Physical Identification of Common Clusters for Outdoor Microcells

Julian David Villegas Gutierrez[1], Domingo Pimienta-del-Valle[2], Claude Oestges[1]
[1]ICTEAM, UCLouvain, Louvain-la-Neuve, Belgium, {julian.d.villegas, claude.oestges}@uclouvain.be
[2]UPM, Madrid, Spain, domingo.pimienta@upm.es

***Abstract*—In geometry-based stochastic models, Interacting Objects (IOs) define scattering obstacles. The identification of groups of waves (clusters) linked to IOs is addressed in this paper, based on measurements conducted in an outdoor microcellular scenario at 3.8 GHz. By contrast to the classical approach clustering waves in the parameter space, the proposed approach is able to identify physical IOs. Moreover, a method to define common clusters at different mobile locations is further described.**

***Index Terms*—clusters, interacting object, propagation.**

I. Introduction

Channel models as COST2100 rely on grouping sets of Mulitpath Components (MPCs) with similar properties. It has been shown experimentally [3], that MPCs tend to appear in clusters.

In order to identify clusters physically, a common approach is to follow the procedure given by Figure 1.

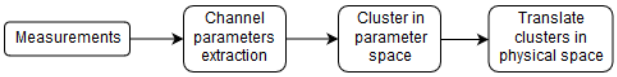

Fig. 1: Common approach in literature

In this case, there is a need to translate clusters in the parameter space into clusters in the physical space, in order to find out the real-world scatterers. Such a translation is performed in [12] and [2], through the use of a measurement-based ray tracer. However, this translation is not trivial and it can lead to a great mismatch between parameter-based clusters and their physical counterpart, as the angular spreads of parameter-based clusters can allow to have different members of the same cluster, located in different regions in space.

Different algorithms have been used to cluster channel parameters. The power-weighted version of K-Means, namely KPowerMeans [4] has been widely used. In [7] a Gaussian Mixture Model (GMM) is used with results for outdoor-to-indoor data. Other approximations, use density-based clustering algorithms [6]. Any of those clustering algorithms requires a distance measure as an input. Because the parameter space is composed by data which ranges in different scales, units, and domains, there is no straightforward approach to develop a particular one. In [4], the Multipath Component Distance (MCD) is introduced as the distance metric used along with K-Power-Means algorithm. However, MCD requires a tuning parameter (ζ) and does not give a direct relationship with the real distance between scatterers in the environment. Therefore, under the situation in which the K-Power-Means algorithm does not perform well, it is not evident how to interpret the closeness given by the MCD between datapoints in the parameter space.

In this paper we follow the approach proposed in [1], in which the clustering algorithm is only performed after channel parameters are related with the physical scenario. However, our physical model is restricted to first order intercepts in both departure and arrival sides. Besides, the distance measure used in [1] implies the existence of a 4D-euclidean space (composed by departure and arrival (x, y) points), whereas the physical space under consideration is two-dimensional and both arrival and departure sides are linked.

Section II shows the measurement set-up carried out during the measurement campaign. After that, section III shows the channel model from which channel parameters are extracted. Section IV is divided into 5 parts: part IV-A, where the physical model is explained; part IV-B deals with the general clustering formalism; part IV-C introduces the distance measure used here; part IV-D presents key points of K-Medoids; and finally, in part IV-E some definitions about common clusters and how to asses their validity are introduced.

II. Measurement Setup

The measurement campaign was carried out in an outdoor environment in Louvain-la-Neuve using the UCLouvain-ULB Elektrobit Channel Sounder.

Fig. 2: Top view of measurement scenario

TABLE I: Measurement setup

Bandwidth	200 MHz
Center frequency	3.8 GHz
Code Length	5.11 μs
Tx Antenna	UCA - 8 elements
Rx Antenna	2 x 4 Planar Array

A Uniform Circular Array (UCA) of 8 dipoles was used as the Tx. The Rx was a 2-by-4 dual polarized planar array of slanted patch antennas. The main setup paramaters for the device and antennas are shown in Table I. During the measurement (see Figure 2), the Rx (blue circle) was static, while the Tx was moving along a closed loop trajectory (red shape).

III. Parameter Extraction

The channel parameters (Direction of Arrival -DoA-, Direction of Departure -DoD-, and τ) are extracted from the channel measurements. During this step, SAGE algorithm was applied to the channel measurements. The SAGE algorithm fits the most suitable channel parameters to the channel model listed in (1) and (2).

$$\vec{r}(t) = \sum_{l}^{L} \vec{s}(t;\theta_l) + \vec{v}(t) \tag{1}$$

$$\vec{s}(t;\theta_l) = e^{j2\pi\nu_l t} \sum_{p_1=1}^{2}\sum_{p_2=1}^{2} \alpha_{l,p_1,p_2} \vec{c_{2,p_2}}(\Omega_{Rx,l}), \quad \vec{c_{1,p_1}}(\Omega_{Tx,l})^T \vec{u}(t-\tau_l) \tag{2}$$

where

- α_{l,p_1,p_2} : complex gain due to path l, polarization p_1 at the transmitter, and polarization p_2 at the receiver.
- ν_l : doppler shift due to path l.
- $\vec{u}(t) \in \mathbb{R}^{M\times 1}$: vector which contains the complex baseband representation of the signal at the input of the mth element of the transmitter antenna array.
- $\vec{c_{1,p_1}}(\Omega_{Tx,l}) \in \mathbb{R}^{M\times 1}$: response of the transmitter antenna array to a wave impinging from direction $\Omega_{Tx,l}$, due to polarization p_1.
- $\vec{c_{2,p_2}}(\Omega_{Rx,l}) \in \mathbb{R}^{N\times 1}$: response of the receiver antenna array to a wave impinging from direction $\Omega_{Rx,l}$, due to polarization p_2.
- $\Omega_{1,l}, \Omega_{2,l}$: Direction of Departure/Arrival of path l; unitary vectors containing the azimuth and elevation angles, or its correspondence in cartesian coordinates.
- $\vec{v} = \sqrt{\frac{N_0}{2}}\vec{W(t)}$: $\vec{W}(t) \in \mathbb{R}^{N\times 1}$ is a vector of independent complex white noise processes, such that $\mathbb{E}[W_k(t_1)^* W_k(t_2)] = 2\delta(t_2 - t_1)$

At the end of the SAGE algorithm, a vector $\theta_l^t = [\tau_l^t, \nu_l^t, \Omega_{Tx,l}^t, \Omega_{Rx,l}^t]$ of *channel parameters* is obtained, for each time-snapshot t and multipath l. The maximum number of paths per snapshot was set to 20.

IV. Physical Identification Procedure

A. Physical model

Considering only the azimuth angles denoted as $\phi_{Tx,l}^t$ and $\phi_{Rx,l}^t$, respectively, our approach is illustrated in Figure 3.

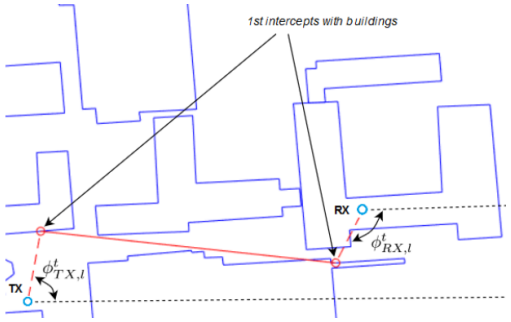

Fig. 3: Physical model used.

Both Tx or Rx, identified by blue circles, transmit or receive the lth wave of time-snapshot t, with a fixed $\phi_{Tx,l}^t$ and $\phi_{Rx,l}^t$, respectively. We consider that the first possible IOs (shown as red circles) where the wave impinges, are the contours of the buildings. Then, our model assumes that the wave follows the shortest path between these 2 first-order IOs, given by the solid red line. Eventually, we obtain physical coordinates $\vec{Z}_{arr,i}^t = [x_{arr,i}, y_{arr,i}]$ and $\vec{Z}_{dep,i}^t = [x_{dep,i}, y_{dep,i}]$, that represent the first-order IO i in time-snapshot t. The model is thus giving an idea of the shortest path between the transmitter and receiver, assuming there is no LoS and that the buildings are the first-possible IO where the wave impinges. The wave propagates through the different propagation mechanisms (either reflection, diffraction or diffuse scattering).

B. Clustering formalism

Let $\Gamma_k^t : \mathcal{Z}^t \rightarrow \mathcal{M}_k$ be a mapping that assigns to each member of the set of IO coordinates $\mathcal{Z}^t \subset \mathbb{R}^{M\times 4} = \{z_1, z_2, ..., z_M\}$, a number in the set $\mathcal{M}_k = \{1, 2, 3, ..., k\}$. Each $z_i \in \mathcal{Z}^t$ is a vector of the form $z_i = [\vec{Z}_{dep,i}^t, \vec{Z}_{arr,i}^t], i = 1, 2, ..., M$. Here, M is the number of IOs in time-snapshot t. We call Γ_k^t a *clusterization*. It is noteworthy to see that for each k there exists a different *clusterization*, and so, there is a need to define a criterion to compare them.

Let $\mathcal{Q}_k^t = \{\mathcal{C}_i^t\}_{i=1}^k$ be the set of clusters for time-snapshot t, such that $|\mathcal{Q}_k^t| = k$, $\mathcal{C}_i^t = \{z_j \in \mathcal{Z}^t \mid i \in \mathcal{M}_k \wedge 1 \leq j \leq M \wedge |\mathcal{C}_i^t| \neq k\}$ and $\sum_i |\mathcal{C}_i^t| = k$. We drop the time index for notation convenience. Near each z_j can be defined a disk $Dz_j(\epsilon)$ of length ϵ, such that: $Dz_j(\epsilon) = \{y \in \mathbb{R}^{1\times 4} \mid \mathbf{d}(z_j, y) \leq \epsilon\}$. We will call the function $\mathbf{d}(\cdot,\cdot)$ a *distance measure*.

We define an *intra-cluster* measure d_β^k for clusterization Γ_k^t

$$d_\beta^k = \max_{\forall \mathcal{C}_i \in \mathcal{Q}_k^t} \max_{\forall x,y \in \mathcal{C}_i} \mathbf{d}(x,y) = \max_{\forall \mathcal{C}_i \in \mathcal{Q}_k^t} d_{\beta,i}^k \tag{3}$$

where $d^k_{\beta,i}$ represents an *intra-cluster* measure of cluster $\mathcal{C}_i \in \mathcal{Q}^t_k$. Before we define an *inter-cluster* measure, the intersection set $\mathcal{I}_i(\epsilon)$ of all disks $Dz_j(\epsilon)$, for a given cluster $\mathcal{C}_i$, is introduced: $\mathcal{I}_i(\epsilon) = \cap_{j=1}^{|\mathcal{C}_i|} Dz_j(\epsilon)$. Then, the *inter-cluster* measure d^k_α for clusterization Γ^t_k is

$$d^k_\alpha = \min_{\forall u \in \mathcal{I}_i(d^k_{\beta,i}),\ \forall v \in \mathcal{I}_j(d^k_{\beta,j}),\ \forall i \neq j} \mathbf{d}(u,v) \tag{4}$$

A practical implementation of (4) is not possible, as it would require to compute the disks $Dz_j(d^k_{\beta,j})$ which are not defined for the distance given by (7). Instead, we replace (4) by (5):

$$d^k_\alpha = \min_{\forall u \in \mathcal{C}_i,\ \forall v \in \mathcal{C}_j,\ \forall i \neq j,\ \mathcal{C}_i, \mathcal{C}_j \in \mathcal{Q}^t_k} \mathbf{d}(u,v) \tag{5}$$

The purpose of any criterion would be to maximize d^k_α and minimize d^k_β at the same time. We use a slightly modified version of Dunn's index to attribute a score $s(\Gamma_k)$ to each clusterization:

$$s(\Gamma_k) = 1 - \frac{d^k_\beta}{d^k_\alpha} \tag{6}$$

The best solution k_B according to the criterion in (6), will be that for which: $k_B = \text{argmax}_k\ s(\Gamma_k)$. Bad clusterizations can be identified immediately for $s(\Gamma_k) < 0$, as in that case $d^k_\beta > d^k_\alpha$.

C. Distance measure

In (3), (4) and (6) the function $\mathbf{d}(\cdot,\cdot)$ needs to be defined. As shown in Figure 3, a point $z_i \in \mathcal{Z}^t$ is composed by two points coupled in euclidean space. To measure the distance between 2 points $z_i,\ z_j \in \mathcal{Z}^t$, the 2 line segments $\mathcal{L}_i = \{\vec{w}_p = \vec{Z}_{dep,i} + p(\vec{Z}_{arr,i} - \vec{Z}_{dep,i}) \mid \forall p \in [0,1]\}$ and $\mathcal{L}_j = \{\vec{v}_p = \vec{Z}_{dep,j} + p(\vec{Z}_{arr,j} - \vec{Z}_{dep,j}) \mid \forall p \in [0,1]\}$, are defined. Then, the distance measure used for the clustering algorithm is:

$$\mathbf{d}(z_i, z_j) = \sum_p \parallel \vec{w}_p - \vec{v}_p \parallel_2,\ \vec{w}_p \in \mathcal{L}_i,\ \vec{v}_p \in \mathcal{L}_j \tag{7}$$

The sum in (7) is performed over different values of p that lie on different positions of lines $\mathcal{L}_i$ and $\mathcal{L}_j$. The notation $\|\cdot\|_2$ means euclidean norm in $\mathbb{R}^2$. The different values of p must be at least 2 (0 and 1), and can be as many as wanted. We choose values of p by dividing the interval $[0,1]$ in N points, and then getting the $p = i * \frac{1}{N-1},\ i = 0, 1, ..., N-1$. No big improvements had been noted for more than 20 values of p.

In Figure 4, because $\vec{Z}_{dep,i} = \vec{Z}_{dep,u} \wedge \vec{Z}_{dep,j} = \vec{Z}_{dep,v}$, as well as, $\vec{Z}_{arr,i} = \vec{Z}_{arr,v} \wedge \vec{Z}_{arr,j} = \vec{Z}_{arr,u}$, it is desired that $\mathbf{d}(z_i, z_j) = \mathbf{d}(z_u, z_v)$. However, when $p > 0$, the euclidean distances b_i in the right-side figure are shorter than distances a from the left-side figure, making $\mathbf{d}(z_i, z_j) > \mathbf{d}(z_u, z_v)$.

Hence, a correction is introduced by replacing points $z_u,\ z_v$ in the right-side graph in Figure 4 by the equivalent points z^{eq}_u and z^{eq}_v:

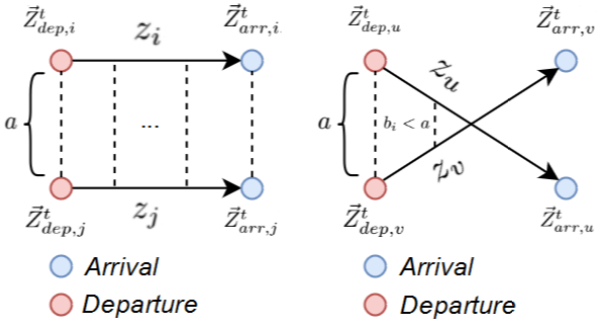

Fig. 4: Two situations for which (7) must be equal

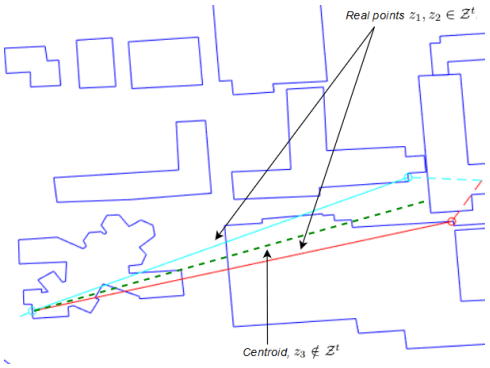

Fig. 5: K-Means centroid problem

$$z^{eq}_u = [\vec{Z}^t_{dep,u}, \vec{Z}^t_{arr,v}] \wedge z^{eq}_v = [\vec{Z}^t_{dep,v}, \vec{Z}^t_{arr,u}] \tag{8}$$

Equation (8) applies to any pair of points z_i, z_j for which there is a crossing in the associated line segments $\mathcal{L}_i$ and $\mathcal{L}_j$.

D. Algorithm

The K-Medoids algorithm is used as a variant of the K-Means algorithm. The difference is that K-Medoids replaces the concept of centroid by the concept of *medoid*. Indeed, the average between a number of points in the set $\mathcal{Z}^t$ can be problematic with the K-Means centroid, as shown in Figure 5: two existing points (identified by the solid lines) $z_1, z_2 \in \mathcal{Z}^t$, and one 'fictious' point $z_3 \notin \mathcal{Z}^t$ (green slotted line) are plotted. Assuming that z_3 is the centroid at a particular iteration of K-Means, and that $\mathbf{d}(z_1, z_3) \approx \mathbf{d}(z_2, z_3)$, K-Means will tend to group z_1, z_2 in the same cluster, even if they are not really close. The K-Medoids method avoids that problem by defining the *medoid* as the point $z_i \in \mathcal{Z}^t$ for which: $\nexists u \in \mathcal{Z}^t \mid \mathbf{d}(u, z_j) < \mathbf{d}(z_i, z_j), \forall z_j \in \mathcal{Z}^t$.

E. Common clusters

Given two time-snapshots t_1, t_2, we call clusters $\mathcal{C}^{t_1}_i \in \mathcal{Q}^{t_1}_{k_1}$ and $\mathcal{C}^{t_2}_j \in \mathcal{Q}^{t_2}_{k_2}$ *common* if:

$$\mathbf{d}(z_p, z_q) < \min\{d^{k_1}_{\beta,i}, d^{k_2}_{\beta,j}\},\ \forall z_p \in \mathcal{C}^{t_1}_i, \forall z_q \in \mathcal{C}^{t_2}_j \tag{9}$$

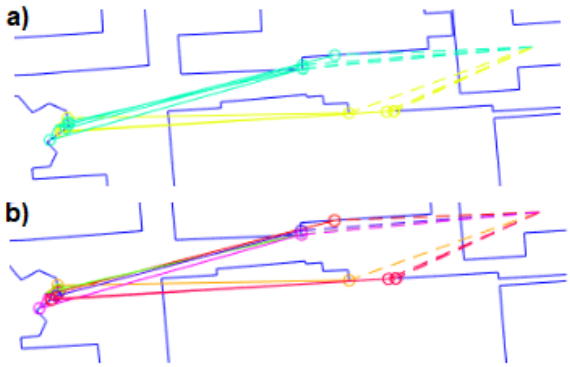

Fig. 6: a) Cluster detection in K-Means. b) Lack of detection of clusters in K-Medoids

Equation (9) is guaranteeing that members of each cluster belong to the other cluster, as the minimum of both *intra-cluster* measures has been taken into account. A more restrictive condition can be imposed on members of both clusters, by checking if all the members of one cluster (say $\mathcal{C}_i^{t_1}$) belong to the intersection set of the other cluster (say $\mathcal{I}_j(d_{\beta,j}^{k_2})$), but it is required to find out the intersection set, and as stated previously, up to now there is no proper definition for $\mathcal{I}_j(d_{\beta,j}^{k_2})$ for the distance measure in (7).

In order to validate the results given by the common clusters, we separately check second-order statistics for two clusters $\mathcal{C}_1$, $\mathcal{C}_2$ that are to said common, and compare them. So, for each pair of clusters, we compute (10) and check the relative error $e_r(\phi_\sigma(\mathcal{C}_1), \phi_\sigma(\mathcal{C}_2))$ in (11).

$$\phi_\sigma(\mathcal{C}_j) = \sqrt{\sum_k |\exp(i\phi_k) - \exp(i\overline{\phi_k})|^2},\ \overline{\phi} = \sum_k \frac{\exp(i\phi_k)}{|\mathcal{C}_j|} \tag{10}$$

$$e_r(a,b) = \frac{|a-b|}{\max\{a,b\}} \tag{11}$$

where (10) satisfy $\forall i \mid z_i \in \mathcal{C}_j,\ \mathcal{C}_j \in \mathcal{Q}_{k_B}^t$

V. Results

A. Clusters per snapshot

Figures 6 and 7 present a visual comparison between the K-Means and K-Medoids clusterization, using the *inner* distance measure defined in (7). It can be seen that there is a trade-off between the accuracy of detected clusters and the error introduced by not grouping points that actually can be regarded as part of one cluster.

On the one hand, from Figure 6-a, we see that K-Means algorithm successfully detect the two clusters identified by yellow and green color solid lines. In Figure 6b, K-Medoids tend to assign a cluster to each IO, and so, almost each solid line has a different color, meaning that it is part of a different cluster. The IOs are the circles at the endpoints of the solid lines in all graphs.

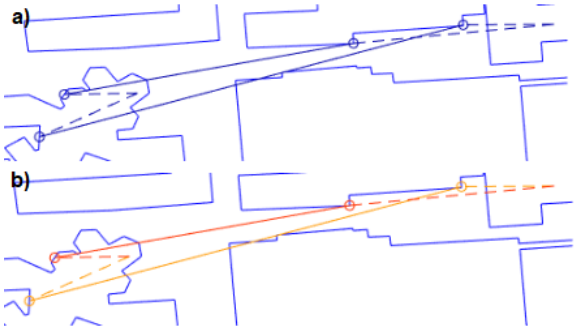

Fig. 7: a) Lack of accuracy in K-Means. b) Accuracy in K-Medoids

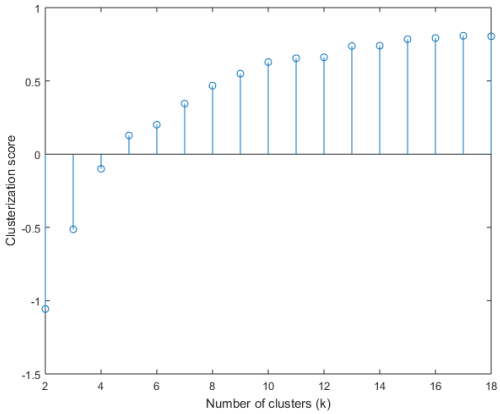

Fig. 8: Clusterization score vs number of clusters

On the other hand, from Figure 7-a, we see that K-Means groups two IOs (endpoints of blue solid lines) that do not belong to the same cluster. In Figure 7-b, the K-Medoids algorithm perceives that those two IOs are, in fact, part of different clusters (represented there by different colors). This situation is happening because (excluded $k = M$), as the possible number of clusters k is growing, the clusterization score tends to 1, i.e. it makes each point $z_i \in \mathcal{Z}^t$ a cluster.

When k grows, both d_β^k and d_α^k tend to diminish: the intra-cluster measure becomes 0, whereas the inter-cluster measure tends to diminish as well, because it is becoming the minimum possible distance between any two points in $\mathcal{Z}^t$. However, d_β^k diminishes faster than d_α^k, and so, the ratio $d_\beta^k/d_\alpha^k \to 0$, causing the clusterization score to tend to 1.

In Figure 8 it can be seen that it tends to be maximum for the maximum k, but as well, it suggest that there are other peaks which are not too far from the maximum peak. We decide to establish a threshold for what does it mean 'too far' from the maximum. If for $s(\Gamma_{k_1}^t)$ and $s(\Gamma_{k_2}^t)$, with $k_1 < k_2$, the relative error in (11): $e_r(s(\Gamma_{k_1}^t), s(\Gamma_{k_2}^t)) < 0.1$, then we take $s(\Gamma_{k_1}^t)$ as the best clusterization score, and $k_B = k_1$ as the best number of clusters for t.

B. Common clusters

Common clusters identify by (9) are expected to exhibit similar statistics. That means that if we compare the angular spreads $\phi_\sigma(\mathcal{C}_i)$ and $\phi_\sigma(\mathcal{C}_j)$ of 2 common clusters $\mathcal{C}_i$, $\mathcal{C}_j$, the relative error in (11) must remain low.

In order to validate the common cluster identification, we find the common clusters between two independent and random sets of 2500 time-snapshots each one, with at most 19 clusters per snapshot. After that, the relative error is computed for all pairs of common clusters. The process is repeated 50 times.

A summary of the descriptive statistics for the relative error of angular spreads is presented in Table II. As can be seen, the mean relative error is under 20%, which is reasonable given the size of the comparison. It is expected to have a mismatch between two cluster angular spreads, as the clusterization score is not giving always the best number of clusters. Modified clusterization scores must be investigated in order to improve the performance of the clustering algorithm.

TABLE II: Statistics for $e_r(\phi_\sigma(\mathcal{C}_i), \phi_\sigma(\mathcal{C}_j))$

Mean	Std. Dev	Median
18.42%	20.86%	11.73%

VI. Conclusion

Physical IOs have been identified in an outdoor micro-cell scenario. The successful identification of such scatterers relies on the performance of the clustering algorithm and the choice of the distance measure. Moreover, the approach followed here, by which clusters are found after translating channel parameters into the physical domain, seems to be an alternative to be researched, as it enables to map clusters directly into the physical domain.

Careful attention must be paid to the built-up of the physical model. First-order clusters of IOs were found; which means that waves follow the shortest path between first intercepts in the departure and arrival sides. By allowing more bounces, the number of IOs will increase, but the found IOs will remain scatterers.

As common clusters must have similar statistical properties, equations (7), (6) and (9) might be changed in order to account for statistical information. Moreover, delay information haven't been taken into account in the physical model. For that reason, an improvement can be made by finding the IOs for which the delay information is bigger than the delay computed from our physical model.

VII. Acknowledgment

This work was funded by the SWIPT project under UCLouvain ARC program. It was also carried out in the framework of COST Action CA15104 IRACON.

References

[1] P. Hanpinitsak, K. Saito, J. Takada, M. Kim and L. Materum, *Multipath Clustering and Cluster Tracking for Geometry-Based Stochastic Channel Modeling*, IEEE Transactions on Antennas and Propagation, vol. 65, no. 11, pp. 6015-6028, Nov. 2017.

[2] M. Zhu, *Geometry-based Radio Channel Characterization and Modeling: Parameterization, Implementation and Validation*, PhD dissertation, Lund University, 2014.

[3] L. Liu, C. Oestges, J. Poutanen, K. Haneda, P. Vainikainen, F. Quitin, F. Tufvesson, P. De Doncker, *The COST 2100 MIMO channel model*, IEEE Wireless Communications, vol. 19, no. 6, pp. 92-99, December 2012.

[4] N. Czink, P. Cera, J. Salo, E. Bonek, J. Nuutinen and J. Ylitalo, *A Framework for Automatic Clustering of Parametric MIMO Channel Data Including Path Powers*, IEEE Vehicular Technology Conference, pp. 1-5, Montreal, 2006.

[5] D. Ganguly, S. Mukherjee, S. Naskar and P. Mukherjee, *A Novel Approach for Determination of Optimal Number of Cluster*, International Conference on Computer and Automation Engineering, pp. 113-117, Bangkok, 2009.

[6] R. He, Q. Li, B. Ai, Y. L. Geng, A. F. Molisch, V. Kristem, Z. Zhong and J. Yu, *A Kernel-Power-Density-Based Algorithm for Channel Multipath Components Clustering*, IEEE Transactions on Wireless Communications, vol. 16, no. 11, pp. 7138-7151, Nov. 2017.

[7] 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.

[8] D. Shutin, *Clustering wireless channel impulse responses in angular-delay domain*, IEEE 5th Workshop on Signal Processing Advances in Wireless Communications, pp. 253-257, Lisbon, 2004.

[9] R. He, B. Ai, A. F. Molisch, G. L Stüber, Q. Li, Z. Zhong and J. Yu, *Clustering Enabled Wireless Channel Modeling Using Big Data Algorithms*, IEEE Communications Magazine, vol. 56, no. 5, pp. 177-183, May 2018.

[10] F. Luan, A. F. Molisch, L. Xiao, F. Tufvesson and S. Zhou, *Geometrical Cluster-Based Scatterer Detection Method with the Movement of Mobile Terminal*, IEEE 81st Vehicular Technology Conference (VTC Spring), pp. 1-6, Glasgow, 2015.

[11] F. Luan, A. F. Molisch, L. Xiao, F. Tufvesson and S. Zhou, *Geometrical Cluster-Based Scatterer Detection Method with the Movement of Mobile Terminal*, IEEE 81st Vehicular Technology Conference (VTC Spring), pp. 1-6, Glasgow, 2015.

[12] J. Poutanen, *Geometry-based radio channel modeling: Propagation analysis and concept development*,PhD dissertation, Aalto University, 2011.

[13] N. Czink, X. Yin, H. OZcelik, M. Herdin, E. Bonek and B. H. Fleury, *Cluster Characteristics in a MIMO Indoor Propagation Environment*, IEEE Transactions on Wireless Communications, vol. 6, no. 4, pp. 1465-1475, April 2007.