



OnRMap

An Online Radio Mapping Approach for Large Intelligent Surfaces

Santos, Herman L.; Croisfelt, Victor; Vaca-Rubio, Cristian J.; Abrao, Taufik; Popovski, Petar

Published in:

2023 IEEE International Conference on Communications Workshops

DOI (link to publication from Publisher):

[10.1109/ICCWorkshops57953.2023.10283491](https://doi.org/10.1109/ICCWorkshops57953.2023.10283491)

Publication date:

2023

Document Version

Accepted author manuscript, peer reviewed version

[Link to publication from Aalborg University](#)

Citation for published version (APA):

Santos, H. L., Croisfelt, V., Vaca-Rubio, C. J., Abrao, T., & Popovski, P. (2023). OnRMap: An Online Radio Mapping Approach for Large Intelligent Surfaces. In *2023 IEEE International Conference on Communications Workshops: Sustainable Communications for Renaissance, ICC Workshops 2023* (pp. 1337-1342). Article 10283491 IEEE (Institute of Electrical and Electronics Engineers).
<https://doi.org/10.1109/ICCWorkshops57953.2023.10283491>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal -

Take down policy

If you believe that this document breaches copyright please contact us at vbn@aub.aau.dk providing details, and we will remove access to the work immediately and investigate your claim.

OnRMap: An Online Radio Mapping Approach for Large Intelligent Surfaces

Herman L. dos Santos*, Victor Croisfelt†, Cristian J. Vaca-Rubio†, Taufik Abrão*, and Petar Popovski†

*Department of Electrical Engineering, Universidade Estadual de Londrina, Londrina, Brazil

†Department of Electronic Systems, Aalborg University, Aalborg, Denmark

E-mail: hermanlds@gmail.com, {vcr, cjvr, petarp}@es.aau.dk, and taufik@uel.br

Abstract—We introduce **OnRMap**, an online radio mapping (RMap) approach for the sensing and localization of active users (AUs), devices that are transmitting radio signals, and passive elements (PEs), elements that are in the environment and are illuminated by the AUs’ radio signals. OnRMap processes the signals received by a large intelligent surface and produces a radio map (RM) of the environment based on signal processing techniques. The method then senses and locates the different elements without the need for offline scanning phases, which is important for environments with frequently changing spatial layouts. Empirical results demonstrate that OnRMap presents a higher localization accuracy than an offline method, but the price paid for being an online method is a moderate reduction in the detection rate.

Index Terms—Large intelligent surface (LIS), sensing, localization, radio mapping (RM).

I. INTRODUCTION

Large intelligent surface (LIS) is an important concept on the evolution path of wireless multi-antenna systems. It originally refers to a continuous electromagnetic surface able to transmit and receive radio waves [1]. In practice, a LIS is envisioned as a collection of closely-spaced antenna elements deployed across a large 2D surface, which can be easily integrated into the propagation environment, *e.g.*, placed on walls or ceilings. In addition to its well-investigated communication capabilities [1], [2], LIS also holds potential for radio sensing and localization [3], with applications in self-driving vehicles, unmanned aerial vehicles, or autonomous robots. These use cases normally require the construction of radio maps (RMs), whose process can exploit the radio frequency (RF) signals emitted by the wireless devices in the environment of interest, termed active users (AUs). Moreover, static and/or dynamics objects that are not transmitting RF signals, termed passive elements (PEs), can also be detected/sensed and located by exploiting the multipath components of the RF signals transmitted by the AUs. We refer to radio mapping (RMap) as the

process of obtaining RMs, which is the main subject of this paper.

In [4], [5], the authors used a LIS to obtain RM, treating the received signals at the LIS as a digital image and creating an RMap method based on techniques from digital image processing and computer vision. Despite the good detection performance of AUs and PEs, digital image processing requires offline processing, which is not suitable for environments with frequently changing spatial layouts. Other previous RMap methods rely on the discretization of outdoor [6] and indoor [7] static environments, evaluating the path between AUs and PEs in a pixel-like manner. The critical problem with pixel-based approaches is that good detection performance requires increased pixel granularity, resulting in exponential complexity. Differently from this, the authors of [8], [9] used the related concept of radio tomographic imaging (RTI) that allowed them to obtain an RM of moving PEs (humans) by imaging their attenuation in a wireless sensor network (WSN) comprised of fixed-located sensors in a square area. However, this demands dedicated sensors, making it more expensive than methods that exploit widespread wireless AUs.

This work proposes *OnRMap*, an online RMap approach based on classical signal processing techniques. In contrast to [4], [5], OnRMap eliminates the need for offline scanning phases, being more robust to dynamic environments at the cost of a reasonably lower detection rate in comparison to [4]. The numerical results indicate that OnRMap provides a higher localization accuracy of the PEs.

II. SYSTEM MODEL

Consider the indoor communication system within an enclosed room and where a LIS is placed on the ceiling, as illustrated in Fig. 1. The LIS covers the whole room beneath it and is composed of a uniform planar array (UPA) containing $N = N_x \cdot N_y$ antenna elements equally spaced by $\lambda/2$. Within the room, there are R PEs and K AUs.

A. Channel Model

Assume that when the K AUs transmit uplink (UL) signals, those impinge at the LIS either as a result of line-of-sight (LoS) and/or as non-line-of-sight (NLoS) propagation. The latter occurs due to reflections of the transmitted signals on the PEs present in the room. For mathematical tractability, we ignore the NLoS components resulting from reflections on

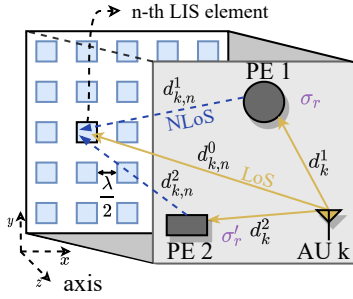


Figure 1. Indoor communication scenario with a LIS installed on the ceiling. The LoS (AU-to-LIS) and NLoS (AU-to-PEs-to-LIS) rays from a single AU are depicted. PEs can be different objects characterized by different parameters σ_r .

the wall and floor in the formulation; besides, we assume one reflection per PE. Let $\mathbf{h}_k \in \mathbb{C}^{N \times 1}$ denote the channel vector of the k -th AU to the LIS. Considering an indoor scenario, we use the spherical wavefront assumption and adopt the Saleh-Valenzuela model [10], where the n -th entry of \mathbf{h}_k is

$$h_{k,n} = \underbrace{\beta_{k,n}^0 e^{-j\frac{2\pi}{\lambda} d_{k,n}^0}}_{\text{LoS}} + \sum_{r=1}^R \underbrace{\beta_{k,n}^r e^{-j\frac{2\pi}{\lambda} d_{k,n}^r}}_{\text{NLoS}}, \quad (1)$$

with λ being the carrier wavelength and the superscript $[\cdot]^r$ denoting the r -th multipath component from a total of $R+1$ components; specially, $\beta_{k,n}^0 \in \mathbb{R}^+$ denotes the LoS component. The Euclidean distance from the AU to the n -th LIS antenna is denoted by $d_{k,n}^0$, while $d_{k,n}^r$ denotes the distance of the n -th LIS' element to the r -th PE. The channel gain of the r -th multipath component can be modeled as [11]:

$$\beta_{k,n}^r = \frac{\lambda}{4\pi} \frac{\sqrt{\sigma_r} e^{-j\Delta\phi_{r,n}}}{(d_k^r + d_{k,n}^r)} \quad (2)$$

with d_k^r representing the distance between the k -th AU and the r -th PE, σ_r denoting the reflection loss by the r -th PE, and $\Delta\phi_{r,n} = 2\pi(d_k^r + d_{k,n}^r - d_{k,n}^0)/\lambda$ is the phase difference (Doppler shift) between the LoS and the r -th NLoS components. The parameter σ_r models a random variable subject to conductivity, relative permittivity, and permeability of the PEs, whose value can differ for different types of PEs.

III. RADIO MAPPING: AN OVERVIEW

Suppose the channel estimation (CHEST) phase of a LIS system. The K AUs transmit using K orthogonal pilots such that the received signal $\mathbf{Y} \in \mathbb{C}^{N \times K}$ at the LIS is:

$$\mathbf{Y} = p\mathbf{H} + \mathbf{W}, \quad (3)$$

where p is the UL transmit power, which is assumed to be equal to all the AUs, $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_K] \in \mathbb{C}^{N \times K}$ is the channel matrix, and $\mathbf{W} \in \mathbb{C}^{N \times K}$ is the receiver noise matrix whose entries are independent and identically distributed (i.i.d.) according to $\mathcal{CN}(0, \sigma_w^2)$. To exploit \mathbf{Y} for RMap, a spatial matched filter (MF) $\hat{\mathbf{h}} \in \mathbb{C}^{N \times 1}$ was proposed in [4], [5], whose filter coefficients are computed as:

$$\hat{h}_n = t_n e^{-j\frac{2\pi}{\lambda} d_n}, \quad \forall n \in \{1, \dots, N_f\}, \quad (4)$$

where t_n is the n -th element of a weighting vector $\mathbf{t} \in \mathbb{R}_+^{N_f \times 1}$, $e^{-j\frac{2\pi}{\lambda} d_n}$ defines a spherical wave phase-matching component, adjusted to the Euclidean distance $d_n = (d_{x,n}^2 + d_{y,n}^2 + d_{z,n}^2)^{1/2}$ between a reference (x, y, z) point in space and the n -th LIS antenna, and $N_f \leq N$ is the number of taps of the filter. Let denote the k -th column of \mathbf{Y} in Eq. (3) as $\mathbf{Y}_k \in \mathbb{C}^{N_x \times N_y}$ with $\mathbf{y}_k = \text{vec}(\mathbf{Y}_k)$. We also let $\hat{\mathbf{h}} = [\hat{\mathbf{h}}, \mathbf{0}] \in \mathbb{C}^{N \times 1}$ be the MF after zero-padding according to $\max(N_f, N)$. Then, the contribution $\mathbf{Y}_k \in \mathbb{C}^{N_x \times N_y}$ to the primary RM from the k -th AU is obtained as follows:

$$\mathbf{Y}_k = \hat{\mathbf{H}} \otimes \mathbf{Y}_k, \quad (5)$$

where $\hat{\mathbf{h}} = \text{vec}(\hat{\mathbf{H}})$. The primary RM $\mathbf{Y} \in \mathbb{C}^{N_x \times N_y}$ is [4]:

$$\mathbf{Y} = \left(\sum_{k=1}^K \mathbf{Y}_k \right) \otimes \hat{\mathbf{H}}. \quad (6)$$

When designing the filter, the authors of [4] considered the components $d_{x,n}$ and $d_{y,n}$ of d_n to be the distances of each n -th LIS element to the center of the room, while the focal height $d_{z,n}$ is a parameter subject to design. Further processing can be cast over \mathbf{Y} to perform sensing and localization.

We highlight two main drawbacks in the pre-processing from [4]. First, by using the combination in Eq. (6), the orthogonality among the signals from different AUs is lost. This now non-orthogonal superposition may incur loss of relevant information, as the PEs are being irradiated from different angles. This can produce different Dopplers' shifts, as indicated by Eq. (2), which can in turn mitigate the contribution of the NLoS components of interest. In other words, the contribution of the PEs can be *overshadowed* in the RMs. Second, the filter structure is capable of matching the phase of the LoS components, but outputs some *distortion*, especially in the neighborhood where the LoS ray impinges.

To illustrate, we consider the scenario described in Appendix A, where we have $K = 9$ AUs and $R = 13$ PEs with $R_{\text{obj}} = 3$ of them being cylindrical metallic objects and $R_{\text{hum}} = 10$ humans. Fig. 2 contains (a) the ground truth RM, where the AUs are represented by red crosses, while the PEs by the geometrical forms – metallic objects by the circles and the humans by the rectangles – (b) the RM of a single AU; and (c) the RM of superposed signals of the $K = 9$ AUs. The MF filter used was the same as in [4], with $t_n = 1/d_n$ computed as in (4). Fig. 2(b) exposes the effect of the distortion around the LoS components with power as high as the rays reflected by PEs. Fig. 2(c) shows the behavior when the signals are superposed. Specifically, we see how the metallic objects are highlighted while the humans get occluded, making it more difficult to correctly detect the latter.

IV. ONRMAP

Here we present OnRMap, an online RMap approach based on classical signal processing theory, which removes the need

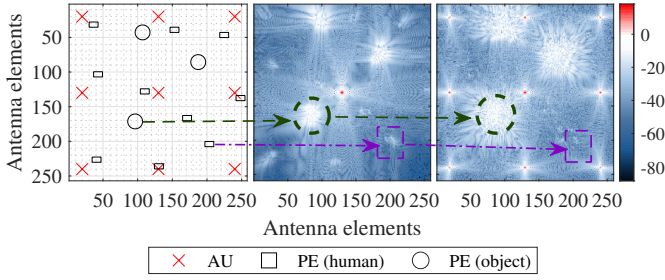


Figure 2. RMs showing one drawback of the MF used in [4] in the indoor scenario specified in Appendix A with $K = 9$ AUs and $R = 13$ PEs, where $R_{\text{obj}} = 3$ are metallic objects and $R_{\text{hum}} = 10$ are humans. *Left*: ground-truth. *Center*: single-AU $K = 1$; *Right*: superposed $K = 9$ AUs. When the signals are superposed, it is noticeable that PEs with higher reflection loss (humans w/ violet line) get occluded, while the others (objects w/ green line) got enhanced.

for offline scanning phases from [4], [5]. We detail the four stages that comprise OnRMap until we sense and locate the AUs and PEs, with an emphasis on the PEs, which are the most challenging to be detected.

A. OnRMap: An Overview

OnRMap consists of four steps, illustrated in Fig. 3.

Step 0. Signal Acquisition and Primary RM. In this initial step, we obtain the contributions \mathbf{Y}_k from the AUs $\forall k \in \{1, 2, \dots, K\}$ to the primary RM by filtering the signals \mathbf{Y}_k with the MF given in Eq. (5). Different from [4] and based on the discussion made in Section III, we empirically chose the weight vector \mathbf{t} using a two dimensional (2-D) Taylor window [12]. This selection was motivated by the empirical reduction of filtering distortion and numerical improvement in the ratio between the NLoS components. The output of this step is the matrix $\tilde{\mathbf{Y}} \in \mathbb{R}^{N \times K}$ containing the K primary RMs stacked in columns, *i.e.*, $\tilde{\mathbf{Y}} = [|\text{vec}(\mathbf{Y}_1)|, |\text{vec}(\mathbf{Y}_2)|, \dots, |\text{vec}(\mathbf{Y}_K)|]$, where $|\cdot|$ here denotes element-wise absolute value.

Step 1. Estimation of the LoS and NLoS Components. Through this step, the primary RM in $\tilde{\mathbf{Y}}$ is the input to the robust principal component analysis (RPCA) algorithm [13], which outputs the matrices $\hat{\mathbf{Y}}^{\text{LoS}}$ and $\hat{\mathbf{Y}}^{\text{NLoS}}$ corresponding to the estimations of the LoS and NLoS components of $\tilde{\mathbf{Y}}$, respectively. We employ this method based on the observation that the NLoS components have similarities among themselves, *e.g.*, the range of the power gain and location. In contrast, LoS components are a few data points with high power gains that are normally far apart. Hence, we can interpret the NLoS components to be low-rank components of $\tilde{\mathbf{Y}}$, whereas the LoS elements are sparse. Thus, RPCA becomes useful, since it is a low-complexity method for estimating the low-rank components of a matrix. Based on the special focus on sensing and locating the PEs, this block outputs the low-rank estimation, $\hat{\mathbf{Y}}^{\text{NLoS}} \in \mathbb{R}^{N_y \times N_x}$.

Step 2. Separation of the NLoS Components. This step translates the NLoS estimation from the previous step into data points for the inference step. To do so, we input $\text{vec}(\hat{\mathbf{Y}}^{\text{NLoS}})$ in k-means clustering [14], which separates the data in two clusters that represent *foreground* (high power) and *background* (low power) classes differentiate by their power level.

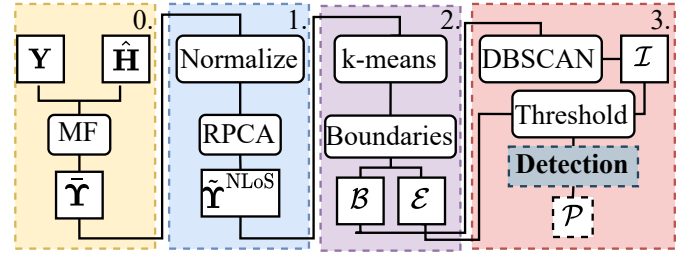


Figure 3. OnRMap architecture.

The foreground forms shapes in 2-D space and their perimeters are estimated with the Moore-Neighbor boundaries estimation algorithm [15]. The points that enclose the perimeters are stored in subsets that compound the set \mathcal{B} and the total power levels each shape comprises in $\tilde{\mathbf{Y}}^{\text{NLoS}}$ are stored in the set \mathcal{E} , constituting the output of this step.

Step 3. Sensing and Localization Inference. In this final block, we use the data in \mathcal{B} and \mathcal{E} to infer whether each subset in \mathcal{B} is a type of PE or just noise. If it is a PE, we can further identify which type of object it is, *e.g.*, a human or a metallic object, and their positions are also inferred. To do so, we employ Density-based spatial clustering of applications with noise (DBSCAN) clustering method [16] to cluster the data in \mathcal{B} based on the distance between samples. It is also assigned to each cluster its total power by looking at the set \mathcal{E} . The classification of each cluster on which type of PE (metallic object or human) follows a decision rule based on the power each cluster has, that is, lower and upper bounds are defined, and the clusters that fit in between are considered humans, those above are considered as objects, while the rest is noise.

B. OnRMap: Detailed Description

Below, we give further details on the other steps apart from Step 0, which was already detailed in (5).

1) *Estimation of the LoS and NLoS Components:* RPCA solves the following optimization problem [13]:

$$\begin{aligned} \min \quad & \|\mathbf{L}\|_* + \lambda_{\text{RPCA}} \|\mathbf{S}\|_1 \\ \text{s.t.} \quad & \mathbf{L} + \mathbf{S} = \mathbf{M}, \end{aligned} \quad (7)$$

where \mathbf{M} is the observation matrix, \mathbf{L} and \mathbf{S} are estimations of the low-rank and sparse components of \mathbf{M} , respectively, $\|\cdot\|_*$ and $\|\cdot\|_1$ are the nuclear and ℓ_1 norm of a matrix, respectively, and λ_{RPCA} is a scalar. In our case, from Step 0 we have the matrix $\tilde{\mathbf{Y}} \in \mathbb{R}^{K \times N}$, which is obtained by stacking and normalizing the $\text{vec}(\mathbf{Y}_k)$, $\forall k \in \{1, 2, \dots, K\}$. Then, we decompose it as:

$$\tilde{\mathbf{Y}} = \tilde{\mathbf{Y}}^{\text{NLoS}} + \tilde{\mathbf{Y}}^{\text{LoS}}. \quad (8)$$

Define $\hat{\mathbf{Y}}^{\text{NLoS}}$ and $\hat{\mathbf{Y}}^{\text{LoS}}$ as estimations of $\tilde{\mathbf{Y}}^{\text{NLoS}}$ and $\tilde{\mathbf{Y}}^{\text{LoS}}$, respectively. Then, substituting in Eq. (7), we have

$$\begin{aligned} \min \quad & \|\hat{\mathbf{Y}}^{\text{NLoS}}\|_* + \lambda_{\text{RPCA}} \|\hat{\mathbf{Y}}^{\text{LoS}}\|_1 \\ \text{s.t.} \quad & \hat{\mathbf{Y}}^{\text{NLoS}} + \hat{\mathbf{Y}}^{\text{LoS}} = \tilde{\mathbf{Y}}, \end{aligned} \quad (9)$$

placing the separation of LoS and NLoS components as equivalent to the optimization problem in (7). The output of the

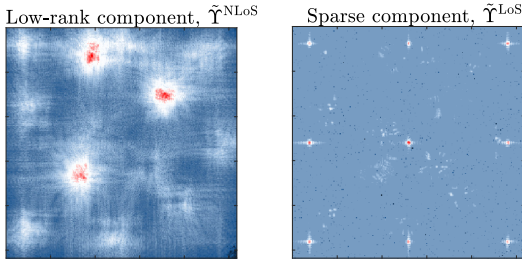


Figure 4. RPCA output. *Left*: low-rank matrix $\hat{\mathbf{Y}}^{\text{NLoS}}$ denoting the NLoS components. *Right*: sparse matrix $\hat{\mathbf{Y}}^{\text{LoS}}$ denoting the LoS components. We set the parameter $\lambda_{\text{RPCA}} = 3N^{-1/2}$ based on [13] and experimentation.

algorithm $\hat{\mathbf{Y}}^{\text{NLoS}} \in \mathbb{R}^{K \times N}$ is summed row-wise and reshaped to $\tilde{\mathbf{Y}}^{\text{NLoS}} \in \mathbb{R}^{N_y \times N_x}$; the same for the LoS-related matrix.

To illustrate how well RPCA performs, Fig. 4 shows the recovered low-rank (left) and sparse matrix (right). As can be seen, RPCA can estimate satisfactorily the NLoS (left) and LoS (right) components. On the other hand, we can point out two disadvantages of this technique. First, although there is a reference value for λ_{RPCA} , it may have to be tuned according to the scenario. Second, the problem of estimating low-rank and sparse components of matrices is NP-hard. RPCA solves a convex relaxed problem, incurring some information loss. However, most of the time, this method estimates satisfactory the components within less than twenty iterations. Despite that, for some realizations, part of the PEs was outputted in the sparse (LoS) component.

2) *Separation of the NLoS Components*: First, let's define the binary k-means clustering operation as $\mathfrak{K} : \mathbb{R}^{N \times 1} \mapsto \{0, 1\}^{N \times 1}$. Then, we perform entry-wise $\text{vec}(\mathbf{Y}^{\text{km}}) = \mathfrak{K}(\text{vec}(\tilde{\mathbf{Y}}^{\text{NLoS}}))$, with $\mathbf{Y}^{\text{km}} \in \{0, 1\}^{N_y \times N_x}$ being the matrix with the class of each data point in $\tilde{\mathbf{Y}}^{\text{NLoS}}$. In particular, we name class 0 as *background* containing points with null to very low power and class 1 as *foreground* containing components above a certain power threshold. We illustrate the output of the k-means \mathbf{Y}^{km} on the left side of Fig. 5. Note the creation of certain regions that can be interpreted as geometrical shapes. Visually inspecting and comparing the shapes with the ground truth, Fig. 2(a), we can infer that components that represent the metallic objects are more likely to be clustered in a larger and denser area, while the humans' ones are smaller and sparser.

Based on the observation made above, we employ the Moore-Neighbor boundaries estimation algorithm [15] to leverage these regions to detect the PEs. The boundary estimation algorithm works in the following way. First, a random point with value '1' (foreground) in \mathbf{Y}^{km} is picked up, defined as a central point, and its eight-point neighborhood values are checked. Then, the neighbors that have value '1' to this central point are considered to be in the same region as the central points. The algorithm starts by further searching considering the neighborhood of newfound points. Finally, a shape is defined when there are no other points left with the value '1' around the perimeter, where the outermost points are retrieved as a perimeter. Both perimeter and internal points are defined by (x, y) tuples representing the column and row in which they are allocated in the \mathbf{Y}^{km} matrix. The perimeter points are

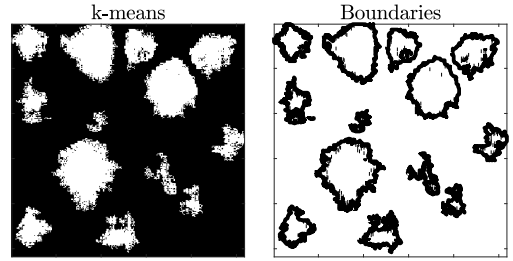


Figure 5. Graphic representation of *left*: the output of the binary k-means clustering, \mathbf{Y}^{km} and *right*: the estimated boundaries set \mathcal{B} obtained via Moore-Neighbor boundaries estimation.

assigned to a subset $\mathcal{B}_s \subset \mathcal{B}$ and all the points that define a shape in $\mathcal{A}_s \subset \mathcal{A}$, where s indexes a shape. Then, we associate a value E_s to each region s depicting the total power level within that region, which is calculated as:

$$E_s = \sum_{i \in \mathcal{A}_s} \left(\tilde{\mathbf{Y}}^{\text{NLoS}}[x_i, y_i] \right)^2. \quad (10)$$

The values E_s 's are further stored in the set \mathcal{E} . The boundaries for this scenario can be seen on the right side of Fig. 5.

3) *Sensing and Localization Inference*: This block receives the sets \mathcal{B} and \mathcal{E} and does the inference process. First, the data is clustered with DBSCAN [16]. This method randomly chooses a core point by observing the data in \mathcal{B} and maps the neighborhood subject to minPts and ϵ constraint parameters, representing the minimum number of points in a cluster and the search radius, respectively. Starting from the core point, the algorithm calculates the distances to all points in the data and assigns as a cluster those that satisfy the parameter constraints. The process is repeated until all the data have been clustered. Points that are sparsely distributed and thus cannot be assigned to a cluster, are considered noise. The indexes of the shapes that compound each cluster are assigned to the i -th cluster $\mathcal{I}_i \subset \mathcal{I}$ that maps both \mathcal{B} and \mathcal{E} sets.

We perform a test of the described algorithm on the illustrative scenario in Appendix A with $\text{minPts} = 2$ and $\epsilon = 2$. The algorithm output 63 identified clusters, which is much higher than the true number of PEs in the environment. To overcome excessive number of clusters, we exploit the metallic objects' lower reflection loss than the humans, $|\sigma_r| \ll |\sigma_{r'}|$, so we can infer that clusters with high power are objects, while clusters with low powers are noise or background distortion; consequently, clusters in the middle represent human positions. Thus, the i -th cluster is considered to be of a human class if it satisfies the following rule over \mathcal{E} :

$$\mathfrak{D}(i) = \begin{cases} 1, & \text{if } \text{th}_{\min} < \sum_{s \in \mathcal{I}_i} \mathcal{E}_s \leq \text{th}_{\max}, \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

where th_{\min} and th_{\max} are the minimum and maximum threshold parameters, respectively. If the cluster passes the threshold rule, then the cluster centroid is stored in a new *set of detected humans* \mathcal{H} .

To evaluate how reasonable the inference is, let $\mathbf{a} \in \mathbb{R}^{+R_{\text{hum}} \times 1}$ be the vector containing the differences between

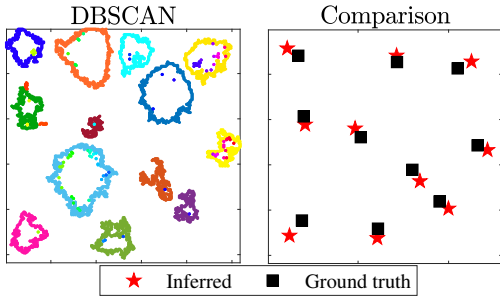


Figure 6. Inference step of OnRmap. *Left*: identified clusters by DBSCAN. The colors represent each cluster; the large ones are more likely to represent PEs, while the clusters with very few elements are likely noise. *Right*: inferred positions comparison against ground truth.

the inferred and the ground truth localization of the humans, giving the localization accuracy per human. Its r -th entry is calculated as:

$$a_r = \min \left(\left\{ \|\mathcal{H}_j - \mathcal{H}_r^{\text{gt}}\| < d_{\text{th}}, j = 1, 2, \dots, |\mathcal{H}| \right\} \right), \quad (12)$$

where \mathcal{H}_j denotes the j -th element of \mathcal{H} and \mathcal{H}^{gt} is the set of ground-truth humans with the r -th element denoted as $\mathcal{H}_r^{\text{gt}}$ and whose cardinality is $|\mathcal{H}^{\text{gt}}| = R_{\text{hum}}$. Furthermore, $\|\cdot\|$ is the Euclidean norm. We adopt as $d_{\text{th}} = 1\text{m}$ the distance threshold for detection via experimentation. The value of accuracy a_r can be *null* in case the distance is higher than the one defined by the threshold; in this case, we assume $a_r = 0$. We define the localization accuracy, LA, and the detection rate, DR, as:

$$\text{LA} = \frac{1}{R_{\text{hum}}} \sum_{i=1}^{R_{\text{hum}}} a_i, \text{ and } \text{DR} = \frac{1}{R_{\text{hum}}} \sum_{i=1}^{R_{\text{hum}}} \mathbb{1}(a_i > 0), \quad (13)$$

where $\mathbb{1}$ is the indicator function.

By empirically setting set $\text{th}_{\text{min}} = 0.03 \frac{1}{S} \sum \mathcal{E}$ and $\text{th}_{\text{max}} = 0.8 \max(\mathcal{E})$, Fig. 6 shows how the data were clustered for the illustrative example of Appendix A. Overall, it was identified 67 clusters, where three have very high energy, corresponding to the metallic objects, ten with low to medium energy, corresponding to humans, and the rest with shallow energy, considered noise. To visually assess the quality of the results, we re-plot on the right side of the figure the ground truth points together with the inferred points. The algorithm was capable of inferring the human's positions with localization accuracy from 0.1m (best case) to 0.62m (worst case) and the average of 0.22m in this single realization.

V. NUMERICAL RESULTS

We now evaluate the effectiveness of OnRMap and compare it with the previous works [4], [5]. For a better evaluation, we consider a more complex indoor scenario as provided by Feko by Altair Engineering ray tracing, as used in [4]. Different from the system model and toy example presented, this scenario includes more reflections from the PEs and reflections from the ground and walls. Despite that, the other parameters are the same as in Appendix A with $R_{\text{obj}} = 3$ metallic objects and $R_{\text{hum}} = 10$ humans. *Throughout this section, we focus on showing results for human detection since the detection of AUs and metallic objects is less challenging.*

We start by illustrating a step-by-step realization of OnRMap in Fig. 7 with $K = 20$ AUs. From the figure, we can note that OnRMap can perform well even in this more unfavorable scenario, given the increase in multipath components. In Fig. 8, we better evaluate the performance of OnRMap by considering 1000 Monte Carlo simulations (MCSs). Fig. 8a shows the average localization accuracy and the average detection rate of the humans when considering different numbers of AUs. We compare these results with the ones presented in [4]. The trade-off in the comparison is that we can achieve higher localization accuracy for all numbers of AU configurations (K) at the cost of lower detection rates on average. It is possible to point out two causes of the lower detection performance compared to [4]. First, some information of the signals is filtered after RPCA and does not enter in the inference process, as opposed to [4] that treats the signals from all AUs. Second, when the boundaries among PEs from different classes are too close, DBSCAN classify them as one cluster and the decision rule considers both as just one type of PE. These are considered the costs of the proposed online method in view of the offline approach. The lack of *a priori* information to compensate static PEs as in [4], compromises the DR to some extent. However, we argue that the high applicability and the significant improvement in the overall LA justify the novelty of this work.

To better understand the impact of the number of AUs in the detection rate, Fig. 8b shows the complementary cumulative distribution function (CCDF) of the detection in terms of the number of correctly detected humans for different numbers of AUs K . Note that the effect of increasing K is to lower the variance on how many humans R_{hum} we can detect. However, the opposite occurs at the points of nine to ten correctly detected humans, highlighted in Fig. 8b. For $K = 20$, the probability of a perfect detection (ten humans) is 0.4 %, and when $K = 5$ it achieves 3.2 %. A reason for this is that higher K implies more signals reflecting on PEs that can be possibly distributed next to the center of the room, while the PEs next to the walls are less exposed, and those reflections start to be interpreted as *background* (distortion/noise) by the method.

VI. CONCLUSIONS

In this paper, we proposed OnRMap, an online method for sensing and localization in indoor environments equipped with LIS systems. The greatest advantages of OnRMap are due to the fact that it is based on signal processing techniques, making it an online method, that is, it does not rely on offline scanning phases. This makes the method more robust for applications where the environment is constantly changing. However, the online feature comes with the cost of an average lower detection rate. But even so, OnRMap turns out to have a fairly good location accuracy. Future works may improve the design of OnRMap to cover the observed weakness and better study its performance.

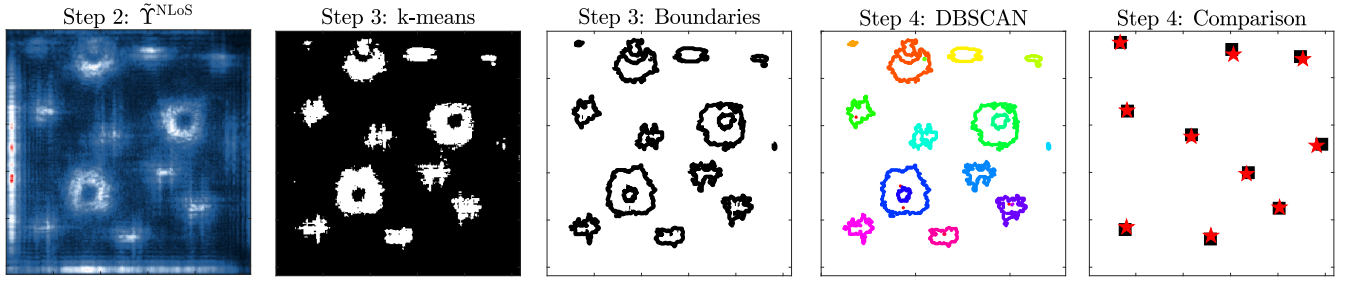
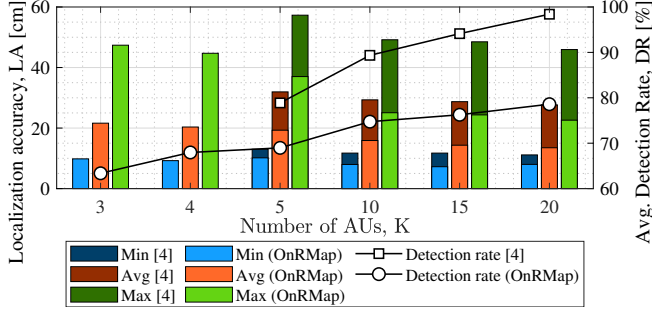
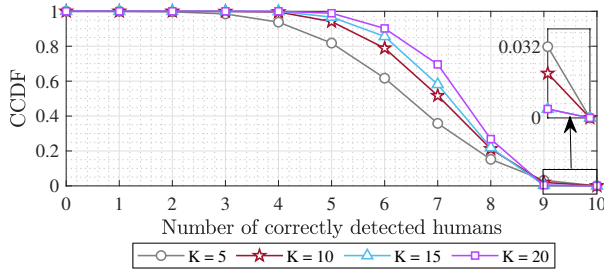


Figure 7. Exemplary application of OnRMap in the simulation environment from [4]. In Step 4: \star is inferred position and \blacksquare ground-truth.



(a) Average localization accuracy and average detection rate, defined in (13), for different numbers of AUs.



(b) CCDF of the number of correctly detected humans for different numbers of AUs.

Figure 8. Performance evaluation for the sensing (detection) and localization of $R_{\text{hum}} = 10$ humans (PEs).

APPENDIX A AN ILLUSTRATIVE INDOOR SCENARIO

The parameters of the simulation scenario are summarized in Table I. Note that we consider a scenario with two types of PEs: i) $R_{\text{obj}} = 3$ cylindrical *metal objects* with polished surfaces $\sigma_r \sim \mathcal{U}[-10, -15]$ dB and $R_{\text{hum}} = 10$ *humans* $\sigma_r \sim \mathcal{U}[-30, -75]$ dB, totaling $R = 13$ PEs.

REFERENCES

- [1] S. Hu, F. Rusek, and O. Edfors, "Beyond massive MIMO: The potential of data transmission with large intelligent surfaces," *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2746–2758, 2018.
- [2] D. Dardari, "Communicating with large intelligent surfaces: Fundamental limits and models," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 11, pp. 2526–2537, 2020.
- [3] C. J. Vaca-Rubio, P. Ramirez-Espinosa, K. Kansanen, Z.-H. Tan, E. De Carvalho, and P. Popovski, "Assessing wireless sensing potential with large intelligent surfaces," *IEEE Open Journal of the Communications Society*, vol. 2, pp. 934–947, 2021.
- [4] C. J. Vaca-Rubio, P. Ramirez-Espinosa, K. Kansanen, Z.-H. Tan, and E. de Carvalho, "Radio sensing with large intelligent surface for 6g," 2021. [Online]. Available: <https://arxiv.org/abs/2111.02783>

Table I
SIMULATION PARAMETERS

Parameter	Value	Parameter	Value
<i>Environment</i>		<i>Channel & System</i>	
# PEs	$R_{\text{obj}} = 3$ $R_{\text{hum}} = 10$ (y) = 10.36 [m]	Trans. power	$p = 20$ [dBm]
Room dim.	(x) = 10.36 [m] (z) = 8 [m]	Noise power	$\sigma_w^2 = -97$ [dBm]
	(y) = 0.3 [m] (x) = 0.5 [m] (z) = 1.7 [m]	LoS PL	$\beta_{k,n}^0 = \frac{\lambda}{4\pi d_{k,n}}$
Obj. dim	rad. = 0.43 [m] (z) = 2 [m]	NLoS ref. loss	$\sigma_r \sim \mathcal{U}[-10, -15]$ dB $\sigma_r \sim \mathcal{U}[-30, -75]$ dB
Hum. dim.	(y) = 0.3 [m] (x) = 0.5 [m] (z) = 1.7 [m]	Carr. freq.	$f = 3.75$ [GHz]
	LIS pos.	$z = 8$ [m]	Carr. wavel.
LIS elem.	$N_x = N_y = 259$	<i>Active users</i>	
		# AUs	$K = 9$
		AUs height	$z_k = 1.8$ [m]

- [5] C. J. Vaca-Rubio, R. Pereira, X. Mestre, D. Gregoratti, Z.-H. Tan, E. De Carvalho, and P. Popovski, "Floor map reconstruction through radio sensing and learning by a large intelligent surface," in *2022 IEEE 32nd International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2022, pp. 1–6.
- [6] X. Tong, Z. Zhang, Y. Zhang, Z. Yang, C. Huang, K.-K. Wong, and M. Debbah, "Environment sensing considering the occlusion effect: A multi-view approach," *IEEE Transactions on Signal Processing*, vol. 70, pp. 3598–3615, 2022.
- [7] X. Tong, Z. Zhang, J. Wang, C. Huang, and M. Debbah, "Joint multi-user communication and sensing exploiting both signal and environment sparsity," *IEEE Journal of Selected Topics in Signal Processing*, vol. 15, no. 6, pp. 1409–1422, 2021.
- [8] J. Wilson and N. Patwari, "Radio tomographic imaging with wireless networks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 5, pp. 621–632, 2010.
- [9] M. Zhao, T. Li, M. Abu Alsheikh, Y. Tian, H. Zhao, A. Torralba, and D. Katabi, "Through-wall human pose estimation using radio signals," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7356–7365.
- [10] A. Saleh and R. Valenzuela, "A statistical model for indoor multipath propagation," *IEEE Journal on Selected Areas in Communications*, vol. 5, no. 2, pp. 128–137, 1987.
- [11] A. Goldsmith, *Wireless Communications*. Cambridge University Press, 2005.
- [12] E. Brookner, *Practical Phased Array Antenna Systems*. Artech House, 1991.
- [13] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 3, jun 2011. [Online]. Available: <https://doi.org/10.1145/1970392.1970395>
- [14] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035.
- [15] R. Gonzalez, R. Woods, and S. Eddins, *Digital Image Processing Using MATLAB*. Pearson Prentice Hall, 2004.
- [16] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, ser. KDD'96. AAAI Press, 1996, p. 226–231.