JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGY
http://e-journal.uum.edu.my/index.php/jict

How to cite this article:
Teologo, A., & Materum, L. (2021). An improved K-Power Means technique using minkowski distance metric and dimension weights for clustering wireless multipaths in indoor channel scenarios. *Journal of Information and Communication Technology, 20*(4), 541-563. https://doi.org/10.32890/jict2021.20.4.4

An Improved K-Power Means Technique Using Minkowski Distance Metric and Dimension Weights for Clustering Wireless Multipaths in Indoor Channel Scenarios

[1]Antipas Teologo Jr. & [2]Lawrence Materum
[1&2]Department of Electronics and Communications Engineering,
De La Salle University, Philippines
[1]Electrical and Electronics Engineering Department
FEU Institute of Technology, Philippines
[2]International Centre, Tokyo City University, Japan

antipas_jr_teologo, materuml @dlsu.edu.ph
atteologo@feutech.edu.ph

Received: 30/11/2019 Revised: 18/2/2021 Accepted: 7/3/2021 Published: 27/9/2021

ABSTRACT

Wireless multipath clustering is an important area in channel modeling, and an accurate channel model can lead to a reliable wireless environment. Finding the best technique in clustering wireless multipath is still challenging due to the radio channels' time-variant characteristics. Several clustering techniques have been developed that offer an improved performance but only consider one or two parameters of the multipath components. This study improved the K-PowerMeans technique by incorporating weights or loads based on

the principal component analysis and utilizing the Minkowski distance metric to replace the Euclidean distance. K-PowerMeans is one of the several methods in clustering wireless propagation multipaths and has been widely studied. This improved clustering technique was applied to the indoor datasets generated from the COST 2100 channel Model and considered the multipath components' angular domains and their delay. The Jaccard index was used to determine the new method's accuracy performance. The results showed a significant improvement in the clustering of the developed algorithm than the standard K-PowerMeans.

Keywords: Channel model, Minkowski distance, multipath clustering, principal component analysis, radio wave propagation.

INTRODUCTION

Channel modeling has been of great importance in mobile communications, especially in system simulations and evaluations. An accurate channel model is necessary for any wireless communication system to ensure good performance evaluation and a reliable system design. The main objective of channel modeling is to characterize the multipath components (MPCs) in different wireless environments, and there are two methodologies involved – clustered and non-clustered structure modeling (He et al., 2017). The non-clustered model characterizes the channel using the individual MPCs and has already been utilized for a long time (Chong et al., 2005; He et al., 2015; Wang et al., 2012).

For clustered structure modeling, MPCs are grouped into clusters where the intra-and inter-cluster statistics are being characterized for parameters, such as delay, number, position, and angular spreads. Much attention has been on the clustering of wireless multipaths by the research community in the past two decades. Cluster-based channel modeling has been the basis of many channel models nowadays, such as the European Cooperation in Science and Technology (COST) 259, COST 2100, 3GPP Spatial Channel Model, and the European Wireless World Initiative New Radio (WINNER) (He et al., 2017).

Accurate channel models are necessary if the goal is to achieve a reliable communication and maximum data rate, especially for wireless multiple-input multiple-output (MIMO) systems. To develop an accurate channel model, one must characterize those MPCs properly, and the clustered structure offers a significant advantage in achieving this. Many studies have ventured into finding the best way or method to group a radio channel's MPC accurately. However, there are still some challenges in terms of the automatic clustering techniques' accuracy performance because the clustering algorithm should consider all the real-world MPCs' attributes. Continuous improvement on how to accurately cluster MPC has been very evident in the studies conducted for the past two decades. Nonetheless, due to the time-variant characteristics of the radio channels, there is still a need to develop new clustering techniques that can group the wireless multipaths more precisely. Some clustering algorithms use weights or loads to their datasets to improve the clustering performance such as in the studies of Gu (2016), Huang et al. (2018), Chen et al. (2019), and Khan et al. (2020); however, further exploration on other methods are still needed.

Being one of the well-known clustering methods, K-PowerMeans (KPM) is also used in this study but is applied to the indoor datasets generated from the COST 2100 Channel Model (C2CM). The objective of this study is to improve the KPM's performance by developing a new clustering technique based on KPM's basic framework that offers higher accuracy in clustering wireless multipaths. The contributions of this study are the incorporation of weights to each dimension of the dataset based on the principal component analysis (PCA) and the utilization of the Minkowski distance as the distance metric. Minkowski distance has been used in various studies for optimization such as in Chouikhi et al. (2017), Xu et al. (2019), Khaldi et al. (2020), and Singh and Jayaram (2020).

BACKGROUND AND RELATED STUDIES

The demand for a broader bandwidth in some wireless communication systems such as in fourth generation (of cellular communications) (4G), fifth generation (5G), and MIMO systems are now increasing. Through the utilization of a cluster-based channel model, a wider

bandwidth requirement can be attained. If a cluster-based channel model is desired, parameterization of clusters' positions, number, delay, and angular spreads is essential, and this can be done through clustering of MPCs.

A cluster can be defined as a group of propagation multipaths exhibiting identical properties. Two types of multipath clustering techniques have already been introduced: manual (visual inspection of data) or automatic, and some are even a combination of manual and automatic. In the past, visual inspection has been widely used (Laurila et al., 2002; Toeltsch et al., 2002; Vuokko et al., 2005; Yu et al., 2005). However, manual inspection has limitations in grouping high-dimensional data. Due to this, automatic clustering has been developed for better channel modeling. Unfortunately, developing a more accurate and much efficient clustering technique is still challenging and in need of more research. Some of these algorithms are K-Means, Variational Gaussian Mixture Model (GMM), K-PowerMeans (KPM) framework, Ant Colony Clustering (ACC), Kernel Power Density (KPD)-based algorithm, and Kurtosis Measure (KuM)-based algorithm.

One of the most popular methods is K-Means. K-Means algorithm is a hard partitional approach that directly divides data objects into some pre-specified number of clusters (Xu & Wunsch, 2005). Typically, K-Means is utilized with a Euclidean metric to determine the distance between points and cluster centers, making it easy to determine spherical or ball-shaped clusters in the data. Although it is a popular method, initialization of the number of clusters is needed in K-Means. To solve this problem, an improvement was made, thus paving the way to the introduction of K-PowerMeans in various studies (Gustafson et al., 2014; Hanpinitsak et al., 2017; Li et al., Mota et al., 2013; Zhang, 2018). The KPM algorithm incorporates the power of MPCs, which makes it different from K-Means.

In the study of Mota et al. (2013), they utilized KPM with a different initialization procedure to cluster synthetic data generated from the Saleh-Valenzuela (SV) model. By considering the various parameters such as delay, azimuth, and power, it was found out that Xie-Beni (XB) and D53 (Dunn's) indices presented the best results with almost equal performances. In the study of Gustafson et al. (2014), the KPM

algorithm was used by taking the multipath component distance as a distance metric in parameter space as being applied in the 60 GHz Channel Model. It was discovered that the cluster peak power variation around the mean could be appropriately modeled using a log-normal distribution. In another study, Hanpinitsak et al. (2017) used the KPM framework to cluster data from the MIMO channel indoor environment at 11 GHz. The geometry of the scattering points (SPs) measured from the ray tracer was exploited and used by the KPM framework for clustering. The results indicated that the proposed method had higher performance than the conventional KPM in terms of characterization of the channel and had less complexity. Recently, Li et al. (2018) also employed KPM for MPCs clustering, space alternating generalized expectation-maximization (SAGE) algorithm to estimate MPCs, and multipath component distance-based algorithm for tracking. This new method was called the hybrid clustering approach. Using the MIMO channel model for the subway station scenario, this novel approach showed an effective way of clustering MPCs and capturing all the characteristics of the clusters. These studies used different datasets and utilized various validation indices to evaluate their clustering performance.

METHODOLOGY

The COST 2100 Indoor Datasets

The COST 2100 Channel Model (C2CM) was used to generate the indoor datasets utilized in this study. The C2CM is a Geometry-Based Stochastic Channel Modeling (GBSCM), which assumes that a wideband propagation channel can be described through the direction and delay domains at the receiving station and transmitting station sides with physical clusters. These physical clusters are groups of MPCs (Verdone & Zanella, 2012). C2CM has a MATLAB implementation and can support indoor and semi-urban channel scenarios representing single-link and multiple MIMO channel access links. In this study, there were two datasets generated from C2CM representing the indoor channel scenarios, which are as follows:

1. Indoor, B1, line-of-sight, single link (channel scenario 1).
2. Indoor, B2, line-of-sight, single link (channel scenario 2).

B1 and B2 refer to band 1 and band 2, respectively. The indoor environments were generated at 5.3 GHz band. Each dataset contained 30 different sets of data pertaining to the 30 trials performed. The datasets were in Excel file format and can be found from the IEEE DataPort. Before being utilized by the clustering technique, these datasets underwent several pre-processes such as directional cosine transform (DCT), clusterability, and whitening transform (WT). DCT is used to overcome the problem of the circular nature of the angular domain. Clusterability, on the other hand, checks the suitability for clustering of the transformed dataset. Furthermore, to standardize the dataset, WT is applied. A more detailed discussion of each pre-processing can be found in Blanza et al. (2019).

The Framework of K-Power Means

The primary basis of the newly developed clustering algorithm in this study is the K-PowerMeans (KPM). KPM is an unsupervised learning algorithm that requires that the clusters' initialization to be known a priori. Below are the main steps needed (He et al., 2017):

1. Initialize randomly K cluster centroids $\mu_1, \mu_2, \ldots, \mu_K$ wherein the positions of K centroid are chosen to be as independent events from the dataset Φ.
2. Assign every sample x of MPC to a particular cluster centroid μ_j: for every set x, as defined in Equation 1.

$$c^{(k)} := \underset{j}{\operatorname{argmin}} \left\{ \alpha_x \cdot d_{\mathrm{MPC}}\left(x, \mu_j^{(k)}\right) \right\} \tag{1}$$

where superscript (k) indicates the k-th iteration and c serves as the store indices in the k-th iteration of MPC clustering.

3. Update the cluster centroids: for each j, set as in Equation 2

$$\mu_j^{(k+1)} := \frac{\sum_{x \in \Phi} 1\{c^{(k)} = j\} \alpha_x \cdot x}{\sum_{x \in \Phi} 1\{c^{(k)} = j\} \alpha_x} \tag{2}$$

4. Perform steps 2 and 3 again until the data have converged. The addition of power weighting to determine the MPC distance, d_{MPC}, is the one that sets KPM to be different from the standard K-Means.

Minkowski Distance Metric

The Minkowski distance (MD) is the distance measurement between two points in the normal vector space. Given two points P_1 and P_2 in N-dimensional space, with $P_1 = (x_1, x_2, \ldots, x_N)$ and $P_2 = (y_1, y_2, \ldots, y_N)$, the Minkowski distance between these two points is given in Equation 3 (Minkowski distance, n.d.).

$$d_{\mathrm{MD}} = \sqrt[p]{(x_1 - y_1)^p + (x_2 - y_2)^p + \ldots + (x_N - y_N)^p} \quad (3)$$

MD is a generalization of the Euclidean distance and the Manhattan distance. A p-value of 2 makes MD equal to the Euclidean distance, and when $p = 1$, MD is just the same as the Manhattan distance.

Principal Component Analysis

The principal component analysis (PCA) summarizes the whole dataset containing various observations through the inter-correlated variables or dimensions. PCA is also one way of visualizing the information in a dataset (Kassambara, 2017).

PCA is utilized to extract the essential information from a range of multivariate data and convert it to principal components that consist of a new set of variables. This new set of variables or dimensions are just the linear combination of the original dataset. PCA's purpose is to determine the different principal components or directions in which there is a maximum variation of data.

In the MATLAB implementation of PCA, the first parameters to obtain are the principal component coefficients known as loadings. These coefficients are represented by the matrix coefficient $\boldsymbol{C}$ in Eq. (4). Given an m-by-n data matrix of X, where m is the number of observations and n is the number of variables or dimensions, the coefficient matrix $\boldsymbol{C}$ produced is n-by-n. Each column of $\boldsymbol{C}$contains coefficients for one principal component. The first column C_{11} represents the first principal component, while column C_{1n} gives the

n-th and the last principal component. These columns are arranged in decreasing order of component variance. On the other hand, each row of $\boldsymbol{C}$ represents the variables or dimension starting from row one as the variable one until the last row for the *n*-th variable as in Equation 4.

$$\boldsymbol{C} = \begin{bmatrix} C_{11} & \cdots & C_{1n} \\ \vdots & \ddots & \vdots \\ C_{n1} & \cdots & C_{nn} \end{bmatrix} \tag{4}$$

Other critical parameters in PCA are the latent values represented by $\boldsymbol{l}$ given in Equation 5, equivalent to the principal component variances. These variances are the eigenvalues of the matrix X and in decreasing order. The λ_1 gives the highest eigenvalue, which corresponds to the first principal component. Moreover, another parameter obtained is e in Equation 6, which presents the percentage of each principal component's variance to the total variance.

$$\boldsymbol{l} = \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{pmatrix} \tag{5}$$

$$\boldsymbol{e} = \begin{pmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{pmatrix} \tag{6}$$

The Improved Clustering Method

KPM is the primary basis of the improved method. The main framework of KPM was used with modifications in some procedures. The new algorithm is shown below:

Algorithm 1 : Improved Clustering Method

Step 1 : Initialize randomly K cluster centroids $\mu_1, \mu_2, \cdots, \mu_K$, wherein the positions of K centroid are chosen to be as independent events from the data set Φ.

(continued)

Algorithm 1 : Improved Clustering Method

Step 2: Assign a particular weight in every feature or dimension of sample x of MPC. The weight can be obtained by running the dataset in PCA and by performing as follows:

2.1 Generate the coefficient matrix $\boldsymbol{C}$, as shown in Equation 4.

2.2 Rank each loading in every column of $\boldsymbol{C}$ according to their contribution in that particular principal component. The loading with the most significant contribution is ranked *n*, followed by ranked *n-1*, until the lowest rank of *1*. This processing transforms the matrix in Equation 4 into a rank matrix *R* shown in Equation 7. Each entry now in the rank matrix is any number from 1 to n.

$$\boldsymbol{R} = \begin{bmatrix} r_{C11} & \cdots & r_{C1n} \\ \vdots & \ddots & \vdots \\ r_{Cn1} & \cdots & r_{Cnn} \end{bmatrix} \tag{7}$$

2.3 Multiply each column in the rank matrix with its corresponding percentage given in Equation 6 divided by 100, giving the new product matrix $\boldsymbol{P}$ as shown in Equation 8.

$$\boldsymbol{P} = \begin{bmatrix} r_{C11} & \cdots & r_{C1n} \\ \vdots & \ddots & \vdots \\ r_{Cn1} & \cdots & r_{Cnn} \end{bmatrix} \text{x} \begin{bmatrix} e_1/100 \\ e_2/100 \\ \vdots \\ e_n/100 \end{bmatrix} \tag{8}$$

2.4 Compute the sum of each row in $\boldsymbol{P}$ and divide by *n* to obtain the weight matrix $\boldsymbol{W}$, as shown in Equation 9, which contains the weight assigned to each variable or dimension of the dataset.

$$\boldsymbol{W} = \begin{bmatrix} \frac{\sum_1^n \left(r_{C11} * \frac{e_1}{100} + \ldots + r_{C1n} * \frac{e_n}{100} \right)}{n} \\ \vdots \\ \frac{\sum_1^n \left(r_{Cn1} * \frac{e_1}{100} + \ldots + r_{Cnn} * \frac{e_n}{100} \right)}{n} \end{bmatrix} \tag{9}$$

(continued)

Algorithm 1 : Improved Clustering Method

Step 3: Designate every weighted sample x of MPC to a particular cluster centroid μ_j : for every set x , as defined in Equation 10.

$$c^{(k)} := \underset{j}{\operatorname{argmin}} \left\{ \alpha_x \cdot d_{\mathrm{MPC}}(x, \mu_j^{(k)}) \right\} \tag{10}$$

where superscript (k) indicates the k-th iteration and c serves as the store indices in the k-th iteration of MPC clustering, α_x is the relative power of the sample x of MPC and for the d_{MPC} , instead of the Euclidean distance, the Minkowski distance in Equation 3 with the optimum p-value is used.

4. Update the cluster centroids: for each j, set as in Equation 11.

$$\mu_j^{(k+1)} := \frac{\sum_{x \in \Phi} 1\{c^{(k)} = j\} \alpha_x \cdot x}{\sum_{x \in \Phi} 1\{c^{(k)} = j\} \alpha_x} \tag{11}$$

5. Perform steps 3 and 4 again until the data have converged.

Accuracy Performance Evaluation

To evaluate the accuracy of performance of KPM and the improved method in clustering wireless propagation multipaths, the Jaccard index, objectively η_{jac}, was used. Jaccard index can have a value from 0 to 1, indicating the degree of accuracy. The higher the Jaccard score of index value, the better the clustering performance, with one being perfect. Jaccard index is just one of the many external comparison indices. It measures the similarity between two partitions as one type of external indices only considers the distribution of points in the various clusters. η_{jac} is given as follows (Varshavsky et al., 2005), as defined in Equation 12.

$$\eta_{\mathrm{jac}} = \frac{n_{11}}{n_{11} + n_{10} + n_{01}} \tag{12}$$

where

n_{11} is the number of pairs that are classified together correctly (Case 1).

n_{01} is the number of pairs that are not classified together correctly by the algorithm (Case 2).

n_{10} is the number of pairs that are incorrectly classified together when they are not supposed to (Case 3).

RESULTS AND DISCUSSION

The Optimal *p*-value for Minkowski Distance

Table 1 shows the tabulated *p*-value used in the Minkowski distance for KPM with the corresponding Jaccard accuracy score for each indoor channel scenario. Ten *p*-values were tested in two indoor channel scenarios, CS1 and CS2, generated from C2CM. Instead of the standard Euclidean distance in finding the value of d_{MPC} , the Minkowski distance was utilized. Different values of *p* were examined to determine the most optimum. Speed was also included to identify the effect of varying the *p*-values in the algorithm's computational duration.

It can be observed in Figure 1 that as the *p*-value increased, the accuracy decreased with both CS1 and CS2 showing the same trend. Moreover, the Minkowski distance with a *p*-value of 2 was just equivalent to the Euclidean distance. By varying the value of *p*, specifically going below the value of 2, a significant improvement in the accuracy performance was noticeable in both indoor channel scenarios. In Minkowski, a *p*-value of 2 was just equivalent to the Euclidean distance but going below a *p*-value of 2, enabling Minkowski to produce a higher distance between the two pints as compared to the Euclidean distance used by KPM. This affected the assignment of each multipath to a particular cluster centroid as the basis was the computed minimum distance. The higher distance obtained by Minkowski helped the algorithm to better identify the correct cluster centroid that each MPC should belong to. In Table 1, it can be seen that the *p*-value of 0.50 was the best option for the indoor channel scenario. Channel Scenario 2 obtained its highest accuracy at p=0.5. For CS1, p=0.5 did not give the highest accuracy but was closely related to p=0.25 and p=0.75. Considering the effect

of p=0.5 in CS2, it can be said that the best choice for p-value in indoor environments was 0.5. Figure 2 shows that the trend in the speed or computational duration of CS1 and CS2 was almost similar. Moreover, there was no consistent trend between speed and p-value, as it can also be seen that the increase in speed occurred not just in higher values of p but also in lower values. However, it can be noticed that at a p-value of 100, both channel scenarios obtained their highest computational duration.

Table 1

Finding the Optimal P-Value

	Channel Scenario 1 (CS1)		Channel Scenario 2 (CS2)	
p-value	Accuracy	Speed (s)	Accuracy	Speed (s)
0.25	0.9493	2.0061	0.9261	1.9093
0.50	0.9427	3.7635	0.9415	3.4145
0.75	0.9468	2.1637	0.9083	1.9933
1.00	0.9363	1.9360	0.8853	1.9327
2.00	0.8915	3.2200	0.8446	3.1400
10.00	0.8842	2.0984	0.8302	2.0211
50.00	0.8712	3.1232	0.8200	3.0916
100.00	0.8523	3.9091	0.7849	4.5087
500.00	0.7495	2.8425	0.7108	3.0635
1000.00	0.6940	1.7228	0.6016	2.7933

Figure 1

Jaccard Index versus P-Value

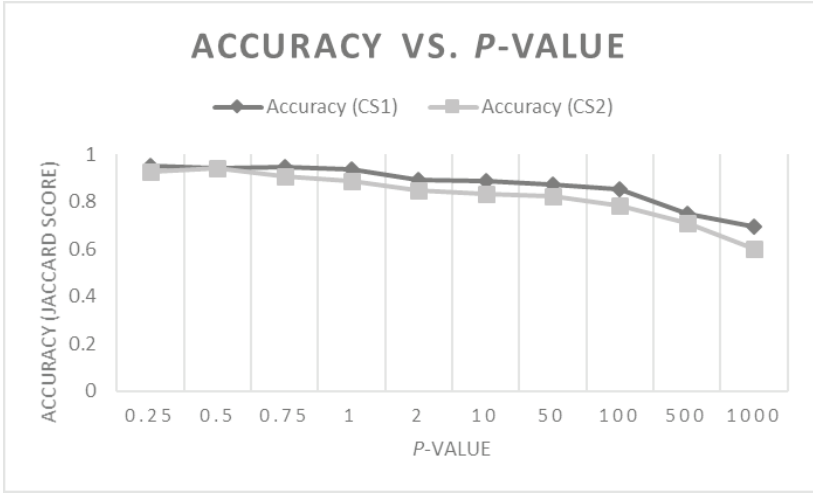
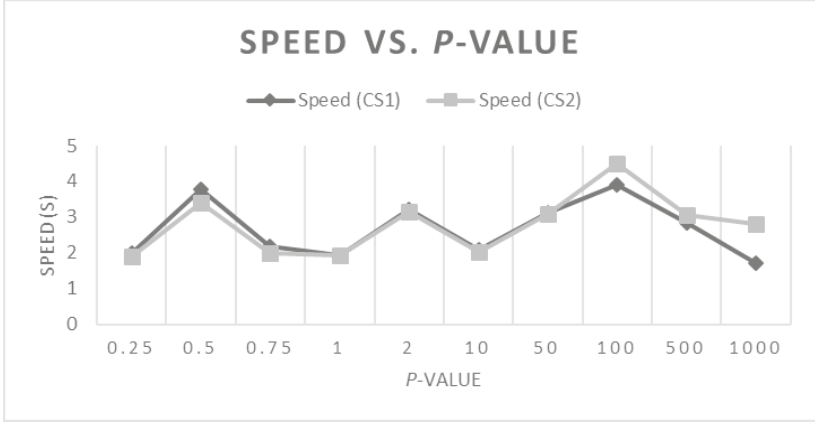


Figure 2

Plot of Speed versus P-Value



PCA-Based Dimension Weights

The first step in finding the weights for each dimension in the datasets was to run the dataset in the PCA in MATLAB and obtain the coefficient matrix *C*. Tables 2 and 3 present the obtained coefficient matrix for CS1 and CS2, respectively. Each table contains seven columns representing the seven principal components (PCs) generated. The PCs were arranged in decreasing order with PC1 as the highest or the most principal component, while PC7 is the least one. The tables also include seven rows for the seven dimensions (D) or features in the dataset. D1 is the whitened *x*-component of the angle of departure ($X_{x,}AoD$), D2 is the whitened *y*-component of the angle of departure ($X_{y,}AoD$), and D3 is the whitened *z*-component of the angle of departure ($X_{z,}AoD$).

On the other hand, D4 is the whitened *x*-component of the angle of arrival ($X_{x,}AoA$), D5 is the whitened *y*-component of the angle of arrival ($X_{y,}AoA$), and D6 is the whitened *z*-component of the angle of arrival ($X_{z,}AoA$). For D7, it is the whitened delay (τ) of the dataset. Each entry in the table represents the loading or weight of every dimension to the corresponding PC.

The second step was to find the latent values or the eigenvalues representing the variance of data for each PC shown in Table 4. Values were in decreasing order and gave an idea of each PC's contribution to the overall dataset. Getting the sum of all these eigenvalues resulted in a value of one. The first latent value was the highest and corresponded to the eigenvalue of PC1 and the same goes for PC2 to PC7. To check the contribution of each PC to the overall dataset, their percentage was obtained. The higher the percentage, the more significant the contribution of a particular PC to data distribution. Figures 3 and 4 illustrate the Scree plot or the eigenvalues' plot from the largest to the smallest. It can be observed that all seven dimensions had a significant contribution to the totality of the data distribution. That meant the seven PCs were considered for data analysis.

The next procedure was to find the specific weight to be used for each dimension in the dataset. To do this, the loading or weight for each dimension in every PC (see Tables 2 and 3) were ranked according to their significance in a particular PC. The higher the loading or weight,

the higher the rank is. The negative sign is not included as it only is indicated in the opposite direction of the PC axis. The highest rank to be given is 7 and the lowest is 1 since there are seven dimensions. The most significant loading gets a value of 7; the next biggest is assigned 6 until the smallest loading is assigned a value of 1. Every dimension has different rankings in each PC, depending on its contribution. Each dimension's rank is multiplied by that particular PC (see Table 4). Using Eq. (9), the sum of the seven columns was divided by 7. The results of this procedure can be found in Table 5. Two sets of weights were obtained, one from the data distribution of CS1 and the other from CS2. Weights generated from CS1 are labeled as SET 1, and weights from CS2 are grouped as SET 2.

Table 2

Principal Components of Channel Scenario 1

	PC1	PC2	PC3	PC4	PC5	PC6	PC7
D1	-0.1852	0.51267	0.3576	-0.0808	0.4194	0.6214	0.0804
D2	0.0216	0.5378	0.0929	0.6408	0.1605	-0.5150	-0.0060
D3	0.6889	-0.0569	-0.1645	0.0954	0.2275	0.1219	0.6476
D4	-0.1974	-0.2396	-0.4507	0.6655	0.0108	0.4930	-0.1202
D5	-0.1533	-0.5152	0.1646	0.0272	0.7889	-0.2342	-0.0774
D6	-0.1933	-0.3317	0.6737	0.3208	-0.3496	0.0526	0.4133
D7	0.6251	-0.1086	0.3904	0.1648	-0.0432	0.1820	-0.6187

Table 3

Principal Components of Channel Scenario 2

	PC1	PC2	PC3	PC4	PC5	PC6	PC7
D1	0.4870	-0.2135	-0.1628	0.0247	-0.1113	0.8142	-0.1228
D2	-0.3560	0.1255	-0.3624	-0.1007	0.7954	0.2881	0.0207

(continued)

	PC1	PC2	PC3	PC4	PC5	PC6	PC7
D3	0.6252	-0.0753	-0.0453	-0.1573	0.3278	-0.2571	0.6349
D4	-0.0218	0.2492	0.7588	-0.5222	0.1337	0.2663	0.0145
D5	0.0110	-0.3719	0.5132	0.6750	0.3752	0.0240	-0.0352
D6	-0.0593	0.6938	0.0266	0.4706	-0.1677	0.2531	0.4481
D7	0.4914	0.5013	-0.0185	0.1212	0.2467	-0.2288	-0.6157

Table 4

The Principal Component Variances for Each Channel Scenario

Channel Scenario 1			Channel Scenario 2		
PC	Latent Values	Percentage	PC	Latent Values	Percentage
PC1	1.4552	21.4547	PC1	1.6781	24.7253
PC2	1.1626	17.1401	PC2	1.2997	19.1499
PC3	1.1160	16.4537	PC3	1.0091	14.8688
PC4	0.9473	13.9659	PC4	0.8861	13.0551
PC5	0.8823	13.0075	PC5	0.8472	12.483
PC6	0.7943	11.7102	PC6	0.7084	10.4380
PC7	0.4251	6.2679	PC7	0.3583	5.2798

Figure 3

Scree Plot of PCs in CS1

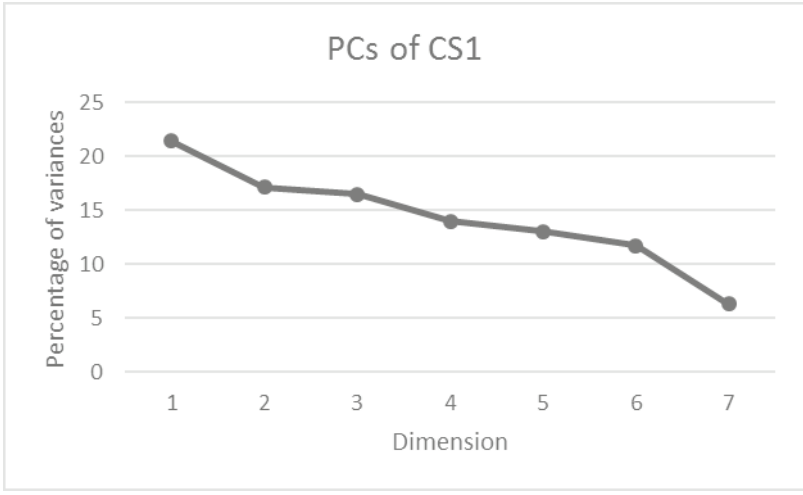


Figure 4

Scree Plot of PCs in CS2

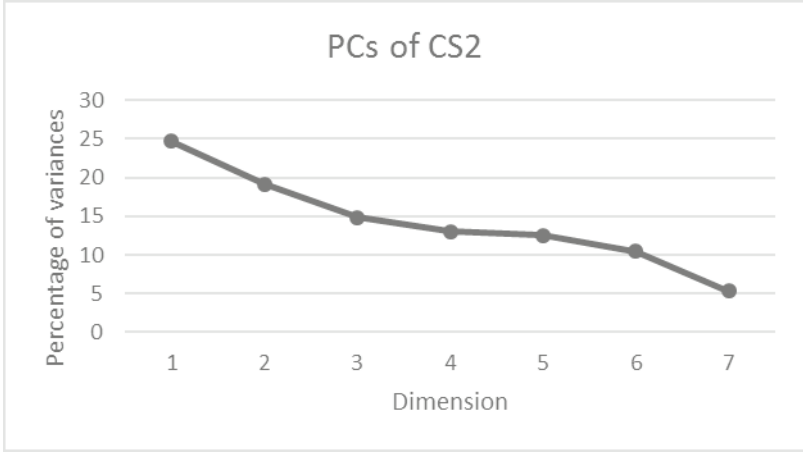


Table 5

Computed Weights for Each Dimension using PCA

	PCA-Based Weights						
	D1	D2	D3	D4	D5	D6	D7
SET 1	0.6038	0.5103	0.5164	0.6454	0.5136	0.6392	0.5713
SET 2	0.5147	0.5689	0.6145	0.5584	0.5746	0.5691	0.5997

Accuracy Performance of the Improved KPM

Table 6 gives the generated Jaccard score or index in each indoor CS. For CS1, the Jaccard index obtained using the original KPM in clustering wireless multipaths was 0.8915 or equivalent to 89.15 percent of accuracy. When the Minkowski distance at p=0.5 was applied in place of the Euclidean distance, the performance greatly improved to 0.9427 with a difference of 0.0512 or 5.12 percent. Using the calculated weights in SET 1 with Minkowski distance, the performance at 0.9358 was still higher than 0.8915. Still, there was a slight drop in performance as compared to using the Minkowski distance only. Nevertheless, when SET 2 weights were employed, there was an increase in performance as compared to only using the Minkowski distance. In the case of CS2, the same trend could be found. Utilizing the Minkowski distance greatly enhanced the performance of KPM as its Jaccard index jumped from 0.8446 to 0.9415. When applying the two sets of weights, it can be observed that SET 1 reduced performance while SET 2 offered some improvement. Considering the results, it can be said that utilizing the Minkowski distance at 0.5 p-value combined with the SET 2 weights in KPM produced an improvement in its clustering performance. The highest accuracy performance obtained in CS1 was 94.71 percent, while in CS2, it was 94.81 percent.

Table 6

Comparison of Accuracy Performance

	η_{jac} (Original KPM)	η_{jac} (KPM using Minkowski distance at p=0.5)	η_{jac} (KPM using Minkowski distance at p=0.5 and added SET 1 weights)	η_{jac} (KPM using Minkowski distance at p=0.5 and added SET 2 weights)
CS1	0.8915	0.9427	0.9358	0.9471
CS2	0.8446	0.9415	0.9391	0.9481

CONCLUSION AND RECOMMENDATIONS

Clustering wireless multipaths is an essential aspect of channel modeling. To attain a reliable wireless channel, it is imperative to have an accurate channel model. Moreover, this requires a clustering technique that can group the various wireless multipaths correctly. In this study, KPM's basic framework was improved by employing the Minkowski distance as the metric in determining the minimum distance of each multipath to the mean centroid of each cluster. Moreover, each dimension or feature in the dataset was given a particular weight based on the computed values in PCA. By combining the Minkowski distance and the PCA-based weights, it can be said that the performance of KPM greatly improved. Channel Scenario 1 showed a 5.56 percent increase in its accuracy performance, while CS2 offered a significant improvement of 10.35 percent. With this, it can be concluded that employing the PCA-based weights in KPM and using the Minkowski distance at an optimum p-value of 0.5 can enhance KPM's performance in clustering indoor datasets of C2CM.

The new clustering technique offered a significant increase in the accuracy of performance with the indoor channel scenario datasets as compared to the standard KPM, but further improvement is still needed to obtain a much higher Jaccard score. Other methods in determining the weight of each dimension of data can be explored.

Apart from that, other distance metrics such as the Hamming distance and Mahalanobis distance can also be employed to enhance the accuracy of performance further.

ACKNOWLEDGMENT

This study results from the research project funded by the De La Salle University (DLSU) and the Commission on Higher Education (CHED). Special acknowledgment also goes to the Computing and Archiving Research Environment (COARE) of the Department of Science and Technology - Advanced Science and Technology Institute (DOST-ASTI) for the computing resources.

REFERENCES

Blanza, J., Teologo, A., & Materum, L. (2019, August). Datasets for mutipath clustering at 285 MHz and 5.3 GHz bands based on COST 2100 MIMO channel model. In *2019 International Symposium on Multimedia and Communication Technology (ISMAC)* (pp. 1–5). IEEE. https://doi.org/10.1109/ISMAC.2019.8836143

Chen, Y., Zhang, Z., Song, X., Liu, J., Hou, M., Li, G., Xu, W., & Ma, A. (2019). Coherent clustering method based on weighted clustering of multi-indicator panel data. *IEEE Access*, *7*, 43462–43472. https://doi.org/dlsu.idm.oclc.org/10.1109/ACCESS.2019.2907270

Chong, C. C., Tan, C. M., Laurenson, D., McLaughlin, S., Beach, M., & Nix, A. (2005). A novel wideband dynamic directional indoor channel model based on a markov process. *IEEE Transactions on Wireless Communications*, *4*(4), 1539–1552. https://doi.org/10.1109/TWC.2005.850341

Chouikhi, H., Saad, M., & Alimi, A. (2017, January). Improved fuzzy possibilistic C-means (IFPCM) algorithms using Minkowski distance. In *International Conference on Control, Automation and Diagnosis* (pp. 402–405). IEEE. https://doi.org/dlsu.idm.oclc.org/10.1109/CADIAG.2017.8075692

COST 2100 channel model. (2018). https://github.com/cost2100/cost2100/tree/master/matlab

Gu, L. (2016, July). A novel sample weighting K-Means clustering algorithm based on angles information. In *International Joint Conference on Neural Networks* (pp. 3697-3702). IEEE.

Gustafson, C., Haneda, K., Wyne, S., & Tufvesson, F. (2014). On mm-wave multipath clustering and channel modeling. *IEEE Transactions on Antennas and Propagation*, *62*(3), 1445–1455.

Hanpinitsak, P., Saito, K., Takada, J.-I., Kim, M., & Materum, L. (2017). Multipath clustering and cluster tracking for geometry-based stochastic channel modeling. *IEEE Transactions on Antennas and Propagation*, *65*(11), 6015–6028.

He, R., Li,, Q., Ai, B., Geng, Y., Molisch, A., Kristem, V., Zhong, Z., & Yu, J. (2017). A kernel-power density-based algorithm for channel multipath components clustering. *IEEE Transactions on Wireless Communications*, *16*(11), 7138–7151. https://doi.org/10.1109/TWC.2017.2740206

He, R., Renaudin, O., Kolmonen, V. M., Haneda, K., Zhong, Z., Ai, B., & Oestges, C. (2015). A dynamic wideband directional channel model for vehicle-to-vehicle communications. *IEEE Transactions on Inducstrial Electronics*, *62*(12), 7870–7882.

Huang, D., Wang, C.-D., & Lai, J.-H. (2018). Locally weigthed ensemble clustering. *IEEE Transactions on Cybernetics*, *48*(5), 1460–1473. https://doi.org/dlsu.idm.oclc.org/10.1109/TCYB.2017.2702343

Kassambara, (2017). Principal component methods in R: Practical guide.http://www.sthda.com/english/articles/31-principal-component-methods-in-r-practical-guide/112-pca-principal-component-analysis-essentials/

Khaldi, B., Harrou, F., Cherif, F., & Sun, Y. (2020, February). Improving robots swarm aggregation performance through the Minkowski distance function. In *6th International Conference on Mechatronics and Robotics Engineering* (pp. 87–91). IEEE. https://doi.org/dlsu.idm.oclc.org/10.1109/ICMRE49073.2020.9064998

Khan, I., Luo, Z., Huang, J., & Shahzad, W. (2020). Variable weighting in fuzzy k-Means clustering to determine the number of clusters. *IEEE Transactions on Knowledge and Data Engineering*, *32*(9), 1838–1853. https://doi.org/dlsu.idm.oclc.org/10.1109/TKDE.2019.2911582

Laurila, J., Kalliola, K., Toeltsch, M., Hugl, K., Vainikainen, P., & Bonek, E. (2002). Wideband 3D characterization of mobile radio channels in urban environment. *IEEE Transactions on Antennas and Propagation*, *50*(2), 233–243. https://doi.org/10.1109/8.998000

Li, Y., Zhang, J., Ma, Z., & Zhang, Y. (2018). Clustering analysis in the wireless propagation channel with a variational gaussian mixture model. *IEEE Transactions on Big Data*, *6*(2), 223–232. https://doi.org/10.1109/TBDATA.2018.2840696

Liu, L., Czink, N., & Oestges, C. (2009). Implementing COST 273 MIMO channel model. In *Proc. NEWCOM-ACoRN Joint Workshop.*

Liu, L., Oestges, C., Poutanen, J., Haneda, K., Vainikainen, P., Quitin, F., Tufvesson, F., & De Doncker, P. (2012). The COST 2100 MIMO Channel Model. *IEEE Wireless Communications*, *19*(6), 92–99. https://doi.org/10.1109/MWC.2012.6393523

Minkowski Distance. (n.d.). https://bit.ly/3uPwGPW

Montaño, R., Alías, F., & Ferrer, J. (2013, September). Prosodic analysis of storytelling discourse modes and narrative situations oriented to text-to-speech synthesis. In *8th ISCA Speech Synthesis Workshop* (pp. 171–176).

Mota, S., Perez-Fontan, F., & Rocha, A. (2013). Estimation of the number of clusters in multipath radio channel data sets. *IEEE Transactions on Antennas and Propagation*, *61*(5), 2879–2883.

Poutanen, J., Haneda, K., Liu, L., Oestges, C., Tufvesson, F., & Vainikainen, P. (2011, April). Parametrization of the COST 2100 MIMO channel model in inddor scenarios. In *Proceedings of the 5th European Conference on Antennas and Propagation (EUCAP)* (pp. 3606–3610). IEEE.

Singh, A., & Jayaram, B. (2020, October). Performance of Minkowski-type distances in similarity search - A geometric approach. In *IEEE 5th International Conference on Computing Communication and Automation* (pp. 467–472). IEEE. https://doi.org/10.1109/ICCCA49541.2020.9250751

Toeltsch, M., Laurila, J., Kalliola, K., Molisch, A., Vainikainen, P., & Bonek, E. (2002). Statistical characterization of urban spatial radio channels. *IEEE Journal on Selected Areas in Communications*, *20*(3), 539–549. https://doi.org/10.1109/49.995513

Varshavsky, R., Linial, M., & Horn, D. (2005, November). COMPACT: A comparative package for clustering assessment. *International Symposium on Parallel and Distributed Processing and Applications* (pp. 159–167). Springer, Berlin, Heidelberg. https://doi.org/10.1007/11576259_18

Verdone, R., & Zanella, A. (2012). *Pervasive mobile and ambient* wireless *communications: COST action 2100.* Springer Science & Business Media.

Vuokko, L., Vainikainen, P., & Takada, J. (2005). Clusters extracted from measured propagation channels in macrocellular environments. *IEEE Transactions on Antennas and Propagation*, *53*(12), 4089–4098. https://doi.org/10.1109/TAP.2005.859763

Wang, W., Jost, T., Fiebig, U. C., & Koch, W. (2012, December). Time-variant channel modeling with application to mobile radio-based positioning. In *2012 Global Communications Conference (GLOBECOM)* (pp. 5038–5043). IEEE. https://doi.org/10.1109/GLOCOM.2012.6503919

Xu, H., Zeng, W., Zeng, X., & Yen, G. (2019). An evolutionary algorithm based on Minkowski distance for many-objective optimization. *IEEE Transactions on Cybernetics*, *49*(11), 3968–3979.https://doi.org/dlsu.idm.oclc.org/10.1109/TCYB.2018.2856208

Xu, R., & Wunsch, D. (2005). Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, *16*(3), 645–678.

Yu, K., Li, Q., & Ho, M. (2005). Measurement investigation of tap and cluster angular spreads at 5.2 GHz. *IEEE Transactions on Antennas and Propagation*, *53*, 2156–2160.

Zhu, M., Eriksson, G., & Tufvesson, F. (2013). The COST 2100 channel model: Parametrization and validation based on outdoor MIMO measurments at 300 MHz. *IEEE Transactions on Wireless Communications*, *12*(2), 888–897. https://doi.org/10.1109/TWC.2013.010413.120620