}^L \lambda_l p(\mathbf{x}_i | \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)}, \quad (3)$$

$$\lambda_l = \frac{1}{N} \sum_{i=1}^N \gamma_{i,l}, \quad (4)$$

$$\boldsymbol{\mu}_l = \frac{\sum_{i=1}^N \gamma_{i,l} \mathbf{x}_i}{\sum_{i=1}^N \gamma_{i,l}}, \quad (5)$$

$$\boldsymbol{\Sigma}_l = \frac{\sum_{i=1}^N \gamma_{i,l} (\mathbf{x}_i - \boldsymbol{\mu}_l) (\mathbf{x}_i - \boldsymbol{\mu}_l)^T}{\sum_{i=1}^N \gamma_{i,l}}. \quad (6)$$

The detailed derivation is given in [13, 14]. The calculation of $\gamma_{i,l}$ is generally considered as the E-step. With

exactly known $\gamma_{i,l}$, the parameters of GMM are updated in the M-step.

We can see that function (2) is nonconvex and thus has several local maxima in the searching space. As the EM algorithm inclines to fall into local maxima and the DE has a strong global searching ability, we embed the EM into the DE framework in Section 3.

3. Differential-Based EM Algorithm

In the DE-EM algorithm, we encode each individual in the population as a parameter set of GMM, that is, a set of parameters $\Theta = \{\lambda_l, \mu_l, \Sigma_l\}$ that models a GMM. The DE is employed to initialize the EM parameters, and the EM is used to estimate the GMM parameters involved in each individual. The log-likelihood (i.e., function (2)) is used as the fitness function. Thereby, the individual with the highest log-likelihood value is the best one in our work. The DE is then employed to generate offspring individuals. After mutation, crossover, and selection, the selected individuals are resubmitted to EM to update the GMM parameters. The two stages repeat until the termination conditions are satisfied.

Because of the less sensitivity to the initialization and exploring the searching space more thoroughly, the DE algorithm can jump out of local optimal solutions with high probability. Compared with the DE, the EM algorithm converges faster when the current solution draws near the optimal solution and further obtains high convergence accuracy, i.e., the EM leads the individuals towards the promising locations. The main purpose of combining EM with DE is to take the advantage of both algorithms. This means that, after proper parameter setting, the DE-EM can enhance the exploitation of the searching space with higher convergence rate.

Because of the memory characteristics of the DE, the DE-EM maintains the monotonic convergence property. This principle guarantees that the best individual does not decrease with the increase of generation. The iteration of DE-EM is terminated if the log-likelihood of the best individual remains stable within several successive iterations.

The framework of DE-EM algorithm is illustrated in Table 1, where the best individual is stored in \mathbf{a}_{\max} and the corresponding evaluation value is in F_{\max} . The evaluation of the individuals is twofold. Firstly, EM cycles are implemented on all the individuals at the generation T , resulting in an update of the parameter set (Θ^T). Secondly, for all the updated individuals, the fitness value is determined by function (2). Therefore, the evaluation process of the individuals

TABLE 1: Operation procedure of the DE-EM.

```

Begin:
  Initializing the generation  $T$ :  $T \leftarrow 0$ 
  Initializing the termination factor  $c_{\text{end}}$ :  $c_{\text{end}} \leftarrow 0$ 
   $P(T) \leftarrow$  randomly initializing the population with
  Encoded GMM parameters
   $P(T)' \leftarrow$  EM steps  $P(T)$ 
   $F \leftarrow$  evaluate  $P(T)'$ 
  While ( $c_{\text{end}} \neq 7$ ):
     $P(T)'' \leftarrow$  crossover  $P(T)'$ 
     $P(T)''' \leftarrow$  EM steps  $P(T)''$ 
     $F' \leftarrow$  evaluate  $P(T)'''$ 
     $[P(T)'''' , F''] \leftarrow$  select  $[(P(T)'''' , F') \cup P(T)', F]$ 
    If  $\max(F'') > F_{\max}$  then
       $F_{\max} \leftarrow \max(F'')$ 
       $\mathbf{a}_{\max} \leftarrow \text{argmax}_F(P(T)'''' )$ 
       $c_{\text{end}} \leftarrow 0$ 
    Else
       $c_{\text{end}} \leftarrow c_{\text{end}} + 1$ 
    End
     $P(T+1) \leftarrow$  mutate  $P(T)''''$ 
     $T \leftarrow T + 1$ 
  End
  EM ( $\mathbf{a}_{\max}$ ) until convergence of the log-likelihood
  is reached
End

```

implements an update of the parameters and their corresponding fitness values. We use $P(T)$ to represent s individuals at generation T . Implementing EM cycles of $P(T)$ generates $P(T)'$. Implementing crossover of $P(T)'$ generates the offspring population called as $P(T)''$. Implementing the EM cycles of $P(T)''$ and evaluating of the off-spring population deliver $P(T)'''$ and F' . Selecting the best 20 individuals from both the parent and offspring population [23], we get $P(T)''''$ and its corresponding values F'' . Performing mutation of $P(T)''''$, we get the $P(T+1)$ for the next DE cycle. The termination factor c_{end} is set to 7, i.e., the DE-EM terminates if the best fitness value is not enhanced within 7 successive generations. The operation procedure is shown in Table 1.

The operation procedure of DE-EM is discussed in the following with more details.

3.1. Encoding. The mean value μ_l and the covariance matrix Σ_l of L components are encoded with floating point value, i.e., they are represented in the continuous space. Since we choose the diagonal matrix as the covariance matrix, the number of values $L = 2 * d$ is used to encode each mixture component. An individual \mathbf{y}_i which encodes a GMM is given as follows:

$$\mathbf{y}_i = \left(\overbrace{\mu_{i,1}^{l=1}, \dots, \mu_{i,d}^{l=1}, \sigma_{i,1}^{l=1}, \dots, \sigma_{i,d}^{l=1}}^{\text{component 1}}, \overbrace{\mu_{i,1}^{l=2}, \dots, \mu_{i,d}^{l=2}, \sigma_{i,1}^{l=2}, \dots, \sigma_{i,d}^{l=2}}^{\text{component 2}}, \dots, \overbrace{\mu_{i,1}^{l=L}, \dots, \mu_{i,d}^{l=L}, \sigma_{i,1}^{l=L}, \dots, \sigma_{i,d}^{l=L}}^{\text{component L}} \right), \quad (7)$$

where $[\mu_{i,1}^{l=1}, \dots, \mu_{i,d}^{l=1}]$ is the mean vector of component l and diagonal $(\sigma_{i,1}^{l=1}, \dots, \sigma_{i,d}^{l=1})$ is the covariance matrix of component l . One thing to mention is that, as diagonal matrix is sufficient to

fit the channel data in this paper, we use diagonal matrix as the covariance matrix. If we want to fit the dataset with full rank matrix, we just need to re-encode the individuals \mathbf{y}_i .

3.2. *Crossover.* Crossover is employed to each pair of mutant $\mathbf{U}_{j,p,T}$ and target vector $\mathbf{Y}_{j,p,T}$ to generate a trial vector $\mathbf{G}_{j,T} = [\mathcal{g}_{j,1,T}, \mathcal{g}_{j,2,T}, \dots, \mathcal{g}_{j,D,T}]$. The binomial crossover, which is commonly considered, is shown as follows:

$$\mathcal{g}_{j,p,T} = \begin{cases} u_{j,p,T}, & \text{rand}_p(0, 1) \leq R \text{ or } p = p_{\text{rand}}, \\ y_{j,p,T}, & \text{otherwise,} \end{cases} \quad (8)$$

where the crossover factor R is prespecified, which balances the proportion copied from the target vector. Parameter p_{rand} is a constant within the range $[0, 1]$. If $\text{rand}_p(0, 1) \leq R$ or $p = p_{\text{rand}}$, the j th parameter of the mutant vector $\mathbf{U}_{j,T}$ enters the corresponding element in the trial vector $\mathbf{G}_{j,T}$. Otherwise, the target vector $\mathbf{Y}_{j,T}$ enters. The rest of parameters for the trial vector $\mathbf{G}_{j,T}$ are constructed by the target vector $\mathbf{Y}_{j,T}$. We use the condition $p = p_{\text{rand}}$ to guarantee the trial vector $\mathbf{G}_{i,T}$ differs from the target vector $\mathbf{Y}_{i,T}$ by at least one parameter. If some parameters of a new trial vector exceed their lower or upper constraints, we reinitialize them into a prespecified range uniformly. After that, all the trial vectors are evaluated by their fitness values.

3.3. *Selection.* In the selection process, the comparison of the fitness values is made between target vector $f(\mathbf{Y}_{j,T})$ and its corresponding trial vector $f(\mathbf{G}_{j,T})$. If the target vector is smaller than the corresponding trial vector by the fitness value, the target vector will be replaced by the trial vector. Otherwise, the target vector remains in the population. The selection can be expressed as follows:

$$\mathbf{Y}_{T+1} = \begin{cases} \mathbf{Y}_{j,T}, & f(\mathbf{G}_{j,T}) < f(\mathbf{Y}_{j,T}), \\ \mathbf{G}_{j,T}, & \text{otherwise.} \end{cases} \quad (9)$$

3.4. *Mutation.* This operator creates a mutant vector $\mathbf{U}_{j,T}$. The mutant vector is also called as differential vector, where the name of DE comes from. After the crossover and selection, the DE produces a population, called as target vectors. For each target vector $\mathbf{Y}_{j,T}$ at the generation T , its related mutant vector $\mathbf{U}_{j,T} = [u_{j,T}^1, u_{j,T}^2, \dots, u_{j,T}^d]$ can be generated via making a difference between several target vectors. Specifically, several commonly used difference strategies [6, 7] are as follows:

(1) DE/rand/1:

$$\mathbf{U}_{j,T} = \mathbf{Y}_{j_1,T} + Q \cdot (\mathbf{Y}_{j_2,T} - \mathbf{Y}_{j_3,T}). \quad (10)$$

(2) DE/best/1:

$$\mathbf{U}_{j,T} = \mathbf{Y}_{\text{best},T} + Q \cdot (\mathbf{Y}_{j_1,T} - \mathbf{Y}_{j_2,T}). \quad (11)$$

(3) DE/rand-to-best/1:

$$\mathbf{U}_{j,T} = \mathbf{Y}_{j,T} + Q \cdot (\mathbf{Y}_{\text{best},T} - \mathbf{Y}_{j_1,T}) + Q \cdot (\mathbf{Y}_{j_1,T} - \mathbf{Y}_{j_2,T}). \quad (12)$$

(4) DE/best/2:

$$\mathbf{U}_{j,T} = \mathbf{Y}_{\text{best},T} + Q \cdot (\mathbf{Y}_{j_1,T} - \mathbf{Y}_{j_2,T}) + Q \cdot (\mathbf{Y}_{j_3,T} - \mathbf{Y}_{j_4,T}). \quad (13)$$

(5) DE/rand/2:

$$\mathbf{U}_{j,T} = \mathbf{Y}_{j_1,T} + Q \cdot (\mathbf{Y}_{j_2,T} - \mathbf{Y}_{j_3,T}) + Q \cdot (\mathbf{Y}_{j_4,T} - \mathbf{Y}_{j_5,T}). \quad (14)$$

The indices $j_1, j_2, j_3, j_4,$ and j_5 are mutually different random indices belonging to the set $1, 2, \dots, \text{NP}$. Parameter $\mathbf{Y}_{\text{best},T}$ is the best individual in the population at generation T . The scale factor $Q \in [0, 1]$ is a control parameter for changing the amplification.

3.5. *Termination Condition.* If F_{max} remains stable for several successive iterations, the DE-EM algorithm is terminated. Otherwise, set $T = T + 1$ and go to ‘‘crossover’’ process. This exhibits that the population gradually improves until the termination condition is satisfied. Finally, the DE-EM will obtain the optimal setting of the parameter set.

4. Experiments and Results

To compare the clustering performances of EM-based GMM and DE-EM-based GMM, synthetic datasets are firstly used to illuminate the DE-EM. Subsequently, experiment based on the simulated channel MPCs are employed. At last, the indoor channel measurement data are used to compare the clustering performance of the two techniques.

4.1. *Parameter Settings.* The weights $\lambda_l^{T=0}$ are assumed to be identical, i.e., $\lambda_l^{T=0} = (1/L)$ ($l = 1, \dots, L$), during the execution of the EM within the evaluation of individuals. The covariance matrix is initialized with an identity matrix. We expect that the initial components are distributed over the entire data space; therefore, the mean values $\mu_l^{T=0}$ are randomly initialized. The maximum iteration number of EM is set to 50 to avoid infinite loop. If $[L(T+1) - L(T)]/L(T) < 10^{-8}$, the iterations are terminated, where L is the log-likelihood of the dataset. The individuals of the DE-EM is set to 20, which can guarantee a relative high convergence precision with an acceptable computation complexity. There are some additional parameters to be predefined in the DE-EM: the crossover of the DE-EM is 0.8, formula (10) is chosen as the mutation strategy, and the scale factor Q is 0.8. When the log-likelihood remains stable for seven successive iterations, the DE-EM algorithm is terminated. We ran the simulation in Matlab software. The

computer configurations are i7 processor, 32 G memory, and Windows 10 system.

4.2. Experiments Based on Synthetic Datasets. We generate a synthetic datasets to validate the performances of EM and DE-EM. We use four synthetic datasets in 2-dimensional space, and 800 data points are generated from four Gaussian distributions. The means and variances are as follows:

$$\begin{aligned}
 \boldsymbol{\mu}_1 &= \boldsymbol{\mu}_2 = [-4, -4]^T, \\
 \boldsymbol{\mu}_3 &= [-1, -6]^T, \\
 \boldsymbol{\mu}_4 &= [2, 2]^T, \\
 \boldsymbol{\Sigma}_1 &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \\
 \boldsymbol{\Sigma}_2 &= \begin{bmatrix} 6 & 0 \\ 0 & 6 \end{bmatrix}, \\
 \boldsymbol{\Sigma}_3 &= \begin{bmatrix} 0.12 & 0 \\ 0 & 0.12 \end{bmatrix}, \\
 \boldsymbol{\Sigma}_4 &= \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix},
 \end{aligned} \tag{15}$$

where the first two components share the same mean value. Through this way, we construct the dataset with overlapping Gaussian distributions. Searching the complicated statistic information in the dataset can highlight the prominent global searching ability of DE-EM. The original dataset and the clustering performances of the DE-EM and EM are shown in Figure 1. In the figure, different clusters are represented by different color dots. From the figure, we can see that EM gets chaotic clustering results, and it cannot grasp the distribution characteristics of the original dataset. By contrast, the DE-EM clustering can grasp the distribution characteristics of the primary dataset, then recurring most of the primary dataset. From the simulation results, we can see that the EM falls into local optimization with a high probability, resulting in poor clustering results. However, the population-based of DE searches the space with the aid of a population of individuals, and then the individuals move towards the optimization by selection, crossover, and mutation operations. The crossover and mutation strategies of DE also enable the DE-EM jump out of local minima. Besides, it has a higher ability to discover distribution and hidden mean value information of the dataset. The same situation usually appears in the real channel MPC data. The experiment based on synthetic data can give us an intuitive visualization result.

The experiment on synthetic dataset illustrates that DE-EM can grasp the distribution characteristics of the dataset even with overlapping clusters. That is the real situation for the channel MPC parameters. In Section 4.3, we will use simulated channel MPCs to validate the clustering performance of the two techniques.

4.3. Experiments Based on Simulated Channel MPCs. In this section, the DE-EM and EM clustering algorithms are evaluated with the simulated channel MPCs, where the channel MPCs are generated by a channel model software (CMS) [29]. At the output, both the channel MPCs and the cluster centroids (CCs) are labeled by the cluster number. We consider indoor scenario. The carrier frequency is 3.5 GHz. The transmitter (Tx) side designates a uniform planar array (UPA) with 32 elements. The receiver (Rx) side employs an omnidirectional antenna (ODA) with 56 elements. The bandwidth is 50 MHz. The azimuth spread of arrival, elevation spread of arrival, azimuth spread of departure, and elevation spread of departure are, respectively, 25.8°, 8.4°, 30.8°, and 23.4°. As the angle spread (AS) is relative large, there may exist cluster overlapping. The number of clusters changes from 4 to 14. The number of MPCs in each cluster is fixed at 20. We evaluate the two clustering techniques by checking whether the MPCs are assigned to the correct cluster. This is simply performed by comparing the original label with the clustering results. The clustering results are shown in Table 2.

As the OCCs are recorded in advance, in the simulation, the cluster number of the clustering result is labeled by the nearest OCC. The values in Table 2 are calculated with the correct clustering MPCs divided by the whole number of MPCs. From Table 2, we can see that the percentage of correct MPCs for the DE-EM clustering is mostly higher than that of the EM algorithm. The simulation result demonstrates that the DE-EM algorithm can grasp the distribution characteristics of the channel MPCs and then get preferable clustering result.

4.4. Experiments Based on Indoor Channel Measurement Data

4.4.1. Measurement Configuration. In channel measurements, the Elektrobit Prosound Sounder is used to collect the channel information [30]. The basic parameters of the sounder are as follows: the carrier frequency is 3.5 GHz, the bandwidth is 50 MHz, the transmit power is 37 dBm, the chip frequency is 127 MHz, the code length is 40 ns, and the cycle duration is 9.28 ms. As shown in Figure 2, the UPA with 32 elements is deployed at the Tx. The ODA with 56 elements is deployed at the Rx. Figure 3 illustrates the layout of the antenna arrays at both sides. Table 3 illustrates the antenna parameters.

As is shown in Figure 3, the channel measurement is conducted in an indoor conference. The conference is 5.10 m wide, 5.85 m long, and 2.40 m high with concrete wall. There is no window in the room. There is a wooden table laying in the middle. The UPA is fixed on the middle of the ceiling. The ODA is at spot 15 in Figure 3.

4.4.2. Clustering Comparisons with Different Numbers of Clusters. Simulations are performed on the real measurement data. After using the space-alternating generalized expectation-maximization (SAGE) algorithm [31], we get 200 MPCs, which can serve as the input of the clustering algorithms. In both algorithms, the data dimension d is set as

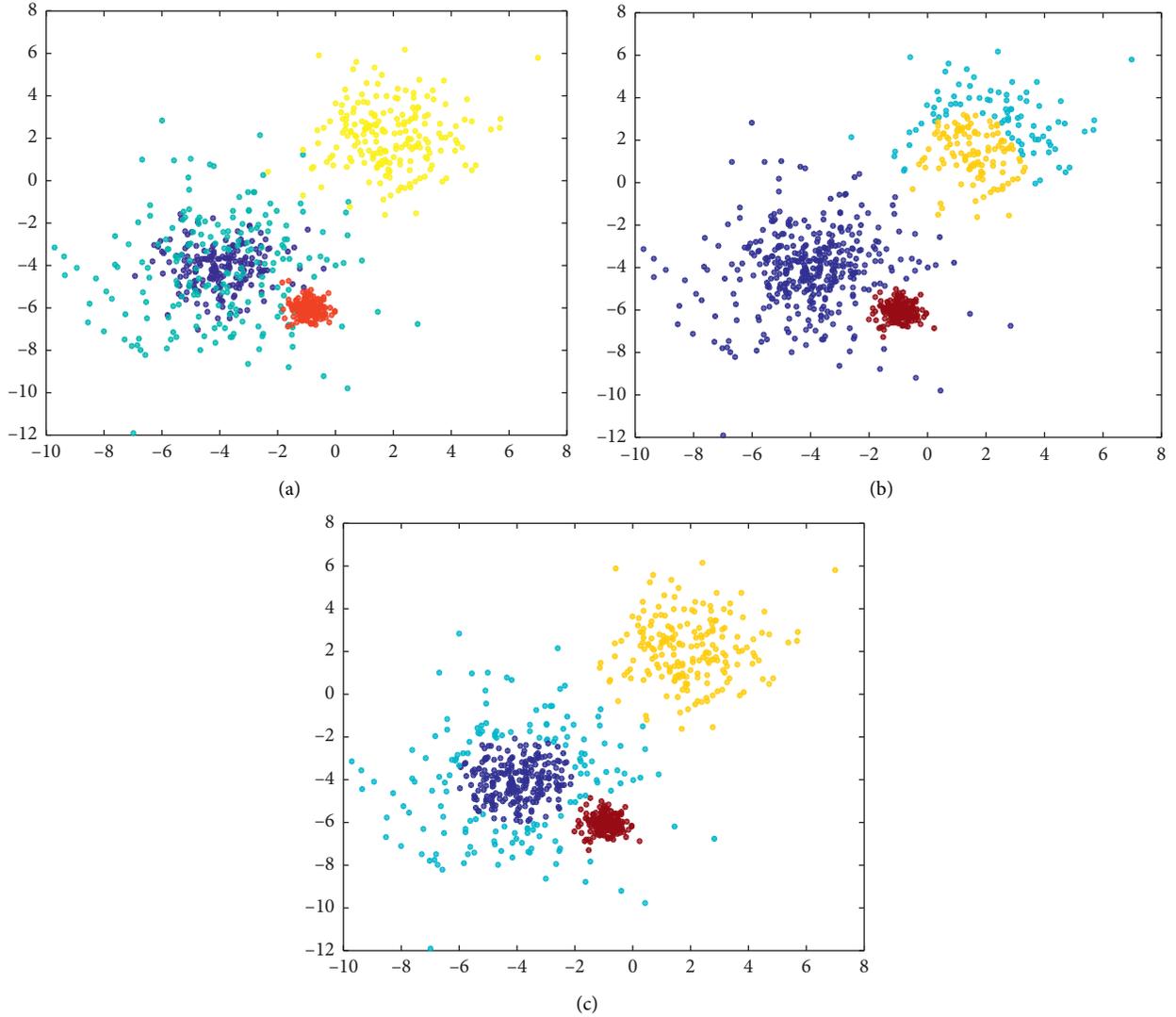


FIGURE 1: Performance comparison of the two algorithms. (a) Original distribution of the synthetic datasets; (b) EM-based clustering result and (c) DE-EM-based clustering result for the synthetic datasets.

TABLE 2: Percentage of correct MPCs.

Number of clusters	4	5	6	7	8	9	10	11	12	13	14
EM	62.5%	70%	65%	65.71%	68.71%	76.67%	74%	77.27%	75%	73.8%	72.14%
DE-EM	77.5%	82%	80%	81.43%	85%	88.89%	84%	87.27%	85.83%	87.69%	85.71%

5, i.e., $[\tau, \phi_{R_x}, \phi_{T_x}, \theta_{R_x}, \theta_{T_x}]$. The input data is arranged in column, i.e., $X = (x_1, x_2, \dots, x_N)$, where $x_i = (x_{i,1}, \dots, x_{i,d})^T$ represents the i th ($i = 1, 2, \dots, N$) multipath vector. The diagonal covariance matrix is designed for each Gaussian components.

Firstly, we compare the convergence log-likelihood value of the EM clustering and DE-EM clustering from 3 to 14 clusters. A higher log-likelihood value denotes a better fit of the GMM to MPCs. The results are shown in Figure 4.

Obviously, the increasing cluster numbers will contribute to the fitting degree of the GMM to the channel MPCs. Thus, the convergence log-likelihoods of the two

algorithms increase with cluster number. Mostly, the convergence log-likelihood of the DE-EM-based clustering algorithm is higher than the EM-based clustering algorithm. This distinction is due to the independence of the DE-EM on the initialization. Furthermore, the searching mechanism and the mutation operator embed the DE-EM with strong global searching ability.

4.4.3. Model Selection Criterion and Clustering Comparison with Different Snapshots. Selecting appropriate number of components usually accompanies with the following

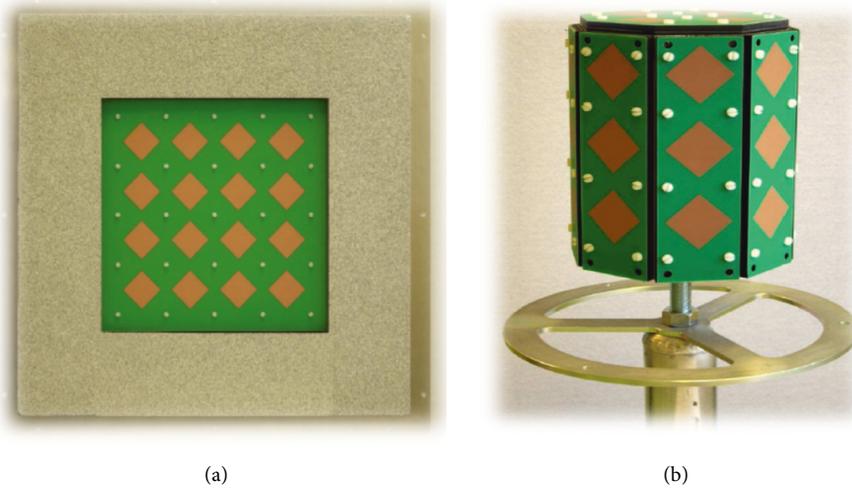


FIGURE 2: Antenna arrays. (a) TX antenna: UPA. (b) RX antenna: ODA.

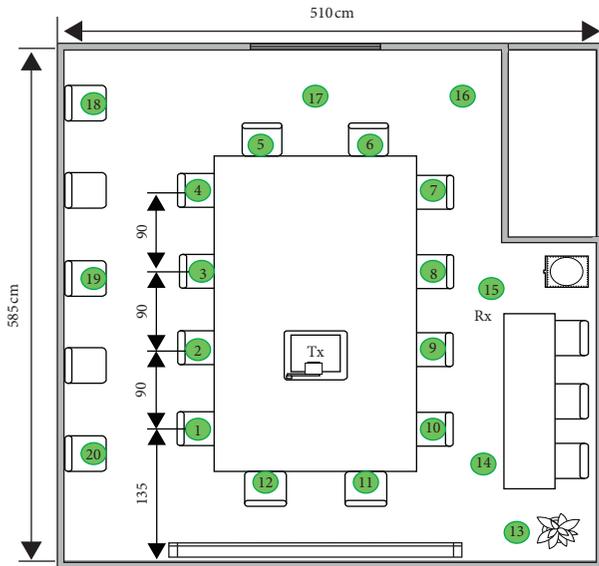


FIGURE 3: Measurement scenario.

TABLE 3: Antenna parameters.

	UPA	ODA
Number	32	56
Polarization	Dual	Dual
Space	0.5 wavelength	0.5 wavelength
Azimuth	-70° ~ 70°	-180° ~ 180°
Elevation	-70° ~ 70°	-55° ~ 90°

tradeoff: selecting too few components may result in a less accurate model of the underlying distribution and choosing too many components leads to an overfit of the data. Many different approaches have been proposed for selecting the component of the finite mixture models [32], and one of the most prominent criteria is the Bayesian Information Criterion (BIC) [33]. The model selection criterion basically

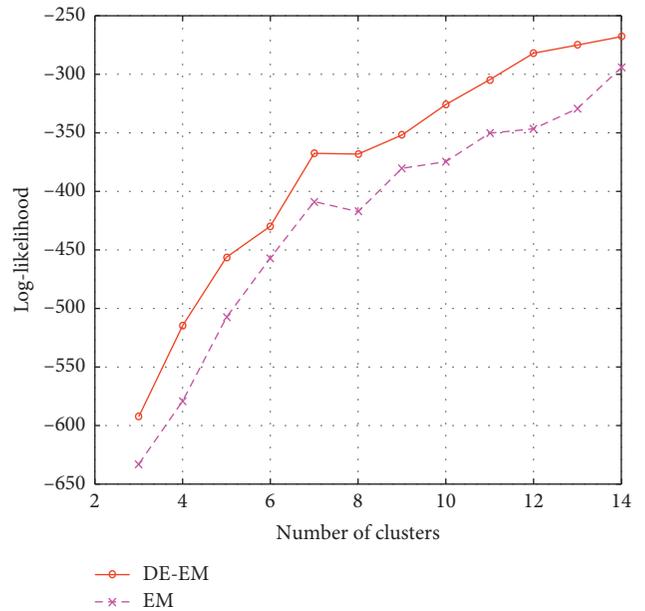


FIGURE 4: Log-likelihood with different clusters.

specifies the tradeoff between the goodness of fit and the complexity of a model. The BIC is expressed as follows:

$$BIC = -2 * \log p(\mathbf{X} | \lambda, \mu, \Sigma) + q * \log n, \quad (16)$$

where q is the number of parameters, n is the number of sampling points in \mathbf{X} , and $\log p(\mathbf{X} | \lambda, \mu, \Sigma)$ is the log-likelihood value of the GMM. A lower BIC value corresponds to a better clustering performance [33]. Then, the minimum BIC value corresponds to the best component of the mixture model. Figure 5 shows the BIC values of DE-EM and EM for different number of clusters. Then, we simulate 190 channel sampling snapshots. A total number of 50 Monte Carlo simulations are carried out, and then the largest convergence log-likelihood is selected. The convergence log-likelihood values in each snapshot are shown in Figure 6.

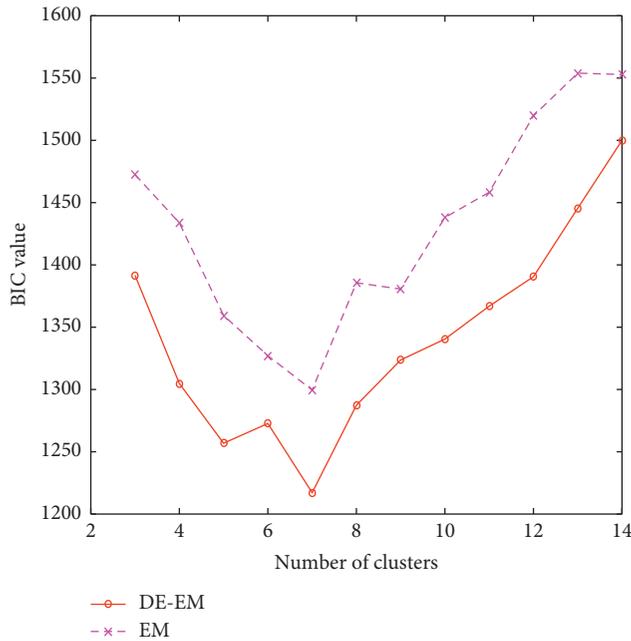


FIGURE 5: BIC value with different clusters.

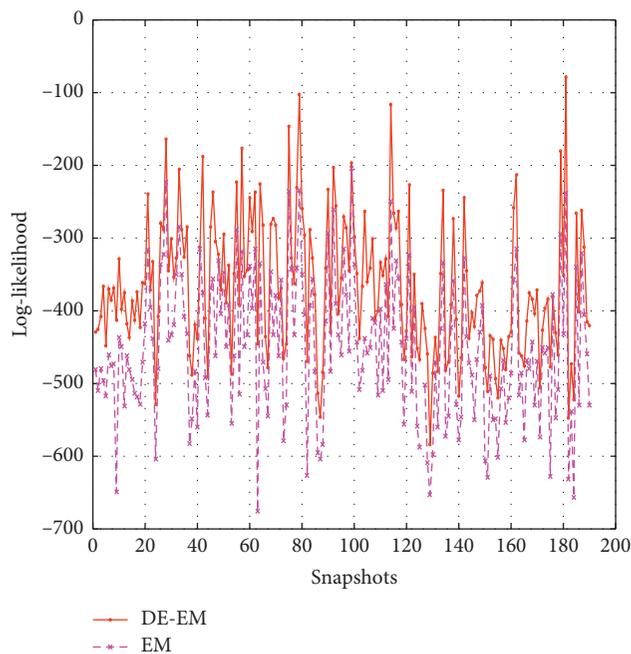


FIGURE 6: Log-likelihood in different snapshots.

From Figure 5, it is found that, for both EM and DE-EM algorithms, the best number of components for the mixture model is 7. Therefore, 7 clusters are suggested as they make an appropriate compromise between the fitness degree and the model complexity. Besides, the DE-EM algorithm achieves lower BIC values compared with the EM algorithm. The fact is again ascribed to the dependency of the EM algorithm to the initialization. The population-based searching mechanism of the DE-EM algorithm explores the searching space more thoroughly,

thus enabling the DE-EM achieve a better BIC performance. In the measurements, the transmit power of the Elektrobit Prosound Sounder changes from time to time. Besides, the environment is not static when we carry out the measurement. Therefore, both the shaking of the transmit power and the environment variation contribute to the change of the channel impulse response (CIR) in the receiving side. This will lead to the change of estimated MPC parameters among snapshots. Therefore, in Figure 6, the log-likelihood profile fluctuates reasonably. We can see that most of the log-likelihood of the DE-EM is higher than that of EM, which demonstrates that the DE-EM has a better fit to the log-likelihood function compared with the EM. In many snapshots, the EM algorithm falls into the local optimization. By contrast, the DE-EM maintains relative stable log-likelihood values and shows a better fit of the Gaussian distributions to MPCs.

5. Conclusion

In this paper, a global optimization algorithm, the hybrid DE-EM, is proposed to explore the potential space of the GMM parameter. The population-based searching of DE makes the DE-EM less sensitive to the parameter initialization compared with the conventional EM. Moreover, the use of EM into the evaluation of the DE individuals provides a parameter update method and the EM is targeted to lead the individuals towards the promising locations. Then, the hybrid DE-EM algorithm is validated by both synthetic dataset and indoor channel measurement dataset. Validation results show that the DE-EM clustering algorithm obtains more preferable clustering performance in visualization and fits the GMM to the dataset with higher precision. In general, the initialization is a common problem hindering clustering analysis and it also restricts the fit precision of the GMM to the MPCs in the channel clustering. In particular, the cost function of GMM possesses several local optima. Consequently, our algorithm can be used for GMM clustering tasks which contributes to developing the optimization procedures of GMM channel clustering. The average simulation times for EM and DE-EM are, respectively, 1.664 s and 22.445 s. That is because DE-EM needs to iterate in the DE framework, leading to consuming more running time than pure EM. Fortunately, the clustering result is used to channel modeling, which does not require high real-time. Therefore, the running time of the two algorithms is acceptable.

Data Availability

The raw data used to support the findings of this paper are supplied by BUPT-RTT2 under license and thus cannot be available freely. Requests for access to these data should be made to Professor Jianhua Zhang at jhzhang@bupt.edu.cn.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Key R&D Program of China under Grant 2018YFF0301201.

References

- [1] C. Xu, J. Zhang, M. Liu, and C. Yin, "Pilot design for sparse channel estimation in large-scale MIMO-OFDM system," *International Journal of Antennas and Propagation*, vol. 2016, Article ID 6142574, 8 pages, 2016.
- [2] Y. Yu, J. Zhang, and L. Tian, "Line-of-sight interference statistics and user scheduling with 3D angular separation," *International Journal of Antennas and Propagation*, vol. 2018, Article ID 7218471, 12 pages, 2018.
- [3] Z. Zheng, J. Zhang, X. Wu, D. Liu, and L. Tian, "Zero-forcing precoding in the measured massive MIMO downlink: how many antennas are needed," *International Journal of Antennas and Propagation*, vol. 2019, Article ID 3518691, 10 pages, 2019.
- [4] 3GPP, "Study on 3D channel for LTE (release 12)," 3GPP TR 36.873 V12.2.0 Technical Report, 3GPP, France, 2015.
- [5] 3GPP, "Study on channel model for frequency spectrum above 6 GHz (release 14)," 3GPP TR 36.900 V14.0.0 Technical Report, 3GPP, France, 2016.
- [6] J. Zhang, "The interdisciplinary research of big data and wireless channel: a cluster-nuclei based channel model," *China Communications*, vol. 13, no. 2, pp. 14–26, 2016.
- [7] K. H. Li, M. A. Ingram, and A. V. Nguyen, "Impact of clustering in statistical indoor propagation models on link capacity," *IEEE Transactions on Communications*, vol. 50, no. 4, pp. 521–523, 2002.
- [8] N. Czink, P. Cera, J. Salo et al., "Automatic clustering of MIMO channel parameters using the multi-path component distance measure," in *Proceedings of the International Symposium on Wireless Personal Multimedia Communications*, Aalborg, Denmark, 2005.
- [9] C. Huang, J. Zhang, X. Nie et al., "Cluster characteristics of wideband MIMO channel in indoor hotspot scenario at 2.35 GHz," in *Proceedings of the 2009 IEEE 70th Vehicular Technology Conference Fall*, pp. 1–5, Anchorage, AK, USA, September 2009.
- [10] J. Hu, C. Wang, C. Liu, and Z. Ye, "Improved K-means algorithm based on hybrid fruit fly optimization and differential evolution," in *Proceedings of the International Conference on Computer Science and Education (ICCSE)*, pp. 464–467, Houston, TX, USA, August 2017.
- [11] F. Wang, "A weighted K-means algorithm based on differential evolution," in *Proceedings of the IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, Xi'an, China, 2018.
- [12] Y. Tian, D. Liu, and H. Qi, "K-harmonic means data clustering with differential evolution," in *Proceedings of the International Conference on Future BioMedical Information Engineering (FBIE)*, pp. 369–372, Sanya, China, 2009.
- [13] Y. Li, J. Zhang, Z. Ma, and Y. Zhang, "Clustering analysis in the wireless propagation channel with a variational Gaussian mixture model," *IEEE Transactions on Big Data*, 2018.
- [14] Y. Li, J. Zhang, and Z. Ma, "Clustering in wireless propagation channel with a statistics-based framework," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, Barcelona, Spain, April 2018.
- [15] S. Theodoridis and K. Koutroumbas, *Pattern Recognition and Neural Networks*, The People's Posts and Telecommunications Press, Beijing, China, 2001.
- [16] W. Peng, Y. Fang, and R. Zhan, "A variable step learning algorithm for Gaussian mixture models based on the bhat-tacharyya coefficient and correlation coefficient criterion," *Neurocomputing*, vol. 239, pp. 28–38, 2017.
- [17] D. Karlis and E. Xekalaki, "Choosing initial values for the EM algorithm for finite mixtures," *Computational Statistics & Data Analysis*, vol. 41, no. 3-4, pp. 577–590, 2003.
- [18] R. Maitra, "Initializing partition-optimization algorithms," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 6, no. 1, pp. 144–157, 2009.
- [19] C. K. Reddy, H. D. Chiang, and B. Rajaratnam, "TRUST-TECH-based expectation maximization for learning finite mixture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 7, pp. 1146–1157, 2008.
- [20] N. Nasios and A. G. Bors, "Variational learning for Gaussian mixture models," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 36, no. 4, pp. 849–862, 2006.
- [21] C. Constantinopoulos and A. Likas, "Unsupervised learning of Gaussian mixtures based on variational component splitting," *IEEE Transactions on Neural Networks*, vol. 18, no. 3, pp. 745–755, 2007.
- [22] N. Ueda, R. Nakano, Z. Ghahramani, and G. E. Hinton, "SMEM algorithm for mixture models," *Neural Computation*, vol. 12, no. 9, pp. 2109–2128, 2000.
- [23] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 398–417, 2009.
- [24] R. Storn and K. Price, "Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [25] S. Das and P. N. Suganthan, "Differential evolution: a survey of the state-of-the-art," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, 2011.
- [26] F. Neri and V. Tirronen, "Recent advances in differential evolution: a survey and experimental analysis," *Artificial Intelligence Review*, vol. 33, no. 1-2, pp. 61–106, 2010.
- [27] D. G. Tzikas, A. C. Likas, and N. P. Galatsanos, "The variational approximation for Bayesian inference," *IEEE Signal Processing Magazine*, vol. 25, no. 6, pp. 131–146, 2008.
- [28] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal Processing Magazine*, vol. 13, no. 6, pp. 47–60, 1996.
- [29] 3GPP, "Study on channel model for frequencies from 0.5 to 100 GHz (release 14)," 3GPP TR 36.901 V14.1.1 Technical Report, 3GPP, France, 2017.
- [30] Y. Yu, J. Zhang, M. Shafi, P. A. Dmochowski, M. Zhang, and J. Mirza, "Measurements of 3D channel impulse response for outdoor-to-indoor scenario: capacity predictions for different antenna arrays," in *Proceedings of the IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 408–413, Hong Kong, China, 2015.
- [31] B. H. Fleury, M. Tschudin, R. Heddergott, D. Dahlhaus, and K. Ingeman 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.
- [32] G. McLachlan and D. Peel, *Finite Mixture Models*, John Wiley & Sons, Hoboken, NJ, USA, 2004.

- [33] S. C. Scott and P. S. Gopalakrishnan, "Clustering via the bayesian information criterion with applications in speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 645–648, Seattle, WA, USA, 1998.

