Scalable and Efficient Clustering for Fingerprint-Based Positioning

Joaquín Torres-Sospedra, Darwin P. Quezada Gaibor, Jari Nurmi, *Senior Member, IEEE*, Yevgeni Koucheryavy, Elena Simona Lohan, *Senior Member, IEEE*, and Joaquín Huerta

***Abstract*—Indoor positioning based on IEEE 802.11 wireless LAN (Wi-Fi) fingerprinting needs a reference data set, also known as a radio map, in order to match the incoming fingerprint in the operational phase with the most similar fingerprint in the data set and then estimate the device position indoors. Scalability problems may arise when the radio map is large, e.g., providing positioning in large geographical areas or involving crowdsourced data collection. Some researchers divide the radio map into smaller independent clusters, such that the search area is reduced to less dense groups than the initial database with similar features. Thus, the computational load in the operational stage is reduced both at the user devices and on servers. Nevertheless, the clustering models are machine-learning algorithms without specific domain knowledge on indoor positioning or signal propagation. This work proposes several clustering variants to optimize the coarse and fine-grained search and evaluates them over different clustering models and data sets. Moreover, we provide guidelines to obtain efficient and accurate positioning depending on the data set features. Finally, we show that the proposed new clustering variants reduce the execution time by half and the positioning error by $\approx 7\%$ with respect to fingerprinting with the traditional clustering models.**

***Index Terms*—*k*-means, Bluetooth low energy (BLE), received signal strength (RSS), Wi-Fi, affinity propagation, clustering, fingerprinting, indoor localization.**

I. Introduction

NEW TECHNOLOGIES for indoor positioning have emerged to provide Quality of Experience (QoE) to the end-user by reducing the error in the position estimation [1], covering large areas [2], and providing energy efficiency [3]. Thus, these new technologies (e.g., Ultrawideband, ZigBee, Li-Fi, and Augmented Reality) have been used for positioning in universities, shopping malls, airports [4], [5], [6], [7], [8], [9], and embedded in Internet of Things (IoT) [10], [11], [12], [13] and wearable devices [14]. However, fingerprinting is a widely used technique despite its limitations in accuracy and scalability [15], [16].

Fingerprinting is built on top of a basic principle, the RSS values of a set of emitters—Wi-Fi access points (APs) or BLE beacons—are representative of the location where they were collected. It requires two stages to operate, the offline stage and the online stage. In the offline phase, many fingerprints are collected in the working area at different reference points, forming a radio map. In the online phase, the operational fingerprint (whose location is not known) is compared to all the reference fingerprints to estimate its position. Due to its simplicity, it is also gaining relevance for outdoor positioning in the IoT context using LoRaWAN or SigFox [17], [18], [19], [20].

Generally, the scalability problems in fingerprinting arise when the radio map contains thousands of fingerprints. The matching algorithm may inefficiently compute the distance of the operational fingerprint to all the reference samples [21]. This limitation has been addressed by means of data compression [22], integration of neighbor relative RSS and trajectory estimation (historical data) [10], and clustering models [23]. Among them, *k*-Means clustering has been widely used in fingerprinting to enhance accuracy and reduce search complexity in the online phase [1], [24], [25].

Although clustering models are popular in fingerprint-based positioning, they neither guarantee an equal distribution between the clusters [26], [27] nor consider the RF signal propagation features. In certain cases, the majority of samples may end up in a few clusters, generating oversized clusters, and thus, increasing the search time in the online phase. We proposed three variants for *k*-Means clustering that improved distribution between the clusters, reduced the computational load in the online phase of Wi-Fi fingerprinting, and provided similar accuracy as traditional *k*-Means [28]. The current work extends the analysis from [28] to seven clustering models and proposes four new variants to further improve efficiency while keeping similar performance, being the main contributions.

1) An extended analysis of three variants, namely, Variants I–III introduced in [28], with multiple clustering methods.

Manuscript received 23 March 2022; revised 5 August 2022 and 9 December 2022; accepted 16 December 2022. Date of publication 20 December 2022; date of current version 6 February 2023. This work was supported by the European Union's H2020 Research and Innovation Programme under the Marie Skłodowska-Curie under Agreement 813278 (A-WEAR, http://www.a-wear.eu/) and Agreement 101023072 (ORIENTATE, http://orientate.dsi.uminho.pt). *(Corresponding author: Joaquín Torres-Sospedra.)*

Joaquín Torres-Sospedra is with the ALGORITMI Research Centre, University of Minho, 4800-058 Guimarães, Portugal (e-mail: jtorres@algoritmi.uminho.pt).

Darwin P. Quezada Gaibor is with the Institute of New Imaging Technologies, Universitat Jaume I, 12071 Castellón, Spain, and also with the Electrical Engineering Unit, Tampere University, 33014 Tampere, Finland (e-mail: quezada@uji.es).

Jari Nurmi, Yevgeni Koucheryavy, and Elena Simona Lohan are with the Electrical Engineering Unit, Tampere University, 33014 Tampere, Finland (e-mail: jari.nurmi@tuni.fi; evgeny.kucheryavy@tuni.fi; elena-simona.lohan@tuni.fi).

Joaquín Huerta is with the Institute of New Imaging Technologies, Universitat Jaume I, 12071 Castellón, Spain (e-mail: huerta@uji.es).

Digital Object Identifier 10.1109/JIOT.2022.3230913

2) Four new variants for fingerprint-based clustering, Variants IV–VII, tested in traditional clustering models.
3) A comprehensive comparison between the seven variants on 25 multitechnology (BLE and Wi-Fi) data sets with recommendations on how to choose a certain variant and clustering model among the available ones.
4) Final validation over a huge LoRaWAN data set.
5) Application guidelines based on the data set features to obtain effective and efficient fingerprinting.

II. Related Work

Clustering has been widely used to group fingerprints with similar characteristics into classes [23], [29]. It helps to reduce the search area in the online phase of fingerprinting and the energy consumption in resource-constrained devices (i.e., devices with low energy, storage, and computational resources). However, not all the clustering models behave similarly and their performance depends on the area where the indoor positioning system (IPS) operates.

One of the most common clustering methods is *k*-Means, which has been applied in several studies in order to divide the data set into subdatasets, reduce the computational load in the user's device, and improve positioning accuracy. For instance, Anuwatkun et al. [29] combined *k*-Means clustering with the difference of signal strength (DIFF) method to improve the search time and accuracy in the position estimation. Lee and Lee [30] developed an algorithm to find the best *k* for *k*-Means, having the main objective to build an accurate radio map and provide a better position estimation.

Other clustering algorithms based on *k*-Means have also been tested for indoor positioning, as is the case of fuzzy *c*-means (FCM). Nevertheless, in contrast with *k*-Means, in FCM, one sample can belong to more than one cluster [31], [32], giving rise to overlapped zones or clusters. Moreover, FCM is not only used for its capability of dividing the data sets into fuzzy clusters, but also for its capability to model uncertainty in data [32]. For instance, Endo et al. [33] introduced a new clustering algorithm based on FCM for uncertain data. In this case, the authors used a quadratic regularization of penalty vectors to reduce the uncertainty in data and provide a better distribution of samples. It is essential to highlight that the performance of *k*-Means and FCM may vary according to the data set where they are applied and the proposed modifications.

Similarly, affinity propagation clustering (APC) has been applied to Wi-Fi fingerprinting radio maps, in order to divide the data set into multiple clusters. For example, Li et al. [34] proposed a two-level positioning algorithm, which uses APC to split the radio map into several subsets. However, Li et al. [34] used the Shepard similarity instead of the Euclidean distance to form the clusters. Unlike the Euclidean distance, Shepard similarity satisfies the logarithmic relationship between RSS and the distance, allowing a better computation of the similarity between fingerprints. Likewise, APC is applied for real-time positioning applications, according to the analysis done by [35] where the authors also determined that APC has a better performance than traditional clustering algorithms with less number of features. Additionally, some authors have proposed some modifications to this algorithm, testing different metrics in order to robustness of the algorithm [36]. e.g., Caso et al. [36] provided a novel mixed similarity metric, which improved the performance of APC in terms of computational time and positioning accuracy.

The *k*-Medoids method is widely used, given its capability to detect and exclude outliers [37]. For this reason, it is used to divide the Wi-Fi fingerprinting data set, providing a better data set partition, and a more accurate cluster centroid selection [38] than *k*-Means.

Clustering models based on density-based spatial clustering of applications with noise (DBSCAN) are more robust to noisy samples (outliers) present in radio maps than *k*-Means or *c*-Means. For instance, Wang et al. [39] introduced two new approaches to reduce the error in the position estimation when the Wi-Fi fingerprinting technique is used. One of these approaches is based on DBSCAN, namely, DBSCAN-KRF. The proposed algorithm (DBSCAN-KRF), thus, was capable of removing outliers and detecting insensitive areas that other algorithms cannot easily identify. Zhou et al. [40] used DBSCAN to achieve a better position estimation, along with F-test and T-test models.

In the literature, clustering has been mostly applied as a mere black box that tries to solve a problem without knowledge of signal propagation. In some works, authors have successfully modified traditional clustering models in order to introduce domain knowledge about signal propagation, such as using RSS differences [29] or a better distance metric [34], improving the accuracy as a result. Lin et al. [10] used the neighbor relative RSS to build the fingerprint database, adopting a Markov-chain model to predict the trajectory, assist positioning, and reduce energy consumption.

Instead of modifying a specific algorithm, using relative measurements or integrating previous predictions, we propose four new variants to better integrate the clusters generated by any traditional clustering model in fingerprinting. These variants can reduce positioning errors and/or improve execution time.

III. Materials and Methods

A. Implemented Clustering Models for Fingerprinting

All the clustering models have in common one key input, the radio map. The clustering models are usually unsupervised learning methods that process a set of samples in order to group them. In fingerprinting, this set of samples is the radio map. As output, they may provide either a centroid per cluster (a vector of mean/median RSS values); a representative per cluster (a single fingerprint that represents all the fingerprints in that cluster); or the cluster index per reference fingerprint. In the latter case, we computed the centroids for all the clusters if the clustering model did not provide them. Once the centroids or representatives are computed, the online search for the nearest neighbors in fingerprinting can be done in two stages: the first one is to select the most relevant cluster (namely, coarse search); and the second stage is to select the most relevant fingerprints within the winning cluster (namely,

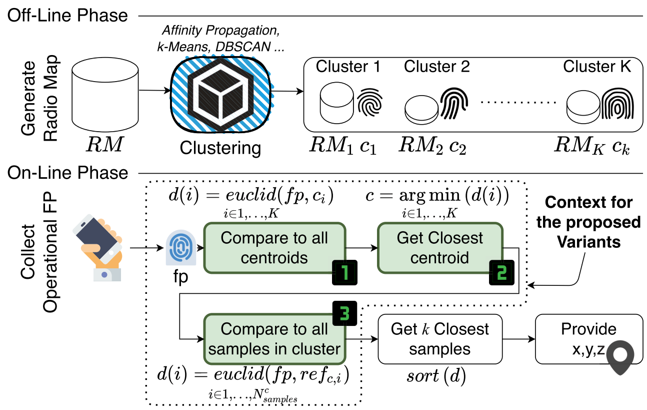


Fig. 1. Fingerprinting with clustering.

fine-grained search) to provide the position estimate. Fig. 1 provides an overview of fingerprinting with clustering, where the process to generate the clusters is often seen as a black box in the offline phase, i.e., the output of the clustering models from Sections III-A1–III-A7. The main issues of clustering in fingerprinting are shown in Section III-B.

In the figure, three main components are numbered 1–3. Those are the modules where the clustering integration can be improved, enhancing the efficiency of fingerprinting. The modifications introduced in [28] (Variants I–III) are described in Section III-D, whereas the four new modifications (Variants IV–VII) are introduced in Section III-E.

1) k-Means: k-Means is an unsupervised algorithm devoted to splitting the data sets into n nonoverlapping zones or classes, which can be represented in terms of a Voronoi diagram. Due to its simplicity, it has been widely used in multiple domains, including indoor positioning and localization. The k-Means method requires the number of clusters as an input parameter, whereas it provides as output the cluster centroids and the cluster indexes for all reference fingerprints. In this work, we use the centroid initialization proposed in [41] and the Manhattan distance as suggested in [15] as this approach is more robust, reducing the variability in the evaluation metrics over different runs.

2) k-Medoids: k-Medoids is a variant of k-Means, which is more robust to noisy samples (outliers) [38], and uses a representative fingerprint of the cluster (sample medoid) instead of the centroid (averaged sample) [37]. Both, k-Means and k-Medoids, have the same input and output parameters.

3) Fuzzy c-Means: Fuzzy c-Means, or simply c-Means, is another variant of k-Means. It introduces the concept of degree of association, allowing one sample to belong to more than one cluster [31]. As output, it also provides the matrix with the degree of membership between samples and clusters.

4) APC: APC bases its functionality on the level of similarity between fingerprints to form the clusters. In this case, the samples share two messages, the first one to determine whether a data point is suitable to be part of a cluster and the second message to indicate how appropriate is a data point as an examplar [42]. During this procedure, it is not required to establish any specific parameter related to the number of clusters. APC returns the cluster indexes and a representative fingerprint as the centroid.

5) DBSCAN: In contrast to k-Means, DBSCAN splits the data set into high-density and low-density classes, enabling outliers detection. It requires two parameters, *Eps* and *MinPts*, which determine the distance to form the neighborhood and the minimum number of samples to form a cluster [43]. DBSCAN only provides the cluster indexes as output.

6) Hierarchical Density-Based Spatial Clustering of Applications With Noise (HDBSCAN): Hierarchical density-based spatial clustering of applications with noise (HDBSCAN) is a variation of DBSCAN capable of adapting to clusters of different shapes and densities with only the *MinPts* parameter.

7) Model-Based Clustering: This method is also known as a mixture model as it combines different statistical models (e.g., Gaussian or Markov Mixture models). It determines the models and clusters based on the Bayesian information criterion (BIC). As a result, it provides the cluster indexes and other useful information [44], [45].

B. Common Issues of Clustering in Fingerprinting

In general, clustering models share the same objectives and they are considered general-purpose algorithms. Thus, we present some challenges that may affect their accuracy in fingerprint-based problems.

1) The RSS values may indicate the distance between the receiver and the emitter. However, the emitters' coverage area may not reach all the operational sites. In addition, the receivers' sensitivity depends on the device, having different cut-off thresholds. When an emitter is not detected, its RSS is filled with a default weak value, which represents any large emitter–receiver distance, a temporal interference or signal blockage, as well as a hardware limitation (e.g., unable to scan the 5-GHz band). Most of RSS values in the radio maps –97% in UJIIndoorLoc [46] are missing data which should not be used as geometric distance indicators.
2) The centroids usually correspond to the average or median RSS values of all the fingerprints in a cluster, which involves mixing real and arbitrary RSS values for nondetected emitters. The resulting centroid might not be representative for those APs/beacons that are close to the receiver (thus, having a strong RSS) but, simultaneously, having intermittent detection at the receiver side.
3) Averaging fingerprints may not be recommended when they are at significantly different locations/orientations, they are collected by different devices (with different antenna gains) or when the noise component in the signal propagation is high. Those measurable features are not explicitly considered by any clustering model.
4) Some clustering models do not provide centroids or representatives, just cluster indexes for all reference samples. In such a case, we have additionally computed the centroids for each cluster.

5) When searching for the most relevant centroid, the coarse search computes the distance of the operational fingerprint to all the centroids. This search may involve unrelated centroids, being inefficient [21].
6) The number of clusters is a key parameter. In some models, the number of clusters must be explicitly set (e.g., k in k-Means) whereas in other models it is obtained by the method itself (e.g., APC). Having a few clusters may not significantly reduce the computational load, whereas having a very large number of clusters may lead to a heavy selection of the most appropriate cluster.
7) The clustering models do not ensure that the fingerprints are equally distributed over the clusters, resulting in clusters much larger than the others. That is, the execution time depends on the operational fingerprint. In an ideal case, the clusters should be equally distributed, ensuring a similar computational cost for all operational fingerprints.
8) Splitting the whole radio map in disjoint subsets limits the information available near the cluster boundaries. As similar fingerprints may end up in different clusters, the information available for estimating the position will be limited to the fingerprints of the most relevant cluster, especially, in those cases where the operational fingerprint lies near the boundary between multiple clusters.

The previous issues can be addressed by acting on the three points highlighted in Fig. 1 by: 1) restricting the cluster search to relevant centroids; 2) improving the selection of the closest centroid(s); and 3) improving the selection of relevant fingerprints within the cluster(s). However, those actions should also be efficient. In the operational phase, any complex procedure to select relevant centroids/fingerprints may have a computational cost similar to or higher than the traditional fingerprint model without clustering. We introduce variants to the clustering workflow which can be integrated into any clustering model for fingerprinting, where the strongest AP is used as a discriminator due to its linear cost.

C. Terminology

For the k-nearest neighbors (k-NNs) model, we use two configurations: 1) *simple configuration* and 2) *best configuration* (see [15], [28]). In the former case, the plain 1-NN model is applied. In the latter, the k-NNs model with optimal hyperparameters (k, similarity function between fingerprints and data representation) is applied.

For those clustering models where the number of clusters must be defined, we have used three values: 1) 25; 2) *rfp1*; and 3) *rfp2* (see [15], [28], [47]). *rfp1* stands for the squared root of the number of samples in the radio map, whereas *rfp2* stands for the number of samples in the radio map divided by 25.

Ideally, the samples should be equally distributed in the clusters. However, we have realized that clusters are of heterogeneous sizes. We, thus, say that a cluster is *oversized* if it has much more samples (factor depending on the variant) than expected, being *expected size* defined as (n_{rfp}/c). That is, the number of samples in the radio map divided by the number of clusters.

We say that a cluster is relevant to the AP i, if the cluster has at least one relevant fingerprint to that AP. We define as relevant fingerprints to the AP i, those samples where the RSS value for AP i is among the strongest RSSs values in the sample. That is, a cluster is relevant to AP i if it has at least one reference fingerprint $fp = (r_1, \ldots, r_{na})$ for which $|r_{\max} - r_i| \leq \rho$, where na is the number of detected APs, $r_{\max}$ is the strongest RSS value of the reference fingerprint, and ρ is a threshold. Finally, we define as operative APs the set of APs that have, at least, one relevant cluster. This set is introduced to ensure any operational fingerprint will find relevant clusters to it.

As evaluation metrics, we use the positioning error ϵ and the execution time τ. As we are working with a set of data sets, we use the normalized metrics to the baseline (the plain k-NNs with *simple configuration*) for each data set as done in [47].

D. Previously Proposed Algorithms and Their Shortcomings

This section introduces the variants proposed in [28].

1) Variant I—Improved Coarse Search: The first variant is devoted to only limiting the coarse search to those clusters that are relevant to the strongest operative AP in the operational fingerprint. This limits the first search to those clusters with samples "near" the strongest operative AP (see Section III-C).

2) Variant II—Soft-Filtered Fine-Grained Search: Unlike Variant I, Variant II not only limits the coarse search to relevant clusters but also constrains the fine-grained search if the cluster size exceeds its expected size.

A cluster is oversized if it is four times the expected size, being the expected size defined as (n_{rfp}/c) (see Section III-C). In the fine-grained search, if the cluster is oversized $(4 \times (n_{rfp}/c))$, all fingerprints that do not contain a valid RSS value for the strongest AP in the operational fingerprint are ignored.

3) Variant III—Hard-Filtered Fine-Grained Search: This variant is very similar to Variant II. However, the main difference between Variant II and III is the way the clusters are post-processed for the fine-grained search.

In the fine-grained search, if the cluster is oversized, all the samples that are not relevant (see Section III-C) are ignored. That is, only relevant fingerprints are used in the second search. In Variants I–III, the relevant clusters, before and after post-processing if they were oversized, were precomputed for all APs in the offline phase of fingerprinting, finding the mapping functions f_1 and f_2 for the coarse-search and fine-grained search, respectively.

4) Variants I–III Assessment: An initial assessment of those variants over the k-Means algorithm was performed in [28] with 16 Wi-Fi data sets and using k-Means as the main clustering model, showing that the proposed variants have a better performance than the original model in terms of positioning error and/or execution time. In this case, Variants II and III with $\rho = 3$ provided the best tradeoff between normalized positioning error and normalized execution time (see Fig. 2).

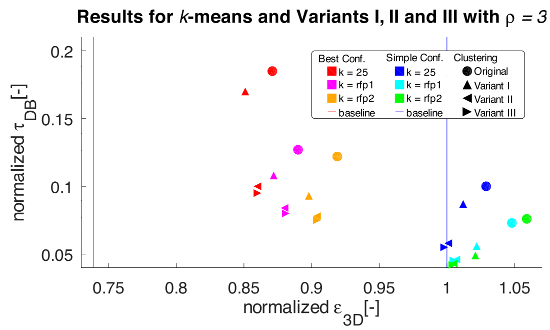


Fig. 2. Excerpt of the results provided in [28]. The color indicates k and configuration, and the shape indicates the variant.

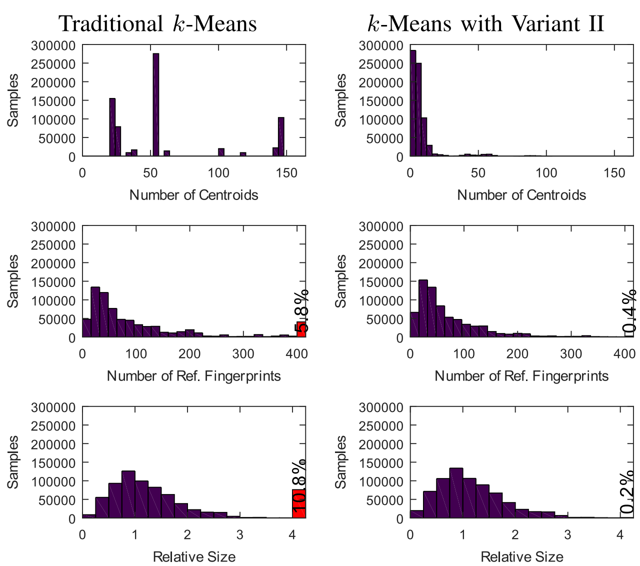


Fig. 3. Histograms of the number of cluster's centroid comparisons, fingerprint comparisons, and cluster size ratio with respect to equally distributed clusters for traditional k-Means ($k = rfp1$) and with Variant II ($\rho = 3$) using the experimental setup from [28]. In red are samples belonging to the last bin and beyond (percentage of total samples indicated).

k-Means with the proposed variants ($\rho = 3$) provides good general results in terms of normalized execution time ($\tilde{\tau}_{3D}$) for both configurations. However, it performs significantly worse than the baseline model for the *best configuration* ($\tilde{\epsilon}_{3D} = 0.74$). Future work should envisage keeping low computational costs while reaching the positioning accuracy of the baseline model, especially, for the *best configuration*. Additionally, the threshold for considering a cluster "oversized" was random and need to be explored.

E. Proposed New Algorithms to Enhance Fingerprinting

Before proposing the new variants, we analyze in Fig. 3 the traditional k-Means with $k = rfp1$ and the Variant II with $k = rfp1$ and $\rho = 3$, as it was pointed as optimal in [28].

According to the results reported in [28] (see Fig. 3), the coarse search in the original k-Means with $k = rfp1$ involved more than 150 distance computations in the largest data set, being above 50 in moderate-size data sets. With the suggested variant, the coarse search was below 20 in 95% of cases, which is a significant improvement with respect to the traditional k-Means. In terms of fine-grained search, the traditional k-Means clustering required more than 200 fingerprint comparisons in the fine-grained search in 15% of cases, whereas with the proposed variant it is reduced to just 5%. The presence of very large clusters was successfully mitigated with Variants II–III and most of the clusters are around their expected size. Despite this improvement, the cluster size was two times higher than expected in 10% of cases and contained less than half of the expected samples in 12% of cases. The applied variants reduced both, execution time and positioning error, with respect to traditional k-Means.

The 10.8% of operational fingerprints in [28] belong to an over-sized cluster. However, more than half of the cases belong to clusters above the expected size, suggesting that we should have defined more conservative rules about oversized clusters.

Given those results, we explore new variants with two objectives, ensuring an upper bound limit in the absolute execution time and reaching the performance provided by the baseline without clustering.

1) Variant IV—Soft-Filtered Fine-Grained Search in n Clusters: Variant IV is built on top of Variant II with a few minor changes that may improve efficiency and accuracy. As the variants we introduced in [28], it requires extracting the strongest operative AP of the operational fingerprint.

In the offline phase, Variant IV finds a function f_1 that maps each AP to the set of clusters that are relevant for it, storing all the mappings. A cluster is said to be relevant for the ith AP if the cluster has at least one fingerprint where AP i is among the strongest RSSs in the sample, as defined in Section III-C. Variant IV also finds a function f_2 that maps each AP and each cluster to the fingerprints that will be used in the fine-grained search. If the cluster is not oversized, all the fingerprints within the cluster will be used. If the cluster is oversized, only the fingerprints that contain a valid RSS value for that AP will be used. In Variant IV, a cluster is considered over sized if it has more than (n_{rfp}/c) samples, being n_{rfp} the number of samples in the radio map and c the number of clusters. That is, it is oversized if it has more samples than expected in an equally distributed clustering.

In the online phase, the coarse search consists in computing the distance between the relevant centroids and the operational fingerprint (step 1 in Fig. 1). The strongest operative AP in the operational fingerprint is used to select the relevant centroids. Then, we select the n clusters whose centroids reported the lowest distance to the operational fingerprint (step 2 in Fig. 1). Finally, the fine-grained search is done over the fingerprints belonging to the n selected clusters (step 3 in Fig. 1). If the clusters are oversized, all fingerprints that do not contain a valid RSS value for the strongest operative AP in the operational fingerprint are ignored.

The procedure to select the relevant clusters and fingerprints in Variant IV, f_1 (coarse), and f_2 (fine-grained) mappings, is very fast as getting the strongest AP is $O(na)$ and the threshold for the oversized cluster is small. Given the RSS variability, an alternative would be to look at the set of common [21] or strongest APs [48]. However, these alternatives would reduce the efficiency as more distance calculations are needed (more

TABLE I
MAIN FEATURES OF THE SELECTED DATABASES AND RESULTS USING THE 1-NN MODEL AND THE k-NNS WITH OPTIMAL HYPERPARAMETERS

											Simple Conf.				Best Conf.					
	DB	$\lvert\mathcal{T}\rvert$	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{A}\rvert$	$\lvert\mathcal{P}\rvert$	δ_{fp}	Dimension/Area	#f	#b	$\delta^{\mathcal{T}}_{2D}$	$\bar{\epsilon_{3D}}[\mathrm{m}]$	$\bar{\tau_{3D}}[\mathrm{s}]$	$\hat{\epsilon_{3D}}[-]$	$\hat{\tau_{DB}}[-]$	data	dist	k	$\hat{\epsilon_{3D}}[-]$	$\hat{\tau_{DB}}[-]$	refs.
Wi-Fi DBs	DSI1	1369	348	157	230	6	100 m × 18 m	1	1	0.73 ± (0.27)	4.95	12.23	1.00	1.00	pow	sorensen	11	0.77	1.15	[49]
	DSI2	576	348	157	230	3	100 m × 18 m	1	1	0.31 ± (0.11)	4.95	5.18	1.00	1.00	pos	PLGD10	9	0.77	2.97	[49]
	LIB1	576	3120	174	48	12	15 m × 10 m	2	1	1.21 ± (0.29)	3.02	46.25	1.00	1.00	pos	SQeuclidean	11	0.82	0.93	[50]
	LIB2	576	3120	197	48	12	15 m × 10 m	2	1	1.21 ± (0.29)	4.18	46.17	1.00	1.00	pos	PLGD10	9	0.54	3.03	[50]
	MAN1	14300	460	28	130	110	50 m × 36 m	1	1	20.88 ± (4.48)	2.82	156.01	1.00	1.00	exp	cityblock	11	0.73	1.00	[51], [52]
	MAN2	1300	460	28	130	10	50 m × 36 m	1	1	1.90 ± (0.41)	2.47	14.37	1.00	1.00	exp	neyman	11	0.75	1.55	[51], [52]
	MINT1	4973	810	11	189	19	1000 m^2 total	1	1	9.76 ± (4.08)	2.67	96.10	1.00	1.00	pow	PLGD10	11	0.80	2.68	[53]
	SAH1	9291	156	775	9291	1	4184 m^2 total	3	1	0.88 ± (0.51)	9.07	41.23	1.00	1.00	exp	neyman	11	0.79	1.62	[54]
	SIM	10710	1000	8	1071	10	50 m × 20 m	1	1	7.65 ± (1.50)	3.24	254.25	1.00	1.00	exp	SQeuclidean	11	0.74	0.91	[15]
	TIE1	10633	50	613	10633	1	5432 m^2 total	6	1	1.05 ± (0.54)	6.55	14.71	1.00	1.00	pos	PLGD40	11	0.37	3.38	[54]
	TUT1	1476	490	309	1476	1	124 m × 57 m	4	1	0.13 ± (0.06)	9.59	18.88	1.00	1.00	pos	PLGD40	3	0.46	3.12	[25], [55]
	TUT2	584	176	354	584	1	145 m × 88 m	3	1	0.04 ± (0.02)	14.37	2.76	1.00	1.00	pow	sorensen	1	0.56	1.16	[25], [55]
	TUT3	697	3951	992	694	1	130 m × 62 m	5	1	0.07 ± (0.04)	9.59	79.50	1.00	1.00	pos	sorensen	3	0.89	1.18	[56]
	TUT4	3951	697	992	3843	1	130 m × 62 m	5	1	0.36 ± (0.17)	6.36	79.87	1.00	1.00	pos	PLGD10	3	0.85	3.67	[56]
	TUT5	446	982	489	446	1	85 m × 145 m	3	1	0.01 ± (0.00)	6.92	11.98	1.00	1.00	pos	PLGD40	3	0.76	3.26	[57]
	TUT6	3116	7269	652	3116	1	135 m × 62 m	4	1	0.33 ± (0.16)	1.94	624.81	1.00	1.00	pos	sorensen	1	0.98	1.17	[58]
	TUT7	2787	6504	801	2787	1	88 m × 137 m	3	1	0.27 ± (0.16)	2.69	511.79	1.00	1.00	pos	sorensen	1	0.83	1.17	[58]
	UJI1	19861	1111	520	933	20	108 703 m^2 total	5	3	0.72 ± (0.43)	10.81	599.87	1.00	1.00	pow	sorensen	11	0.61	1.16	[46]
	UJI2	20972	5179	520	1967	1	108 703 m^2 total	5	3	0.73 ± (0.44)	8.05	2938.38	1.00	1.00	exp	neyman	11	0.76	1.59	[46]
	UTS1	9108	388	589	1466	3	44 000 m^2 total	16	1	0.46 ± (0.29)	8.74	96.32	1.00	1.00	exp	neyman	11	0.80	1.62	[59]
BLE DBs	UEXB1	417	102	30	139	3	1000 m^2 total	4	1	0.16 ± (0.06)	3.71	1.05	1.00	1.00	exp	neyman	3	0.86	1.55	[60]
	UEXB2	552	138	30	184	3	1800 m^2 total	5	1	0.16 ± (0.08)	4.65	1.86	1.00	1.00	pos	SQeuclidean	3	0.93	0.93	[60]
	UEXB3	240	60	30	120	2	5800 m^2 total	5	1	0.04 ± (0.02)	7.14	0.36	1.00	1.00	pos	SQeuclidean	3	0.92	0.92	[60]
	UJIB1	732	900	24	24	30	151 m^2 total	1	1	3.08 ± (0.76)	3.05	15.95	1.00	1.00	exp	neyman	11	0.54	1.55	[61]
	UJIB2	576	240	22	24	24	176 m^2 total	1	1	1.83 ± (0.37)	4.33	3.39	1.00	1.00	pos	LGD	11	0.58	1.86	[61]

APs involved) and the computational cost of sorting APs in fingerprints is $O(na \cdot \log na)$.

This variant requires two parameters, ρ and n. The former controls when a fingerprint is relevant to an AP, being $\rho = 0$ the most strict but also the most efficient in terms of computational time. The latter controls how many clusters are included in the fine-grained search. The larger n, the more clusters are involved and the higher the computational cost.

2) Variant V—Hard-Filtered Fine-Grained Search in n Clusters: Variant V applies the same steps as Variant IV, but has a more restrictive f_2 mapping equation.

The only difference with respect to Variant IV is in the fine-grained search. Suppose a cluster is oversized in Variant IV. In this case, all the fingerprints in that cluster that are not relevant for the strongest AP in the operational fingerprint (as defined in Section III-C) are ignored, and therefore, not used in the fine-grained search.

3) Variant VI—Soft-Filtered Fine-Grained Search With O(1): Despite the efforts introduced in Variants I–V, none of them can guarantee a constant computational cost for any operational fingerprint, as some clusters may be much larger than others. Variant VI limits the centroids distance computations to n and the fingerprint distance computations to m, ensuring a cost of $O(1)$ across all data sets.

In the offline phase, the number of relevant fingerprints (see Section III-C) is computed for each AP and each cluster. For a particular AP, its f_1 mapping function includes only the top n clusters. That is, those clusters with the highest number of relevant fingerprints within. Clusters without any relevant fingerprint are ignored, ensuring that the coarse search will mostly consist of n distance computations.

The f_2 mapping function considers all fingerprints of the cluster if it is not oversized. If the cluster is oversized, only m random relevant fingerprints from all fingerprints that contain a valid RSS value for the strongest AP in the operational fingerprint are included in f_2. In Variant VI, a cluster is considered oversized if it has more than m samples.

Variant VI requires three parameters, ρ, n, and m. ρ controls when a fingerprint is relevant to an AP as in the previous variants, whereas n and m limit the number of distance calculations in the coarse and fine-grained searches.

4) Variant VII—Hard-Filtered Fine-Grained Search With O(1): Variant VII applies the same steps as Variant VI with a more restrictive f_2 mapping function in the offline stage.

The only difference with respect to Variant VI is in the fine-grained search. If the cluster contains more than m fingerprints, just m random relevant fingerprints for the strongest AP in the operational fingerprint are selected.

F. Wi-Fi and BLE Fingerprinting Data Sets

The fingerprint-based models are commonly assessed in a controlled environment using a private data set. The new trends approach the machine-learning assessment using multiple diverse data sets. In this work, we have extended the 16 data sets used in [15], [28], and [62] by adding four Wi-Fi and five BLE data sets.

Table I introduces the main features of the 25 data sets used in this work. The table includes the number of reference and evaluation samples ($|\mathcal{T}|$, $|\mathcal{V}|$), the number of APs detected ($|\mathcal{A}|$), the number of reference locations ($|\mathcal{P}|$), the number of samples in each reference location (δ_{fp}), the size of the operational area, the number of floors considered in multistorey locations, and the density of samples around every reference

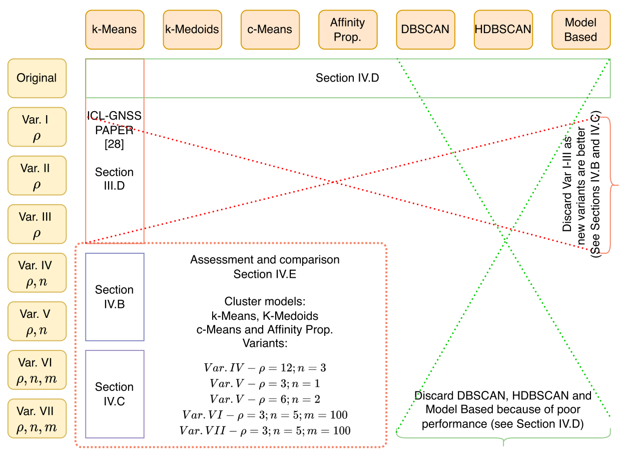


Fig. 4. Diagram with the distribution of the empirical experiments performed.

location ($\delta^{\mathcal{T}}_{2D}$). This density indicates the average and standard deviation of the number of fingerprints per m^2 that are around the reference points [15].

Additionally, Table I includes the results with the plain 1-NN model (*simple configuration*) and an optimized *k*-NNs model (*best configuration*) are provided. The results are normalized for each data set to the baseline. In this article, the baseline corresponds to the results obtained with *k*-NNs with *simple configuration*. That is, using $k = 1$, the Manhattan distance and positive data representation [28].

IV. Experiments and Results

This section is devoted to describing the experiments done and reporting the empirical results we have obtained. As the number of clustering models, variants, and parameters explodes, we distributed them in four rounds (see Fig. 4). For Variants I–III the results with 16 data sets and *k*-Means were published in [28] and updated for all 25 data sets in Sections IV-B and IV-C. For Variants IV–VII, the results with the 25 data sets and *k*-Means are shown in Sections IV-B and IV-C. The feasibility of traditional clustering models is analyzed in Section IV-D. The final results on the best clustering models and best variants are reported in Section IV-E.

A. Experimental Setup

We have used the experimental setup defined in [28]. It has the *k*-NNs algorithm as core IPS, two sets of hyperparameters for *k*-NNs (*simple configuration* and *best configuration*), the seven clustering models, 25 data sets and 10 execution runs. The clusters have been randomly generated ensuring that the implemented clustering models and all the proposed variants share the same initialization for each data set and execution run. The experiments were performed in a computer with Intel Core i7-8700 CPU, 16 GB of RAM, and Octave 4.0.3.

The hyperparameters for *k*-NNs are the RSS representation, and the *k* value and the distance function for *k*-NNs. *Simple configuration* stands for $k = 1$, Manhattan distance and positive data representation. *Best configuration* stands for the hyperparameter configuration that reported the lowest positioning error for a data set after evaluating 144 alternatives [15].

For *k*-Means, *k*-Medoids, and *c*-Means, we used three different values (25, *rfp*1, and *rfp*2) for the number of generated clusters (*k* or *c*). With the first value, the model generates 25 clusters, whereas *rfp*1 and *rfp*2 refer to heuristics based on the number of reference fingerprints in the radio map (see Section III-C). In DBSCAN and HDBSCAN, we selected the optimal values for *MinPts* and *Eps*.

The results collected for this article are the mean 3-D positioning error (ϵ_{3D}) and the computational time (τ_{DB}) resulting from processing all the operational fingerprints. As some clustering models (e.g., *k*-Means) rely on random initialization, the positioning error and the execution time might vary among runs. Therefore, the empirical evaluation was repeated ten times. We summarize the results by providing the averaged values, $\bar{\epsilon}_{3D}$ and $\bar{\tau}_{DB}$, over the ten runs.

Due to the data set heterogeneity, we report the normalized values, $\hat{\epsilon}_{3D}$ and $\hat{\tau}_{DB}$, against the results from a baseline method, the plain 1-nearest neighbor (NN) with the *simple configuration* for each data set (see Table I). Then, the normalized values are averaged across all data sets to obtain the general normalized metrics, $\widetilde{\epsilon}_{3D}$ and $\widetilde{\tau}_{DB}$, as described in [47]. Thus, 25×10 absolute values are summarized into just 1 value per evaluation metric.

B. First Assessment of Variants IV and V

First, we assess Variants IV and V. We have used all the 25 data sets and the clusters have been generated with *k*-Means clustering. The experiments have been repeated ten times. The results for Variants I–III also considers the 25 data sets, updating the results published in [28].

As the number of variants and parameters (5ρ values and $3n$ values) is considerably high, we restrict the assessment to visual analysis. Fig. 5 shows a scatter plot where the normalized error is compared to the normalized execution time for the extreme values $\rho = 3$ and $\rho = 12$ as the results reported on [28] showed that the former provided the best tradeoff and the latter provided the best general positioning accuracy. The results of the original *k*-Means model as well as the results for Variants I–III are also included for reference.

Several conclusions can be drawn from the scatter plots. First, most of the variants improve the traditional *k*-Means clustering in both dimensions ($\widetilde{\epsilon}_{3D}$ and $\widetilde{\tau}_{DB}$). However, the new Variant IV provides worse execution times than the traditional clustering when $n = 3$. As the value of n increases, the fingerprints from more clusters are considered in the fine-grained search, and therefore, the execution time increases. Variant V shows a similar behavior, but the cluster post-processing is more aggressive when it is over-sized than Variant IV. Thus, it provides better execution time at the expense of a worse positioning error.

For both new variants, $n = 1$ is providing similar results to Variant II and Variant III, respectively. For $n > 2$ both new variants provide excellent accuracy, but at the expense of high computational costs. Balancing positioning error and execution time, $n = 2$ seems to provide optimal results.

The election of the optimal value of ρ is a critical step in Variants IV and V. The lower the threshold, the more strict

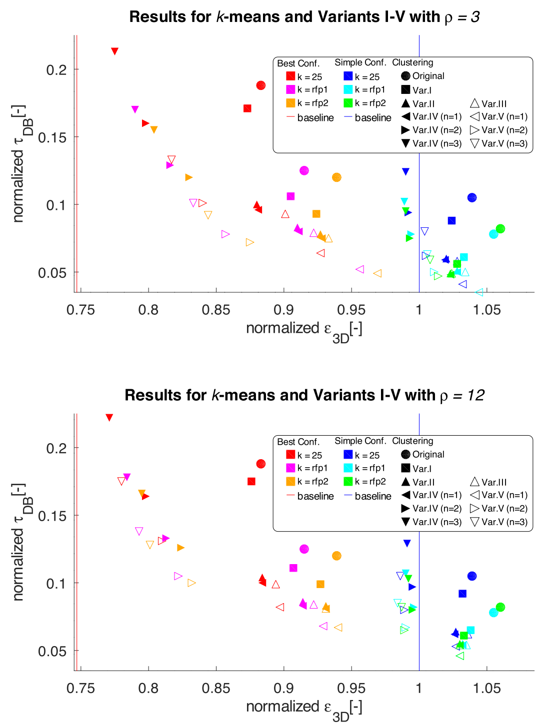


Fig. 5. Excerpt of the results provided by Variants IV and V. The color indicates k and configuration, and the shape indicates the variant.

the election of relevant fingerprints, and therefore, the lower number of reference fingerprints analyzed in the fine-grained search. However, execution time is decreased at the expense of increasing the error. Applying the rule of the elbow, a ρ value between 3 and 6 provides a good tradeoff between both metrics, $\widetilde{\epsilon_{3D}}$ and $\widetilde{\tau_{DB}}$.

Comparing Variants IV and V, we cannot foresee a clear winner. Variant IV is providing a better positioning error at the expense of a higher execution time, whereas Variant V is computationally better at the expense of a worse positioning error. Considering all the possible combinations, Fig. 5 shows that $\widetilde{\epsilon_{3D}}$ and $\widetilde{\tau_{DB}}$ may have a strong negative correlation. We confirmed our assumption with Pearson's correlation for both variants, which reported a correlation coefficient of -0.91 for Variant IV and -0.87 for Variant V. Thus, we can conclude that our new proposed variants provide a wide range of solutions, from a computationally efficient one to an accurate one, that improve the traditional k-Means clustering.

Finally, the new proposed Variants IV and V are reporting the best overall alternatives. Variant V with $n = 1$ and $\rho \leq 3$ would fit better for applications requiring low latency and a good position estimation, whereas Variant IV with $n = 3$ and $\rho = 12$ would fit better in those applications requiring the best positioning without a significant delay. Variant V with $n = 2$ and $\rho = 6$ presents good balance between $\widetilde{\epsilon_{3D}}$ and $\widetilde{\tau_{DB}}$.

C. *First Assessment of Variants VI and VII*

Similarly, we assess Variants VI and VII using visual analysis. Fig. 6 shows an scatter plot where $\widetilde{\epsilon_{3D}}$ is compared against $\widetilde{\tau_{DB}}$ for the ρ values 3 and 12. We set the values $n = 5$ and $m = 100$ in the new variants. This means that the proposed

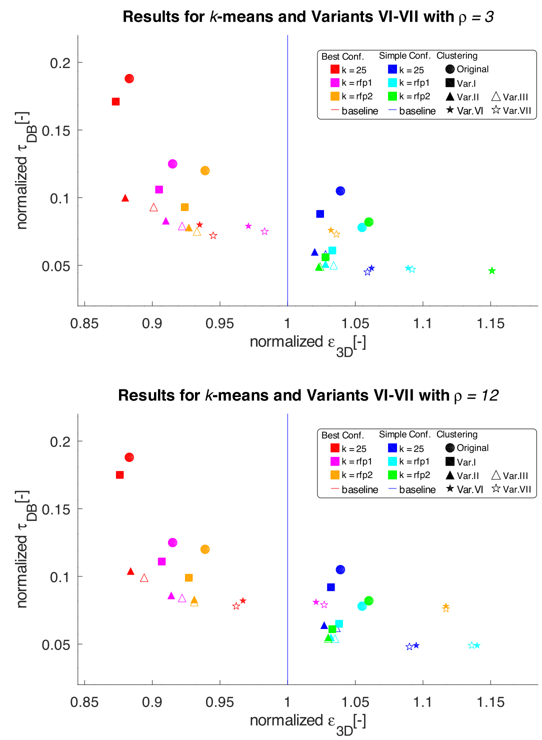


Fig. 6. Excerpt of the results provided by Variants VI and VII. The color indicates k and configuration, and the shape indicates the variant.

variants restrict the coarse search to the five most relevant clusters and the fine-grained search to 100 random relevant reference fingerprints in both new variants, being the concept of relevant fingerprint for the fine-grained search the main difference between both.

At first sight, we can see that the new Variants VI and VII are the worst-performing variants. Reducing the searches to 5 centroids and 100 fingerprints is not appropriate. In fact, the normalized error for the new variants ($k = rfp2$ and *simple configuration*) is over 1.15, and therefore, not plotted.

Regarding k, the number of clusters generated with k-Means, the performance degrades as k increases having a strong negative impact on the new Variants VI and VII. Only $k = 25$ reports results in phase with the traditional k-Means and Variants I-III, Variants VI and VII provide very poor positioning for $k = rfp1$ and $k = rfp2$. As the larger the value of k the smaller the clusters, this finding would suggest that restricting to just five relevant clusters is not enough or that the relevance function is failing when k is large.

The relation between the accuracy and ρ is inverse in the variants presenting an $O(1)$. That makes sense as larger values of ρ result in larger clusters and less relevant reference fingerprints are kept within the cluster. As the selection of the m ($m = 100$) reference samples is random, the probability of having a less related reference fingerprint in the reduced cluster is much higher as ρ increases. The optimal parameter for ρ is between 0 (strongly related fingerprints) and 3.

On the positive side, we can conclude that the results provided by Variants VI and VII are promising when the number of clusters is reduced (i.e., $k = 25$) and $\rho = 3$. They provide slightly worse normalized positioning accuracy to the

TABLE II
RESULTS REPORTED BY THE SELECTED CLUSTERING METHODS

method	params	Simple Conf.		Best Conf.	
		$\widetilde{\epsilon_{3D}}[-]$	$\widetilde{\tau_{DB}}[-]$	$\widetilde{\epsilon_{3D}}[-]$	$\widetilde{\tau_{DB}}[-]$
baseline	–	1.000	1.000	0.736	1.805
k-means	$k = 25$	1.039	0.105	0.883	0.188
k-means	$k = rfp1$	1.055	0.078	0.915	0.125
k-means	$k = rfp2$	1.060	0.082	0.939	0.120
k-medoids	$k = 25$	1.056	0.117	0.892	0.213
k-medoids	$k = rfp1$	1.070	0.082	0.920	0.133
k-medoids	$k = rfp2$	1.081	0.086	0.941	0.126
c-means	$k = 25$	3.616	0.204	3.015	0.373
c-means	$k = rfp1$	3.083	0.171	3.331	0.287
c-means	$k = rfp2$	1.449	0.142	1.435	0.271
Affinity Prop.	–	1.080	0.101	0.989	0.137
DBSCAN	–	1.723	0.152	1.905	0.296
HDBSCAN	–	1.809	0.277	1.927	0.396
model-based	–		0.188	2.080	0.458

traditional k-Means with $k = rfp2$ and similar normalized execution time than Variants II and III also with $k = rfp2$. Despite that selected new variants do not stand out in the plot, they are the only solutions reporting results in phase to traditional k-Means ensuring a constant computational complexity in the operational phase across all data sets. None of the operational fingerprints involved more than 105 vector comparisons, which was not guaranteed by Variants I–III.

Anyway, despite the low number of comparisons involved in both searches, $100 + 5$, the computational overload is not significantly lower with respect to Variants II and III. This shows that the computational cost of fingerprinting also depends on other factors that are external to fingerprint matching. Probably, we could have set slightly higher values for n and m, still ensuring a constant upper bound in the number of comparisons ($O(1)$), and therefore, a good $\tilde{\tau}_{DB}$ without decreasing the accuracy.

D. Assessment of the Clustering Models

Before assessing the proposed variants in all the clustering models, we evaluate them first on the original implementation as depicted in Fig. 1 without any modification/variant. Table II shows the results for the baseline method as well as the original clustering models applied to fingerprinting.

According to the results reported in Table II, k-Means, k-Medoids, and APC are providing the best results considering the two metrics. For those models, the accuracy is slightly worse than the baseline but the computational burden is significantly reduced (around ten times lower).

The other clustering models, namely, fuzzy c-Means, DBSCAN, HDBSCAN, and model-based, perform quite worse than the baseline in terms of positioning error without providing an extraordinary reduction of the computational costs. DBSCAN, HDBSCAN, and model-based did not successfully cluster all data sets, providing a unique cluster in a few cases. Moreover, DBSCAN and HDBSCAN labeled a significant number of reference fingerprints as noise in some data sets. Therefore, they are excluded in forthcoming analyses.

TABLE III
PROPOSED ALTERNATIVES TO IMPROVE CLUSTERING METHODS

Variant	Parameters	Reason
Variant IV	$\rho = 12$; $n = 3$	Best $\tilde{\epsilon}_{3D}$
Variant V	$\rho = 3$; $n = 1$	Best $\tilde{\tau}_{DB}$
Variant V	$\rho = 6$; $n = 2$	Good trade-off
Variant VI	$\rho = 3$; $k = 25$; $n = 5$; $m = 100$	$O(1)$
Variant VII	$\rho = 3$; $k = 25$; $n = 5$; $m = 100$	$O(1)$

E. Assessment of Selected Variants and Clustering Models

So far, the results have been presented for k-Means clustering, which resulted in complex tables/plots. As the number of parameters in the proposed variants adds complexity to the comparison, we focus the final assessment on the five cases detailed in Table III.

We selected those five configurations as they provide, respectively, the best general positioning accuracy, the best execution time, a good tradeoff between the two evaluation metrics and a constant computational cost ($O(1)$) for the fingerprints comparisons. For the last two selected variants, we considered that five relevant clusters and 100 relevant fingerprints per cluster was reasonable. None of the variants proposed in [28] (i.e., Variants I–III) has been included, as the new variants have shown more promising results.

1) Database Analysis: First, we independently analyze the results for each data set. As they are of different nature (location, collection strategy, and/or positioning technology) and they report different ranges on the positioning error and execution time, we have used the normalized values with respect to the baseline (the plain k-NNs algorithm) in the comparison shown in Fig. 7. In contrast to the previous section, the reported normalized values and baselines stand for the particular data set, i.e., we use $\hat{\epsilon_{3D}}$ and $\hat{\tau_{DB}}$ as main evaluation metrics for the comparisons.

According to Fig. 7, there are seven data sets (MAN1, SAH1, SIM, TUT6, UTS, UJI1, and UJI2) where clustering has considerably reduced the computational load. Clustering methods can reach the positioning error of the baseline with a significant execution time reduction of around 98%–99%, except in TUT6. It is worth noticing that the data sets in this first group have either a very large number of reference samples or they covered a large area (e.g., multistorey buildings) with a moderate/high density ($\delta_{2D}^{\mathcal{T}}$ in Table I) of fingerprints. Thus, the number of samples in the reduced radio maps is diverse enough even when the clustering model is generating a large number of clusters.

The second group is formed by DSI1, DSI2, LIB1, MAN2, TUT2, TUT3, TUT7, UEXB1–3, and UJIB1. The proposed alternatives present similar results to the first group but the execution time reduction 85%–95%, despite being significant, it was slightly lower than for the first group. Here, two data sets, TUT7 and UJI2, present a high difference in the execution time reduction of the proposed variants with respect to the traditional clustering methods without any optimization. Also, DSI2 presents interesting results as it is a clean version of DSI1, but the accuracy in the best case is not as good as

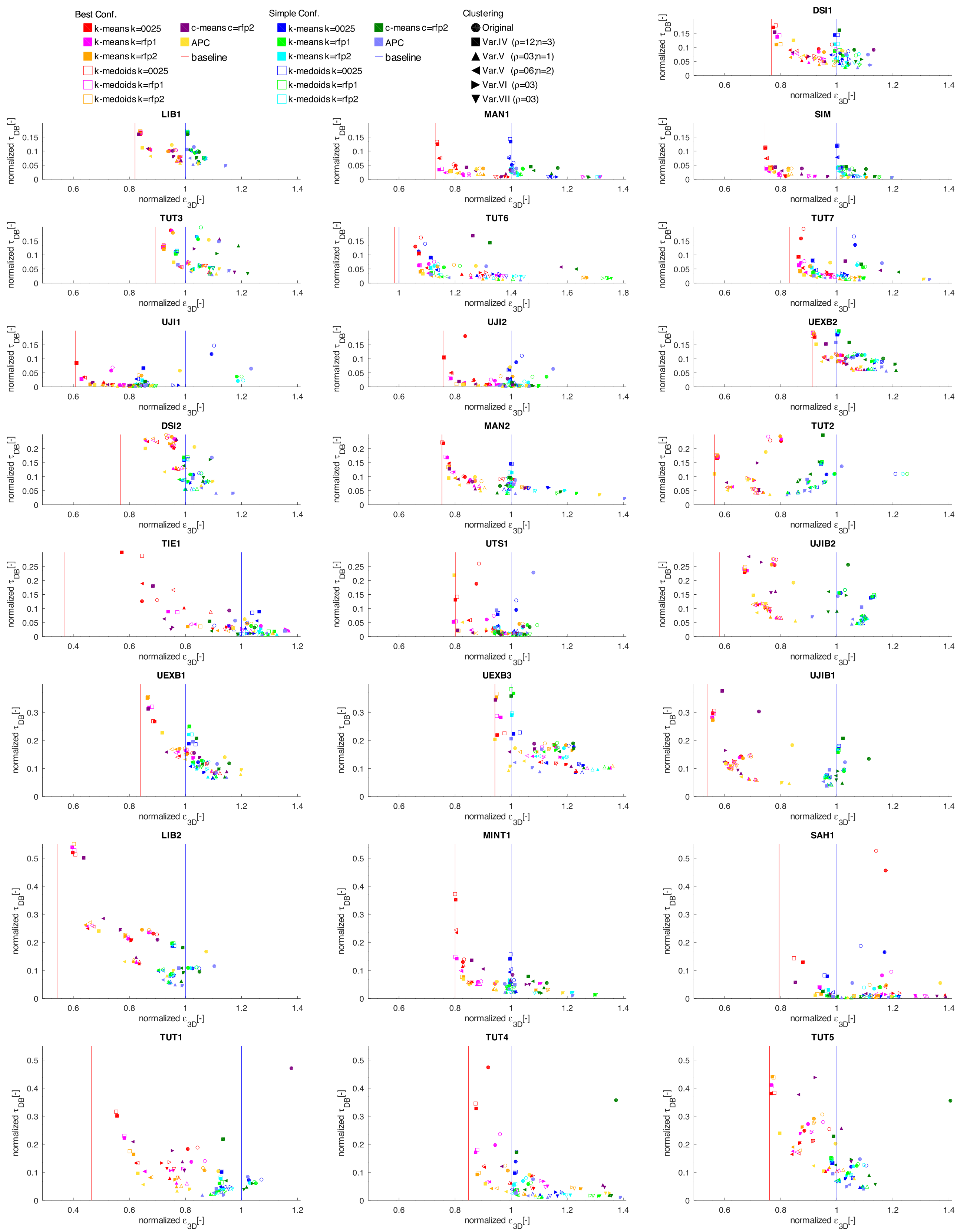


Fig. 7. Normalized results—$\epsilon_{\hat{3D}}$ and $\tau_{\hat{DB}}$—for the 25 data sets. The color indicates the clustering method (including parameters) whereas the shape indicates the clustering implementation (original method and the five selected variants). Results report a clear tradeoff between $\epsilon_{\hat{3D}}$ and $\tau_{\hat{DB}}$.

expected. DSI2 uses the Probabilistic Log-Gaussian Distance in the k-NNs model for the *best configuration*, so the clusters might probably not correctly mimic the related fingerprints as clustering models are usually based on Euclidean distance. We need to integrate advanced vector distances representing better the relation of two fingerprints.

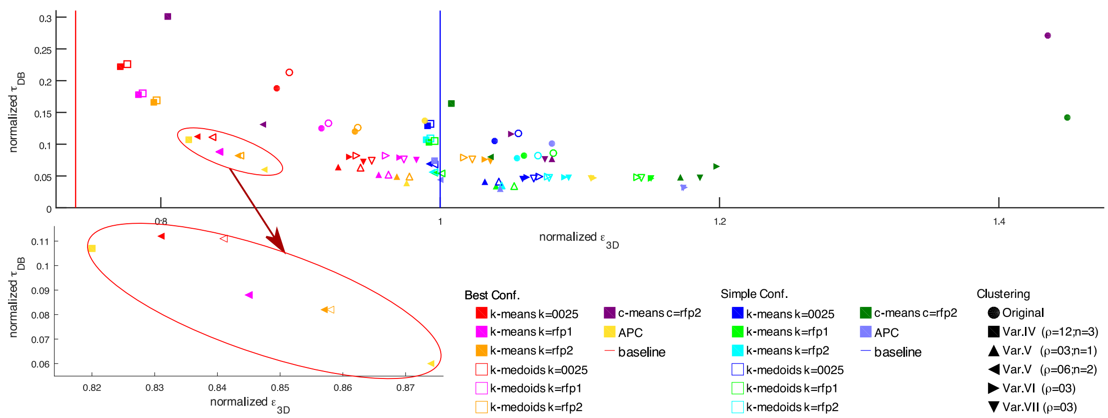


Fig. 8. General normalized results for the clustering models, including the original method and the proposed variants.

The third and last group involves data sets LIB2, MINT1, TIE1, TUT1, TUT4, TUT5, and UJIB2 all of them adopting a Log-Gaussian Distance in the *best configuration*. The computational cost of that distance metric is considerably higher (around ×3 times according to [15]) than traditional distance functions (e.g., Euclidean or Manhattan). This makes the *best configuration* computationally much more demanding in those data sets. Clustering has partially solved the problem at the expense of a worse positioning, probably because the clusters are generated by means of traditional distance metrics instead of the advanced ones. That is, the distance metrics used to compute the dissimilarity between the operational fingerprints and the centroids are not optimized for fingerprinting. Our variants, especially, those related to better positioning accuracy, almost reach the positioning accuracy of the baseline for the *best configuration*.

2) General Analysis: The general results, considering all data sets, for the proposed variants in the four selected clustering models are provided in Fig. 8, zooming the most promising ones in the bottom part of the figure.

First, k-Means and k-Medoids provide similar general results, k-Means being slightly better. Second, c-Means is providing good accuracy when combined with Variant IV ($\rho = 12, n = 3$) and Variant V ($\rho = 6, n = 2$) despite the bad results provided by the traditional version. However, the positioning error still remains slightly high because of data set TUT6. Third, APC has also been significantly improved with Variant IV ($\rho = 12, n = 3$) and Variant V ($\rho = 6, n = 2$), reaching very good results in both dimensions with the former.

Applying the rule of the elbow on the results reported in Fig. 8, k-Means with Variant V ($\rho = 6, n = 2$) and APC with Variant IV ($\rho = 12, n = 3$) are the best solutions for fingerprint-based positioning. However, both clustering methods have drawbacks as the k-Means requires to set k and APC is computationally demanding.

F. Discussion

The results have shown that not all the clustering models may be of relevance to fingerprint-based indoor positioning. The diversity on available data sets, the data representations available to linearize the RSS values, and the strategies to collect the data made some clustering models discouraging for the purpose of radio-map clustering. Despite the fact that k-Means, k-Medoids, and APC are providing a good trade-off between accuracy and efficiency, our previous work [28] showed that there may be still room to improve these three clustering methods by introducing domain-specific knowledge. In general, the novel variants proposed in this article have significantly improved the traditional models.

Regarding the results reported for each data set, the best model depends on the data set and application requirements. Nevertheless, we identified the following trends.

1) There is a tradeoff between execution time and positioning error. Metrics based on the Pareto efficiency may cope with multiple goals in IPS [63], [64].
2) For radio maps covering a small/medium size area with few fingerprints per reference point, k-Means with Variant V ($k = rfp1, \rho = 6, n = 2$) and APC with Variant IV ($\rho = 12, n = 3$) showed to be the most suitable models; the former providing better efficiency and the latter providing better accuracy among the two.
3) For radio maps covering large areas (including several floors in multistorey buildings) with few/several fingerprints per reference point, all the proposed variants significantly reduced the computational costs with respect to the baseline. In particular, the k-Means with Variant IV ($k = rfp1, \rho = 12, n = 3$) presents a very good tradeoff between efficiency and positioning accuracy.
4) For radio maps covering large areas with only one fingerprint per reference point, all the proposed variants significantly reduced the computational costs with respect to the baseline. In particular, APC with Variant IV ($\rho = 12, n = 3$) presents a very good tradeoff between efficiency and positioning accuracy in all data sets.
5) c-Means fails when the reference points have just one fingerprint and the distance between points is high.

To demonstrate the feasibility of the proposed clustering approaches with knowledge-based rules, we introduce Table IV. For each database, we have selected one of the previously suggested approaches based on the area size and density of fingerprints. For small/medium size environments, we apply k-Means and Variant V ($k = rfp1, \rho = 6, n = 2$).

TABLE IV

RESULTS OF THE PROPOSED VARIANTS DEPENDING THE DATABASE FEATURES. RESULTS INCLUDE THE MIN, AVERAGE, AND MAX POSITIONING ERROR OVER THE TEN RUN, AND THE MIN, AVERAGE, AND MAX EXECUTION TIME FOR ALL THE FINGERPRINTS EVALUATED OVER THE TEN RUNS

			Simple Conf.						Best Conf.					
DB	Clustering	Variant and parameters	$\overset{\blacktriangledown}{\epsilon}_{3D}$[m]	$\bar{\epsilon}_{3D}$[m]	$\overset{\blacktriangle}{\epsilon}_{3D}$[m]	$\overset{\blacktriangledown}{\tau}_{fp}$[ms]	$\bar{\tau}_{fp}$[ms]	$\overset{\blacktriangle}{\tau}_{fp}$[ms]	$\overset{\blacktriangledown}{\epsilon}_{3D}$[m]	$\bar{\epsilon}_{3D}$[m]	$\overset{\blacktriangle}{\epsilon}_{3D}$[m]	$\overset{\blacktriangledown}{\tau}_{fp}$[ms]	$\bar{\tau}_{fp}$[ms]	$\overset{\blacktriangle}{\tau}_{fp}$[ms]
DSI1	k-Means ($k = rfp1$)	Var.V ($\rho = 6, n = 2$)	4.74	4.91	5.12	0.30	2.13	6.33	4.01	4.16	4.28	0.30	2.66	7.29
DSI2	k-Means ($k = rfp1$)	Var.V ($\rho = 6, n = 2$)	4.72	4.87	5.02	0.19	1.31	2.89	4.03	4.23	4.37	0.25	3.40	8.42
LIB1	k-Means ($k = rfp1$)	Var.V ($\rho = 6, n = 2$)	3.06	3.08	3.09	0.25	1.50	3.01	2.61	2.64	2.67	0.26	1.55	3.02
LIB2	k-Means ($k = rfp1$)	Var.V ($\rho = 6, n = 2$)	3.71	3.78	3.83	0.26	1.47	3.10	2.59	2.71	2.89	0.38	3.81	8.57
MAN1	k-Means ($k = rfp1$)	Var.IV ($\rho = 12, n = 3$)	2.79	2.81	2.86	3.68	12.64	33.45	2.06	2.09	2.13	3.39	11.51	27.20
MAN2	k-Means ($k = rfp1$)	Var.V ($\rho = 6, n = 2$)	2.35	2.47	2.62	0.86	2.50	5.02	1.82	1.96	2.09	1.29	3.65	8.76
MINT1	k-Means ($k = rfp1$)	Var.V ($\rho = 6, n = 2$)	2.58	2.65	2.71	1.16	5.05	9.60	2.12	2.20	2.27	3.33	11.48	24.97
SAH1	Affinity Propagation	Var.IV ($\rho = 12, n = 3$)	8.94	8.94	8.94	1.18	1.92	4.14	8.37	8.37	8.37	2.04	3.01	5.76
SIM	k-Means ($k = rfp1$)	Var.IV ($\rho = 12, n = 3$)	3.16	3.24	3.30	5.15	10.17	17.79	2.36	2.42	2.47	4.95	9.39	15.55
TIE1	Affinity Propagation	Var.IV ($\rho = 12, n = 3$)	7.57	7.57	7.57	1.73	5.78	8.27	6.35	6.35	6.35	4.63	8.71	12.39
TUT1	Affinity Propagation	Var.IV ($\rho = 12, n = 3$)	8.72	8.72	8.72	0.42	1.59	3.16	6.04	6.04	6.04	0.86	3.70	7.61
TUT2	Affinity Propagation	Var.IV ($\rho = 12, n = 3$)	12.85	12.85	12.85	0.77	1.41	2.45	8.09	8.09	8.09	0.72	1.72	3.43
TUT3	Affinity Propagation	Var.IV ($\rho = 12, n = 3$)	9.33	9.33	9.33	0.26	1.30	2.69	9.18	9.18	9.18	0.27	1.48	2.98
TUT4	Affinity Propagation	Var.IV ($\rho = 12, n = 3$)	6.47	6.47	6.47	0.53	3.00	6.46	5.78	5.78	5.78	1.14	6.71	14.85
TUT5	Affinity Propagation	Var.IV ($\rho = 12, n = 3$)	6.82	6.82	6.82	0.43	1.15	1.58	5.51	5.51	5.51	0.98	2.92	4.05
TUT6	Affinity Propagation	Var.IV ($\rho = 12, n = 3$)	2.16	2.16	2.16	0.35	2.58	5.18	2.11	2.11	2.11	0.37	2.83	5.85
TUT7	Affinity Propagation	Var.IV ($\rho = 12, n = 3$)	2.47	2.47	2.47	0.30	2.08	3.96	2.38	2.38	2.38	0.32	2.31	4.52
UJI1	k-Means ($k = rfp1$)	Var.IV ($\rho = 12, n = 3$)	9.02	9.14	9.23	1.00	11.95	30.85	6.76	6.81	6.87	2.83	14.94	43.33
UJI2	k-Means ($k = rfp1$)	Var.IV ($\rho = 12, n = 3$)	7.92	8.00	8.11	0.28	11.54	26.85	6.23	6.32	6.45	0.29	16.87	50.24
UTS1	k-Means ($k = rfp1$)	Var.IV ($\rho = 12, n = 3$)	7.99	8.23	8.40	1.91	7.89	23.52	6.74	6.94	7.10	3.16	12.81	48.68
UEXB1	Affinity Propagation	Var.IV ($\rho = 12, n = 3$)	3.81	3.81	3.81	1.21	2.00	2.77	3.45	3.45	3.45	1.48	2.32	3.37
UEXB2	Affinity Propagation	Var.IV ($\rho = 12, n = 3$)	4.63	4.63	4.63	1.28	2.08	2.90	4.35	4.35	4.35	1.26	1.99	2.82
UEXB3	Affinity Propagation	Var.IV ($\rho = 12, n = 3$)	7.14	7.14	7.14	0.88	1.18	1.62	6.56	6.56	6.56	0.86	1.15	1.60
UJIB1	k-Means ($k = rfp1$)	Var.V ($\rho = 6, n = 2$)	2.89	2.95	3.00	0.36	1.18	2.33	1.76	1.87	1.93	0.42	1.78	3.30
UJIB2	k-Means ($k = rfp1$)	Var.V ($\rho = 6, n = 2$)	4.56	4.70	4.79	0.48	1.04	1.78	2.97	3.13	3.33	0.56	1.64	3.01

For large areas with one fingerprint per reference point, we apply Affinity Propagation and Variant IV ($\rho = 12, n = 3$). For large areas with multiple fingerprints per reference point, we apply k-Means and Variant IV ($k = rfp1, \rho = 12, n = 2$).

As the clusters may not be equally distributed, we provide the minimum ($\overset{\blacktriangledown}{\tau}_{fp}$), average ($\bar{\tau}_{fp}$), and maximum ($\overset{\blacktriangle}{\tau}_{fp}$) computational time to process one fingerprint for each data set in the ten runs. For example, estimating the position for UJI 2 may take 0.29 to 50.24 ms, being 16.87 ms on average. In most of the cases, positioning takes a few ms on average and the range of values is reasonable, whereas the maximum overall time, 50.24 ms, is a major achievement for the upper bound limit.

In terms of positioning, we provide the minimum ($\overset{\blacktriangledown}{\epsilon}_{3D}$), average ($\bar{\epsilon}_{3D}$), and maximum ($\overset{\blacktriangle}{\epsilon}_{3D}$) mean positioning error over the ten runs. The positioning accuracy is, in general, similar to the traditional k-NNs model without any optimization. One collateral effect of k-Means clustering is that the accuracy depends on the random centroid initialization, being the average accuracy in phase to the traditional k-NNs. Anyway, the variability between runs is not high.

V. Final Validation on Huge LoRaWAN Data Set

We have assessed the new variants on k-Means (Sections IV-B and IV-C) and the selected variants on selected clustering models (Section IV-E) using 25 data sets, reaching general discussion (Section IV-F). However, a question still remains open, *Would our results generalize to any fingerprint problem?*

In an attempt to answer that question, we validate our proposal on the huge LoRaWAN for outdoor positioning data set (Antwerp, Belgium) [65], [66]. It has 123 528 fingerprints, which were sorted according to the timestamp and then split into training and test sets (ratio $\approx$ 80 : 20) so that samples collected in the first days were used for training and the samples collected in the last days were used for evaluation. For the k-NNs model, we use the hyperparameters set in [67].

TABLE V

FINAL VALIDATION OVER THE LORAWAN FINGERPRINT DATA SET

	Method	ϵ_{3D}[m]	ϵ_{3D}^{median}[m]	ϵ_{3D}^{95prc}[m]	τ_{fp}[ms]
	plain k-NN	475.2	335.9	1410.8	2467.2
	Gallagher *et al.* [21]	475.2	335.9	1410.8	457.8
	Moreira *et al.* [48]	477.5	330.7	1441.8	204.4
k-means $k = rfp1$	original	487.1±1.2	340.2±1.0	1458.6±7.0	20.4±13.4
	Var.V ρ=0 n=1	486.2±1.8	338.5±2.4	1458.7±6.8	10.4± 8.7
	Var.V ρ=3 n=1	483.2±1.4	337.1±1.8	1448.5±9.3	12.4±10.2
	Var.V ρ=6 n=2	475.0±1.3	333.9±1.1	1419.4±7.3	23.2±18.0
	Var.IV ρ=12 n=3	473.9±0.9	334.0±0.6	1411.4±4.4	39.0±27.9

Table V introduces the results of k-NNs, optimization rules, and variants based on k-Means. APC was not explored as the required resources to generate the clusters were prohibitive. The median and 95th percentile errors, ϵ_{3D}^{median} and ϵ_{3D}^{95prc}, are also included as suggested in the ISO18305 standard.

As expected, the plain k-NNs is not computationally efficient. With our implementation run in Octave, processing one evaluation fingerprint takes, on average, 2.46 s to provide a position estimate. This is not practical neither for real-time positioning nor for hyperparameter selection.

The clustering model based on common APs [21] [here base stations (BS)] reduces the computational cost to $\approx$ 18.5%, as some LoRaWAN BS have been detected in a significant

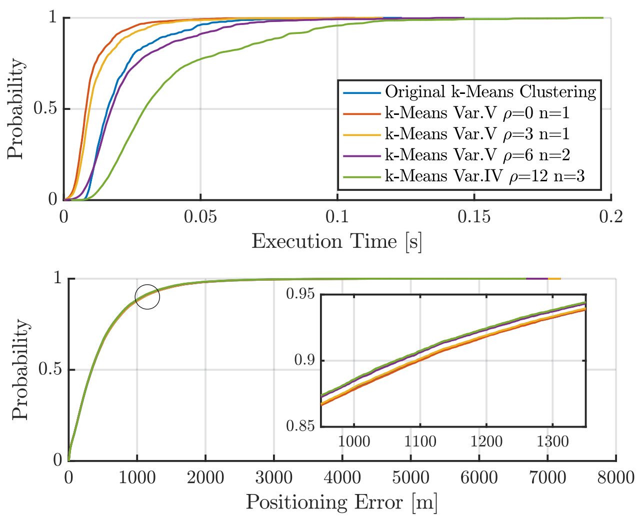


Fig. 9. CDF for execution time and positioning error for all the evaluation fingerprints using the original k-Means and the selected variants.

number of fingerprints. The model based on the strongest AP [48] only reduces the computational burden to, around, the tenth part in both data sets. Despite the original k-Means providing an excellent execution time, it also provides the worst positioning accuracy among all compared methods in terms of mean, median, and 95th percentile positioning errors.

The methods based on k-Means have been run ten times because of its random initialization, and therefore, the average and standard deviation over the ten runs are reported for the positioning errors (mean, median, and 95th percentile) and mean execution time for a fingerprint.

We have assessed the proposed variants providing the best computational time, the best tradeoff between the two evaluation metrics and the best positioning error (see Table III). The results are coherent to Section IV-E and the proposed alternatives provide better positioning accuracy than the original k-Means. Depending on the strategy to filter the centroids and clusters, the accuracy is similar to the plain k-NNs. The most permissive variant provides slightly better accuracy than plain k-NNs, while reducing its computational cost 63 times. The most conservative variant provides similar performance as traditional k-Means being ≈ 2 times faster (237 times faster than plain k-NNs).

The individual execution time and positioning errors for all the operational fingerprints in the ten runs are reported in Fig. 9 as a CDF plot (top). Although the LoRaWAN data set is challenging, the proposed variants can significantly reduce computational cost or almost keep it with similar positioning accuracy, showing that the proposed variants also work for outdoor positioning with significantly different technology.

VI. Conclusion

This article significantly extends the work in [28] by introducing four novel ways to modify clustering in fingerprinting and assessing them over seven clustering models and 26 open-source data sets. The data sets and code are available in [68], allowing the community to validate this work (reproducibility and replicability), and also to extend it by adding other data sets or clustering models.

First, we have performed an analysis of the new proposed clustering variants with k-Means. The number of variants and parameters is so high, that makes unfeasible a full comparison with all the clustering models. Based on the results with k-Means, we have selected the five most promising Variants (including parameters), all of them belonging to the ones that we propose in this article.

Second, our analysis showed that not all the clustering models are good for fingerprint-based indoor positioning. This is because most of RSS values in the radio maps usually correspond to undetected values (missing data) which may degrade the accuracy of clustering methods in generating the groups or detecting outliers. Among the seven considered clustering methods, only k-Means, k-Medoids, and affinity propagation clustering (APC) succeed in the context of fingerprint-based indoor positioning, by reducing the computational costs at the expense of slightly worse positioning accuracy compared to methods which do not rely on clustering. Additionally, c-Means is also working well for a few data sets if the number of clusters, k, is high ($k = rfp2$).

Third, a full analysis has been performed on the selected clustering methods (k-Means, k-Medoids, c-Means, and APC) and the selected five configurations with Variants IV–VII. The results showed that Variant V ($\rho = 6$, $n = 2$), the one with the best balance between positioning error and execution time in k-Means clustering, works well with all the considered clustering methods, providing good results for c-Means. For APC, Variant IV ($\rho = 12$, $n = 3$) also significantly improves the original model in terms of efficiency ad accuracy. However, cluster generation in APC is demanding and not scalable. All these findings were validated on a huge LoRaWAN data set.

Despite the differences between positioning technologies, our results show that the most relevant features to select a particular clustering model and a particular variant mainly depend on the geographical-area size and on the density of fingerprints collected in that area. Our analysis has shown that k-Means with Variant V ($k = rfp1$, $\rho = 6$, $n = 2$) is good and efficient for radio maps covering a small/medium-sized operational area. Variant IV ($\rho = 12$, $n = 3$) is the best variant for those radio-maps covering very large areas, where k-Means is probably better suited for those data sets with multiple fingerprints per reference point and APC is better suited for those data sets with only one fingerprint per reference point. However, any choices based on the rule of the elbow may be subjective as there is a negative correlation between the efficiency of an algorithm and its positioning accuracy. The final choice depends on the application. For example, wearables might favor an efficient variant, while applications for high-end devices might prefer a variant with low positioning error.

In summary, the best traditional clustering models have improved the efficiency of fingerprinting at the expense of a higher positioning error compared to fingerprinting without clustering. Our proposed modifications to clustering algorithms not only have improved their efficiency but also they have significantly improved their positioning accuracy, even in very large deployments. Thus, the proposed variants offer higher efficiency than the traditional methods without clustering as well as the same or better positioning accuracy.

Finally, most of the clustering models still rely on traditional distance metrics. We recommend revisiting clustering models to introduce signal propagation knowledge in the cluster-generation process. For real-time navigation, we will explore the concepts of neighbor relative RSS and trajectory analysis proposed in [10].

REFERENCES

[1] B. Wang, X. Liu, B. Yu, R. Jia, and X. Gan, "An improved WiFi positioning method based on fingerprint clustering and signal weighted euclidean distance," *Sensors*, vol. 19, no. 10, p. 2300, 2019. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6567165/

[2] Z. Li, X. Li, G. Mou, D. Jiang, X. Bao, and Y. Wang, "Design of localization system based on ultra-wideband and long range wireless," in *Proc. 11th IEEE Int. Conf. Adv. Infocomm Technol.*, 2019, pp. 142–146.

[3] N. Kuxdorf-Alkirata, O. Spathmann, O. Koch, and D. Bruckmann, "Improved energy efficiency of indoor positioning systems by adaptive sampling and smart post-processing of sensor data," in *Proc. 16th IEEE Int. New Circuits Syst. Conf. (NEWCAS)*, 2018, pp. 225–228.

[4] M. A. Afzal, D. He, Z. Zhu, and Y. Yang, "Performance evaluation of Wi-Fi Bluetooth low energy Li-Fi technology in indoor positioning," in *Proc. 23rd IEEE Int. Conf. Digit. Signal Process.*, 2018, pp. 1–5.

[5] D. Yan, B. Kang, H. Zhong, and R. Wang, "Research on positioning system based on ZigBee communication," in *Proc. 3rd IEEE Adv. Inf. Technol., Electron. Autom. Control Conf.*, 2018, pp. 1027–1030.

[6] V. Renaudin et al., "Evaluating indoor positioning systems in a shopping mall: The lessons learned from the IPIN 2018 competition," *IEEE Access*, vol. 7, pp. 148594–148628, 2019.

[7] B. Molina, E. Olivares, C. E. Palau, and M. Esteve, "A multimodal fingerprint-based indoor positioning system for airports," *IEEE Access*, vol. 6, pp. 10092–10106, 2018.

[8] L. Flueratoru, S. Wehrli, M. Magno, E. S. Lohan, and D. Niculescu, "High-accuracy ranging and localization with ultrawideband communications for energy-constrained devices," *IEEE Internet Things J.*, vol. 9, no. 10, pp. 7463–7480, May 2022.

[9] F. S. Daniş, A. T. Naskali, A. T. Cemgil, and C. Ersoy, "An indoor localization dataset and data collection framework with high precision position annotation," *Pervasive Mobile Comput.*, vol. 81, Apr. 2022, Art. no. 101554. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1574119222000098

[10] K. Lin, M. Chen, J. Deng, M. M. Hassan, and G. Fortino, "Enhanced fingerprinting and trajectory prediction for IoT localization in smart buildings," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 3, pp. 1294–1307, Jul. 2016.

[11] D. Sikeridis, B. P. Rimal, I. Papapanagiotou, and M. Devetsikiotis, "Unsupervised crowd-assisted learning enabling location-aware facilities," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4699–4713, Dec. 2018.

[12] B. Huang, Z. Xu, B. Jia, and G. Mao, "An online radio map update scheme for WiFi fingerprint-based localization," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6909–6918, Aug. 2019.

[13] H. Li, Z. Qian, C. Tian, and X. Wang, "TILoc: Improving the robustness and accuracy for fingerprint-based indoor localization," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 3053–3066, Apr. 2020.

[14] A. Ometov et al., "A survey on wearable technology: History, state-of-the-art and currentchallenges," *Comput. Netw.*, vol. 193, Jul. 2021, Art. no. 108074. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128621001651

[15] J. Torres-Sospedra et al., "A comprehensive and reproducible comparison of clustering and optimization rules in Wi-Fi fingerprinting," *IEEE Trans. Mobile Comput.*, vol. 21, no. 3, pp. 769–782, Mar. 2022.

[16] N. Swangmuang and P. V. Krishnamurthy, "On clustering RSS fingerprints for improving scalability of performance prediction of indoor positioning systems," in *Proc. MELT*, 2008, pp. 61–66. [Online]. Available: https://doi.org/10.1145/1410012.1410027

[17] G. G. Anagnostopoulos and A. Kalousis, "A reproducible comparison of RSSI fingerprinting localization methods using LoRaWAN," in *Proc. Workshop Positioning, Navigation Commun.*, 2019, pp. 1–6.

[18] T. Janssen, R. Berkvens, and M. Weyn, "Benchmarking RSS-based localization algorithms with LoRaWAN," *Internet Things*, vol. 11, Sep. 2020, Art. no. 100235. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2542660520300688

[19] P. Masek et al., "Measurements of LoRaWAN technology in urban scenarios: A data descriptor," *Data*, vol. 6, no. 6, p. 62, 2021. [Online]. Available: https://www.mdpi.com/2306-5729/6/6/62

[20] E. Svertoka et al., "Evaluation of real-life LoRaWAN localization: Accuracy dependencies analysis based on outdoor measurement datasets," in *Proc. 14th Int. Conf. Commun. (COMM)*, 2022, pp. 1–7.

[21] T. J. Gallagher, B. Li, A. G. Dempster, and C. Rizos, "A sector-based campus-wide indoor positioning system," in *Proc. Int. Conf. Indoor Positioning Indoor Navigation*, 2010, pp. 1–8.

[22] L. Klus et al., "Direct lightweight temporal compression for wearable sensor data," *IEEE Sensors Lett.*, vol. 5, no. 2, pp. 1–4, Feb. 2021.

[23] J. Ren, Y. Wang, C. Niu, W. Song, and S. Huang, "A novel clustering algorithm for Wi-Fi indoor positioning," *IEEE Access*, vol. 7, pp. 122428–122434, 2019.

[24] H. Shin and H. Cha, "Wi-Fi fingerprint-based topological map building for indoor user tracking," in *Proc. Int. Conf. Embedded Real-Time Comput. Syst. Appl.*, 2010, pp. 105–113.

[25] A. Razavi, M. Valkama, and E. Lohan, "K-means fingerprint clustering for low-complexity floor estimation in indoor mobile localization," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, 2015, pp. 1–7.

[26] Y. P. Raykov, A. Boukouvalas, F. Baig, and M. A. Little, "What to do when K-means clustering fails: A simple yet principled alternative algorithm," *PLoS One*, vol. 11, no. 9, pp. 1–28, Sep. 2016. [Online]. Available: https://doi.org/10.1371/journal.pone.0162259

[27] G. Ding, Z. Tan, J. Zhang, and L. Zhang, "Fingerprinting localization based on affinity propagation clustering and artificial neural networks," in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2013, pp. 2317–2322.

[28] J. Torres-Sospedra, D. Quezada-Gaibor, G. M. Mendoza-Silva, J. Nurmi, Y. Koucheryavy, and J. Huerta, "New cluster selection and fine-grained search for k-means clustering and Wi-Fi fingerprinting," in *Proc. Int. Conf. Localization GNSS*, 2020, pp. 1–6.

[29] A. Anuwatkun, J. Sangthong, and S. Sang-Ngern, "A DIFF-based indoor positioning system using fingerprinting technique and K-means clustering algorithm," in *Proc. 16th Int. Joint Conf. Comput. Sci. Softw. Eng.*, 2019, pp. 148–151.

[30] S. G. Lee and C. Lee, "Developing an improved fingerprint positioning radio map using the K-means clustering algorithm," in *Proc. Int. Conf. Inf. Netw.*, 2020, pp. 761–765.

[31] H. Zhou and N. Van, "Indoor fingerprint localization based on fuzzy C-means clustering," in *Proc. 6th Int. Conf. Meas. Technol. Mechatron. Autom.*, Jan. 2014, pp. 337–340.

[32] D. J. Suroso, P. Cherntanomwong, P. Sooraksa, and J. Takada, "Fingerprint-based technique for indoor localization in wireless sensor networks using fuzzy C-means clustering algorithm," in *Proc. Int. Symp. Intell. Signal Process. Commun. Syst.*, 2011, pp. 1–5.

[33] Y. Endo, Y. Hamasuna, Y. Kanzawa, and S. Miyamoto, "On fuzzy C-means clustering for uncertain data using quadratic regularization of penalty vectors," in *Proc. IEEE Int. Conf. Granular Comput.*, 2009, pp. 148–153.

[34] F. Li, M. Liu, Y. Zhang, and W. Shen, "A two-level WiFi fingerprint-based indoor localization method for dangerous area monitoring," *Sensors*, vol. 19, no. 19, p. 4243, 2019.

[35] P. A. Karegar, "Wireless fingerprinting indoor positioning using affinity propagation clustering methods," *Wireless Netw.*, vol. 24, no. 8, pp. 2825–2833, 2018. [Online]. Available: https://doi.org/10.1007/s11276-017-1507-0

[36] G. Caso, L. De Nardis, and M.-G. D. Benedetto, "A mixed approach to similarity metric selection in affinity propagation-based WiFi fingerprinting indoor positioning," *Sensors*, vol. 15, no. 11, pp. 27692–27720, 2015.

[37] H. Lin and L. Chen, "An optimized fingerprint positioning algorithm for underground garage environment," in *Proc. Int. Conf. Inf. Netw.*, 2016, pp. 291–296. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/ICOIN.2016.7427079

[38] J. Cheng, Y. Cai, Q. Zhang, J. Cheng, and C. Yan, "A new three-dimensional indoor positioning mechanism based on wireless LAN," *Math. Problems Eng.*, vol. 2014, Feb. 2014, Art. no. 862347.

[39] K. Wang et al., "Learning to improve WLAN indoor positioning accuracy based on DBSCAN-KRF algorithm from RSS fingerprint data," *IEEE Access*, vol. 7, pp. 72308–72315, 2019.

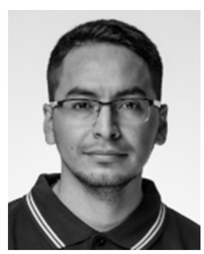
[40] M. Zhou, Y. Wei, Z. Tian, X. Yang, and L. Li, "Achieving cost-efficient indoor fingerprint localization on WLAN platform: A hypothetical test approach," *IEEE Access*, vol. 5, pp. 15865–15874, 2017.

[41] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," in *Proc. 18th Annu. ACM-SIAM Symp. Discrete Algorithms*, 2007, pp. 1027–1035.
[42] B. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, pp. 972–976, Mar. 2007.
[43] J. Dromard, G. Roudière, and P. Owezarski, "Online and scalable unsupervised network anomaly detection method," *IEEE Trans. Netw. Service Manag.*, vol. 14, no. 1, pp. 34–47, Mar. 2017.
[44] J. L. Andrews and P. D. Mcnicholas, "Model-based clustering, classification, and discriminant analysis via mixtures of multivariate t-distributions," *Stat. Comput.*, vol. 22, no. 5, pp. 1021–1029, 2012. [Online]. Available: https://doi.org/10.1007/s11222-011-9272-x
[45] W. Zhang and Y. Di, "Model-based clustering with measurement or estimation errors," *Genes*, vol. 11, no. 2, p. 185, 2020. [Online]. Available: https://www.mdpi.com/2073-4425/11/2/185
[46] J. Torres-Sospedra et al., "UJIIndoorLoc: A new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems," in *Proc. Int. Conf. Indoor Positioning Indoor Navigation*, 2014, pp. 261–270.
[47] J. Torres-Sospedra et al., "Towards ubiquitous indoor positioning: Comparing systems across heterogeneous," in *Proc. Int. Conf. Indoor Positioning Indoor Navigation*, 2021, pp. 1–8.
[48] A. Moreira, M. J. Nicolau, F. Meneses, and A. Costa, "Wi-Fi fingerprinting in the real world—RTLS@UM at the EvAAL competition," in *Proc. Int. Conf. Indoor Positioning Indoor Navigation*, 2015, pp. 1–10.
[49] A. Moreira, I. Silva, and J. Torres-Sospedra, "The DSI dataset for Wi-Fi fingerprinting using mobile devices." 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3778646
[50] G. M. Mendoza-Silva, P. Richter, J. Torres-Sospedra, E. S. Lohan, and J. Huerta, "Long-term WiFi fingerprinting dataset for research on robust indoor positioning," *Data*, vol. 3, no. 1, p. 3, 2018.
[51] T. King, S. Kopf, T. Haenselmann, C. Lubberger, and W. Effelsberg, "CRAWDAD dataset mannheim/compass (v. 2008-04-11)." 2008. [Online]. Available: https://crawdad.org/mannheim/compass/20080411
[52] T. King, T. Haenselmann, and W. Effelsberg, "On-demand fingerprint selection for 802.11-based positioning systems," in *Proc. Int. Symp. World Wireless, Mobile Multimedia Netw.*, 2008, pp. 1–8.
[53] A. Moreira, M. J. Nicolau, I. Silva, J. Torres-Sospedra, C. Pendão, and F. Meneses, "Wi-Fi fingerprinting dataset with multiple simultaneous interfaces." Sep. 2019. [Online]. Available: https://doi.org/10.5281/zenodo.3342526
[54] E. S. Lohan, J. Torres-Sospedra, and A. Gonzalez, "WiFi RSS measurements in Tampere university multi-building campus, 2017." Aug. 2021. [Online]. Available: https://doi.org/10.5281/zenodo.5174851
[55] A. Cramariuc, H. Huttunen, and E. S. Lohan, "Clustering benefits in mobile-centric WiFi positioning in multi-floor buildings," in *Proc. Int. Conf. Localization GNSS*, 2016, pp. 1–6.
[56] E.-S. Lohan, J. Torres-Sospedra, H. Leppäkoski, P. Richter, Z. Peng, and J. Huerta, "Wi-Fi crowdsourced fingerprinting dataset for indoor positioning," *Data*, vol. 2, no. 4, p. 32, 2017.
[57] P. Richter, E. S. Lohan, and J. Talvitie, "WLAN (WiFi) RSS database for fingerprinting positioning." 2018. [Online]. Available: https://zenodo.org/record/1161525
[58] Lohan, "Additional TAU datasets for Wi-Fi fingerprinting-based positioning." May 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3819917
[59] X. Song et al., "A novel convolutional neural network based indoor localization framework with WiFi fingerprinting," *IEEE Access*, vol. 7, pp. 110698–110709, 2019.
[60] F. J. Aranda, F. Parralejo, F. J. Álvarez, and J. Torres-Sospedra, "Multislot BLE raw database for accurate positioning in mixed indoor/outdoor environments," *Data*, vol. 5, no. 3, p. 67, 2020. [Online]. Available: https://www.mdpi.com/2306-5729/5/3/67
[61] G. M. Mendoza-Silva, M. Matey-Sanz, J. Torres-Sospedra, and J. Huerta, "BLE RSS measurements dataset for research on accurate indoor positioning," *Data*, vol. 4, no. 1, p. 12, 2019. [Online]. Available: https://www.mdpi.com/2306-5729/4/1/12
[62] N. Saccomanno, A. Brunello, and A. Montanari, "What you sense is not where you are: On the relationships between fingerprints and spatial knowledge in indoor positioning," *IEEE Sensors J.*, vol. 22, no. 6, pp. 4951–4961, Mar. 2022.
[63] F. Domingo-Perez, J. L. Lazaro-Galilea, I. Bravo, A. Gardel, and D. Rodriguez, "Optimization of the coverage and accuracy of an indoor positioning system with a variable number of sensors," *Sensors*, vol. 16, no. 6, p. 934, 2016. [Online]. Available: https://www.mdpi.com/1424-8220/16/6/934
[64] G. G. Anagnostopoulos, M. Deriaz, and D. Konstantas, "A multiobjective optimization methodology of tuning indoor positioning systems," in *Proc. Int. Conf. Indoor Positioning Indoor Navigation (IPIN)*, 2017, pp. 1–8.
[65] M. Aernouts, R. Berkvens, K. Van Vlaenderen, and M. Weyn, "Sigfox and LoRaWAN Datasets for fingerprint Localization in large urban and rural areas." Jul. 2019. [Online]. Available: https://doi.org/10.5281/zenodo.3904158
[66] M. Aernouts, R. Berkvens, K. Van Vlaenderen, and M. Weyn, "Sigfox and LoRaWAN Datasets for fingerprint Localization in large urban and rural areas," *Data*, vol. 3, no. 2, p. 13, 2018. [Online]. Available: https://www.mdpi.com/2306-5729/3/2/13
[67] F. Lemic et al., "Regression-based estimation of individual errors in fingerprinting localization," *IEEE Access*, vol. 7, pp. 33652–33664, 2019.
[68] J. Torres-Sospedra, "Supporting materials for 'scalable and efficient clustering for indoor positioning based on fingerprinting.'" 2021. [Online]. Available: https://doi.org/10.5281/zenodo.5167932

Joaquín Torres-Sospedra received the Ph.D. degree from Universitat Jaume I, Castellón de la Plana, Spain, in 2011.

He is currently an Assistant Researcher with the University of Minho, Guimarães, Portugal, where he works on indoor positioning and machine learning for industrial applications. He has authored more than 170 articles in journals and conferences; and supervised 16 masters and six Ph.D. students.

Dr. Torres-Sospedra is the Chair of the IPIN International Standards Committee and IPIN Smartphone-based off-site Competition.



Darwin P. Quezada Gaibor received the bachelor's degree in mechatronic engineering from Universidad Tecnológica América, Quito, Ecuador, in 2013, and the master's degree in radioengineering—GNSS receivers: hardware and software from Samara National Research University, Samara, Russia, in 2017. He is currently pursuing the Ph.D. degree with Universitat Jaume I, Castellón de la Plana, Spain, and Tampere University, Tampere, Finland.

His main interests are VoIP, cloud computing, networking, servers, and open-source software.

Jari Nurmi (Senior Member, IEEE) received the D.Sc.(Tech.) degree from Tampere University of Technology (TUT), Tampere, Finland, in 1994.

He has been a Professor with Tampere University (formerly, TUT) since 1999. He held various positions with TUT from 1987 to 1994, and was the Vice President of SME VLSI Solution Oy, Helsinki, Finland, from 1995 to 1998. Since 2013, he has been a Partner at research spin-offs. He is the Director of National DELTA Doctoral Training Network of over 200 Ph.D. students, the Coordinator of H2020 ETN APROPOS, and the Head of H2020 EJD A-WEAR at TAU. He has supervised 27 Ph.D. and about 150 M.Sc. theses, and been a opponent/reviewer of over 40 Ph.D. theses worldwide. He has edited five Springer books, and published over 350 international publications. He works on embedded computing, wireless localization, and software-defined radio/networks.

Dr. Nurmi is an associate editor/handling editor of three international journals. He is a member of the Technical Committee on VLSI Systems and Applications at IEEE CASS, and in steering committees of four international conferences (chairman in two).

Yevgeni Koucheryavy received the Ph.D. degree from Tampere University of Technology, Tampere, Finland, in 2004.

He is currently a Full Professor with the Unit of Electrical Engineering, Tampere University, Tampere. He has authored numerous publications in the field of advanced wired and wireless networking and communications. His current research interests include various aspects of heterogeneous wireless communication networks and systems, and emerging communication technologies for digitally augmented future beings.

Elena Simona Lohan (Senior Member, IEEE) received the M.Sc. degree in electrical engineering from the Polytechnics University of Bucharest, Bucharest, Romania, in 1997, the D.E.A. degree (French equivalent of master) in econometrics from École Polytechnique, Paris, France, in 1998, and the Ph.D. degree in telecommunications from Tampere University of Technology, Tampere, Finland, in 2003.

She is a Professor with Tampere University, Tampere, where she is currently a Professor with Electrical Engineering Unit and the Coordinator of the MSCA EU A-WEAR network. Her current research interests include wireless location techniques, wearable computing, and privacy-aware positioning solutions.

Joaquín Huerta received the Ph.D. degree from Universitat Jaume I, Castellón de la Plana, Spain, in 1999.

He is currently a Full Professor with the Department of Information Systems, University Jaume I, where he teaches several courses related to GIS and Internet Technologies. He is the Head of the GEOTEC Research Group, and the Director of the Erasmus Mundus Master of Science in Geospatial Technologies, run jointly with the universities of Münster and Nova de Lisboa, Lisboa, Portugal;. He is and has been a PI of several research projects, including EU projects, such as A-WEAR, GEO-C, and EUROGEOSS. He is a Founding Member of UBIK Geospatial Solutions. His current research interests are indoor positioning, smart cities, GIS applications, and augmented reality.