



Contents lists available at ScienceDirect

Journal of King Saud University –  
Computer and Information Sciencesjournal homepage: [www.sciencedirect.com](http://www.sciencedirect.com)Analysis of wireless network access logs for a hierarchical  
characterization of user mobility

Francisco Talavera, Isaac Lera, Carlos Guerrero \*

Crta. Valldemossa km 7.5, Palma E07121, Spain

## ARTICLE INFO

## Article history:

Received 8 November 2021

Revised 8 March 2022

Accepted 11 March 2022

Available online 29 March 2022

## Keywords:

User mobility

Mobility modeling

User behavior simulation

Fog computing

## ABSTRACT

This paper presents a method that generates a hierarchical user mobility model from the analysis of the data available from Wi-Fi connections. The data obtained from the Wi-Fi infrastructure is defined in terms of the coverage areas of the access points that the users move through. These access points are recursively grouped into different levels of granularity based on their geospatial features. The track of a user is defined as a sequence of Wi-Fi access points, which is enough to simulate user mobility in, for example, fog scenarios. The hierarchical definition of the region under study is proposed to reduce the complexity of the model in high-scale scenarios and to increase the adaptability between scenarios with different geospatial features. The model creation is based on a user profiling method that uses a clustering algorithm and each user type is defined with a transition matrix between coverage areas and a time length vector for the areas. The method is applied to the case of the campus of the University of the Balearic Islands. From the analysis of the mean square error of the results, we determined that the proposed method obtains good results for the transition matrices, but that the time vector definition should be improved. The results also show lower complexity in the case of the hierarchical model, with one area for each building and three levels, in regard to a non-hierarchical model, with only one area and one level for the whole campus.

© 2022 The Author(s). Published by Elsevier B.V. on behalf of King Saud University. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

## 1. Introduction

The number of research studies in the field of fog computing has increased significantly during the last years. In most of the cases, the experimental phase and the testing of the new research proposals are performed in simulation environments, mainly because of the difficulties to access real infrastructures with real users. Simulated results are only reliable if the simulation uses a model based on a real scenario. Thus, simulations typically include synthetic traces created from a model defined from real data (Eeckhout et al., 2000). But a quick review of the up-to-date literature in the field of fog computing (Ogundoyin and Kamil, 2021; Brogi et al., 2020) shows that most of the experiments are performed with random non-realistic models, because of the difficulties to access real data from these environments to create the models.

Moreover, fog environments are strongly influenced by the mobility of the users in the system, contrary to other traditional distributed architectures. Consequently, user mobility models,

based on real-data, are also necessary to obtain reliable results in the simulated experiments. User mobility influences fog architecture concerning the access points (APs) to which the users are connected to. In other words, the connection of a user to a given AP determines, for example, the origin of the requests, the number of requests sent from each AP, the network route the requests and responses are transmitted along, the network load, etc. Thus, in a simulation, the fog users' mobility model determines the APs where the users are connected to. This model is not influenced by the underlying network technology, and these connections could be established with, for example, a Wi-Fi AP, a 5G radio access network, etc. without altering the mobility patterns of the users.

The importance of user information for technological companies compromises its publication in open data catalogues, and user mobility is one such example. There are many studies in the literature that require of user mobility and that illustrate the importance of having user mobility models (Toch et al., 2019). There are examples of the key aspect of this user mobility modeling in technological research areas –such as mobile networks (Zonoozi and Dassanayake, 1997; Zhang and Dai, 2019), location-based applications (Noulas et al., 2012), intelligent transport systems

\* Corresponding author.

E-mail addresses: [f.talavera@uib.es](mailto:f.talavera@uib.es) (F. Talavera), [isaac.lera@uib.es](mailto:isaac.lera@uib.es) (I. Lera), [carlos.guerrero@uib.es](mailto:carlos.guerrero@uib.es) (C. Guerrero).

(Quessada et al., 2020), cloud performance (Secci et al., 2016), or fog infrastructures (Bittencourt et al., 2017)– and in social areas –such as crowds management (Huang et al., 2018), travel patterns discovering (Hoogendoorn and Bovy, 2005), natural areas management (Meijles et al., 2014), or analysis of sport activities (Lera et al., 2017)–.

Modeling methodologies to characterize user mobility still have room for new proposals. One example is the case of our domain problem, in which the mobility of the user is modeled hierarchically and created from a small set of mobility data, only requiring the data related to the APs in which the users are connected to.

In this work, we present a method to create a hierarchical user mobility model that recursively defines geospatial levels by grouping the neighboring APs into regions and zones of different granularity. The input data of the method is a data set of user connections to the APs of a Wi-Fi infrastructure. We focus our study on the use of Wi-Fi technology because this type of data is more accessible than 5G data, which requires the collaboration of public communication companies. But our method could be easily applied in other types of connection networks.

Note that we do not propose a new system or tool for user location or for gathering this data. We use such a type of systems to gather user location data, which are subsequently used by our proposed method for the creation of the user mobility models. In our particular case study, we decided to use Aruba Location Engine (ALE) because it was already deployed in our infrastructure, but any other system, such as Cisco Prime Infrastructure would be also suitable to gather the data.

The paper is organized as follows: in the remainder of this introductory section, the motivations and contributions of the proposed work are highlighted; Section 2 reviews related research works; Section 3 introduces the context of user mobility modeling; Section 4 includes the details of the proposed method for user mobility modeling; Section 5 presents the application of the proposed method in the study of the user mobility in the campus of the UIB, and the results are analyzed; and finally, Section 6 summarizes the conclusions and establishes future research lines.

### 1.1. Motivation and contributions

The evaluation of new fog proposals is usually performed in a simulator in their early experimental phases, because of the large scale of these infrastructures. Simulations need real data to obtain reliable results. But the number of open data sets related to user mobility in IT infrastructures is very limited (Luca et al., 2021), and most of them do not include the relationship with the computing infrastructures (Mimouna et al., 2020).

Obtaining data from real scenarios is complex and many researchers do not have access to suitable data sets, nor to infrastructures where they can collect the data.

From the analysis of the experiments of most of the research works (Ogundoyin and Kamil, 2021; Brogi et al., 2020), it is observed that there is an important challenge related to the use of real models in simulations. When researchers of fog infrastructures need to test and validate their proposals, they need to face two important challenges, depending on the resources they have.

The first case is when the researchers have access to the deployed infrastructures they want to simulate, and they can collect users' mobility data and their interactions with the computing infrastructure. In those cases, the main challenge is that the computational complexity to create the user mobility model is very high.

The second case is when the researchers can only access open data sets or models, whose availability is very limited, and the existing models need to be adapted to the infrastructure and the geospatial features of their study case.

Consequently, two research questions arise from these research challenges:

- RQ1: Is it possible to model user mobility in such a way that allows us to extrapolate/adapt a model between case studies with different geospatial features?
- RQ2: Is it possible to reduce the complexity of the mobility creation for large scale scenarios, such as fog infrastructures?

We have analysed several up-to-date reviews and surveys in the field of user mobility modeling, as we comment in Section 2, and to the best of our knowledge, current user modeling proposals lack both challenges, the computational complexity of the mobility model creation for large scale domains, and the adaptation of the mobility models between scenarios with different geospatial features.

Our proposal addresses both challenges by combining approaches from the fields of user characterization and geospatial studies. We combine the use of transition matrices to model user mobility behavior (Barbosa-Filho et al., 2018) and the definition of a hierarchical geospatial organization (Xie et al., 2014; Xu et al., 2015) of the networking resources in which the users are connected to. We concrete our proposal in the following research hypothesis: User mobility can be modeled by defining hierarchical levels of the geospatial zones of the region under study, and modeling mobility at each of these levels with transition matrices between the zones. This hierarchical approach reduces complexity and increases the extrapolation of the resulting model.

The great volume of data regarding user connections requires a data analysis solution that reduces the problem size without compromising results. With the use of a geospatial hierarchical decomposition of the region under study into levels and zones, the modeling process can be split into smaller sets, reducing the computational complexity.

Additionally, the hierarchical definition of the mobility model allows us to easily adapt it to scenarios with different geospatial features by scaling it up/down. For example, zones with given features can be duplicated or removed to adapt the mobility model to other case studies.

We particularize our proposal in the case of Wi-Fi infrastructures and apply it in a real study case. Thus, the contributions of this paper are:

- A method for hierarchical user mobility modeling. The model is defined with pairs of a stochastic transition matrix and a time vector, that respectively model the changes in the coverage areas of the APs (i.e., the AP that the user is connected to) and the time that the user stays in that coverage area. The input of the model is the connection data obtained from Wi-Fi APs. The proposed method can be easily extended to other connection technologies, for Example 5G networks.
- The application of the proposed method to a real scenario (the campus of the University of the Balearic Islands, UIB, a medium size university) to validate and test the proposed method. The obtained model has been also published in an open repository to allow other researchers to use/adapt/extend the model to their specific necessities.

We have highlighted the use of mobility data in the evaluation of fog infrastructures and we have focused on the applicability of our method to mobility generation in this type of environment. But our mobility models can be used in any type of evaluation/experiment that requires simulating the user mobility in a wireless environment. The only constraint is that the user mobility is modeled and expressed in terms of the APs where the users are connected to, i.e., the coverage area of the network access devices.

## 2. Related work

The study of human mobility data plays a nuclear function in understanding scientific areas related, as in our case, the evaluation of fog infrastructures. But it is not less important its role in fields such as culture, spread of epidemics, environmental impact, tourism industry. There is an extended diversity of studies on the analysis of human mobility in so many scientific areas, reflecting its importance. Many surveys collect and classify these researches, generally based on the data set type and the purpose of the study (Barbosa-Filho et al., 2018; Solmaz and Turgut, 2019; Thornton et al., 2018; Toch et al., 2019; Wang et al., 2019; Luca et al., 2021; Pappalardo et al., 2019; Kulkarni et al., 2019; King et al., 2021; Hess et al., 2016).

Becker et al. (2013) demonstrated the value of cellular network data for human mobility modeling. But their analysis of the mobility was focused on geographic aggregation of the results, instead of a single user-aware modeling. Consequently, traces of user movement can not be obtained from their mobility characterization.

Azevedo et al. (2009) analyzed mobility real traces obtained with GPS technology. They proposed to base the mobility characterization in the study of the probability and the cumulative distribution functions of the velocity, acceleration, direction angle change and pause time of the user traces. They identified that velocity and acceleration components follow a Normal distribution and that the direction angle change component and the pause time are better represented by a Log-normal. This results are very valuable to generate synthetic traces of users in general scenarios. But for our target, emulating the user connections to APs, a more detailed model is required.

Wu et al. (2017) dealt with the complexity of the model creation by proposing a distributed method for the mobility modeling. They propose a distributed version of the centralized algorithm NN-K-SVD. But the resulting model is not suitable for the emulation of user connections to APs.

Thuillier et al. (2018) characterized user mobility by first profiling the users with a the k-mean clustering algorithm. Although our proposal also creates user clusters, in our case the clustering group users with similar mobility. On the contrary, Thuillier et al. based the clustering in the features of call detail records.

In the last decades, data from GSM network records have allowed the characterization of mobility patterns (Barbosa-Filho et al., 2018) with a greater volume of samples than other technologies i.e. GPS. Following this ubiquitous technological evolution, phones' Wi-Fi adapters describe in-detail indoor spatio-temporal resolutions, which were difficult to obtain with previous technologies, and also provide a large volume of samples. Any Wi-Fi device sends beacon-like messages to Wi-Fi APs which are gathered in the AP for logging purposes.

Traunmueller et al. (2018) presented a study focused on improving urban management and planning decisions using Wi-Fi probes. They used 54 APs in Lower Manhattan over one week getting 30 million observations from 800.000 unique devices. After data anonymization and cleaning processing, they conducted a graph analysis modeling the AP locations as the nodes, and the user movement between consecutive APs as the edges. They generated a street usage intensity network model and paths of travel at a time. They used the model to interpret pedestrian routes and frequencies of points of interest (ferries, buildings, street connectors, etc.). Our study differs in that it is mainly focused on the method to obtain the mobility model, instead of the obtained mobility model. In any case, our case study, in a comparative scale, uses 425 APs located on floors and rooms in 18 buildings, resulting in 633000 filtered trajectories recorded in one week. There are other more sim-

plistic studies which do not consider the user paths, and they are also focused on the Wi-Fi data set instead of the modeling method.

Uras et al. (2019) conducted another human mobility study to identify human density, flows, patterns and heat maps through Wi-Fi probe data. The experiment setup was on three locations: a street in Turing, the Alba center, and the University of Cagliari where they deployed a small number of APs in comparison with other studies —1, 5 and 8 devices respectively—. They simplified the statistical methods by setting seven time periods and counting the permanence time of unique devices in the AP radius. This study shows the flexibility of these logs to obtain mobility indicators and ad hoc models. But the method they used to create the model is very limited for large scale scenarios.

Another important issue with these data sets is the estimation of the number of mobile devices present at a certain place and time and the number of people. Oliveira et al. (2019) focused on this problem using several threshold values with Pearson coefficient correlations. The article of Balzotti et al. (2018) uses mobile phone data provided by an Italian telecommunication company to adjust the length of the movement flow. Hoteit et al. (2017) proposed a technique to reduce the error between real and estimated human trajectories and to identify the period of time where users' locations remain undefined. We addressed this problem using an experiment threshold since our context offers less noise in this undefined user state.

Gao et al. (2010) based mobility modeling in the use of Hidden Markov Models obtained from data of Wi-Fi infrastructures. But their goal was to obtain a fine-grained model (a higher precise user location), instead of a course-grained model (determined by the AP location or the coverage area). Because we are interested in the generation of user emulation in fog environments, course-grained models are more suitable.

Additionally to the work of Gao et al., Markov modeling is studied in an important number of researches for modeling user mobility, most of them using GPS traces. Examples of variants for the Markov modeling are: simple Markov model using GPS traces (Ashbrook et al., 2002); semi-Markov model using GPS, GSM and Wi-Fi datasets (Chon et al., 2012); hidden Markov model using GPS traces (Mathew et al., 2012); mixed Markov model using GPS traces (Asahara et al., 2011); mobility Markov chain using GPS traces (Gambs et al., 2012), extended mobility Markov chain using call detail records (Amirrudin et al., 2013), variable-order Markov model using GPS traces (Yan et al., 2013), hidden semi-Markov model (Yu and Kobayashi, 2003), or spherical hidden-Markov model using geotagged Twitter datasets (Zhu et al., 2018).

Our proposal of modeling user mobility with transition matrices is supported by all these related references. Although they cover a wide range of alternatives, to the best of our knowledge, we are the first work that considers a hierarchical modeling based on transition matrices by fixing the user position in terms of the APs where they are connected to.

Finally, we cite some related studies that implemented similar spatio-temporal aggregations with the objective of focusing the study on a specific region without losing information on the context of the subsumed regions.

Xie et al. (2014) implemented a histogram tree to identify the region through which a vehicle circulates. Xu et al. (2015) apply a hierarchical decomposition of the phone location data set from Shenzhen, China, in mobile phone towers. Both studies obtain a better performance analysis using this hierarchical decomposition but in our case, the decomposition is used to describe the flow model not only to facilitate the operative. It is a vision of the movement with different granularity levels.

### 3. Problem statement

In general terms, users' movements are represented as a temporal sequence of geographical positions. Often the trajectories are recorded by GPS devices that offer fine-grained accuracy (Lera et al., 2017). However, other technologies, and their infrastructures, make the capture of the passage of users over regions possible, by associating personal devices and bind points. Phones and cellular radio towers are an example or, as in our case, Wi-Fi devices and APs.

It is observed that mobility modeling usually has a high computational complexity (Fülöp et al., 2009), and the generated models are particular for specific scenarios and they lack in adaptability to other scenarios with different geospatial features (Xie et al., 2013). We propose a geospatial hierarchy user mobility modeling to address the problem of the computational complexity of large scale mobility scenarios and the low capacity of adaptation of the obtained models. Our model defines the interactions between users and network access devices, and the evolution of these interactions, i.e., the model represents the sequence of APs the users are connected to during their sessions/tracks.

Fig. 1.a shows an example of a user<sup>1</sup> trajectory in the university campus of the UIB. The user follows the red line path over the background campus image, that also includes the coverage areas –represented with hexagons– and the AP of each area –represented with a numbered blue dot in the center of the hexagon–. This trajectory also follows a time distribution –the orange dots– that synthesizes the stay length in that space region.

From the fog infrastructure perspective, the trajectory depends on the communication probes between the user devices and Wi-Fi APs. These probes depend on the coverage mesh and the messages that both entities exchange. In any case, the traffic link, between the device with the AP in a point in time, enables the characterization of the user movement. Thus, we can create an access log that includes 3-tuples lines relating to time, device/user and AP. We name this type of data set as wireless session access log. The session of a given user corresponds to the sub-set of samples with the same device identifier. Generally, the wireless session access log interleaves samples of different user sessions. Fig. 1.b shows the wireless session access log from the example of Fig. 1.

The key samples of a user session can be even reduced to the first and the last ones (that indicates the starting and ending points) and the first one of each new AP connection (blue colored lines in Fig. 1.b). The other samples are only related to the accuracy of the samples and they do not provide any additional information. With this subset of samples, the handoffs<sup>2</sup> of the user connections are identified. We name the handoff trace log to this data subset that only includes the filtered wireless sessions. Fig. 1.c shows the hand-off trace log including a single filtered wireless connection session, of the example in Fig. 1.

Our proposal represents the user mobility model, as in other previous studies, with a stochastic transition matrix (the Origin–Destination matrix Barbosa-Filho et al., 2018), that represents the probability of AP handoffs. We extend that model with the inclusion of a time vector that represents the average length time of the connection of a user to a given AP. Fig. 1.d shows the transition matrix and the time length vector of the example.

We also extend the user modeling by defining a geospatial hierarchy of the network devices where the users are connected to. The hierarchical modeling is performed by grouping the neighboring APs in different levels of granularity, by taking into account the

geospatial organization of the region under study. When this geospatial organization is taken into account, the adaptation of the resulting model to regions with different features will be easier by duplicating zones with similar features or by removing the ones without similarities.

For example, in a first level, the groups could be created with the APs in the same building. In a second level, a set of APs in a building could be split into the parts of the building (for example, floors or wings), and this would be repeated recursively until the granularity reaches only one AP per group. For example, Fig. 2 shows a specific case of three levels of granularity.

This recursive geospatial division of the region is directly reflected in the mobility model because the model is also split into independent transition matrices for each geospatial zone and each level. This is also shown in Fig. 2, where the first level is represented with a matrix of the size of the number of buildings, that models the movement between buildings in the region under study. In the second level, a new matrix for each of the buildings is created to model the movement inside each of them. The matrix size corresponds to the number of parts a building is divided into.

Once the researchers have a mobility model, they need to adapt it to a specific problem definition to generate specific synthetic traces from a general mobility model. A simple and general adaptation refers to changing the percentages of each user type, the number of user types, or the user speed, between others. But if geospatial features need to be adapted, non-hierarchical models can not be adapted easily. In those cases, the use of hierarchical geospatial models simplified the adaptation process.

Fig. 3 shows an example in the context of the study case in Section 5, a university campus. Imagine that a group of researchers wants to test the performance of a new fog infrastructure simulating the movement of the users in their campus (university A). These researchers do not have access to real data from the students' movement and they decide to adapt a mobility model from another university that is available in an open repository (university B).

The university B model is defined with our hierarchical proposal and it models a campus with one classroom building (with two zones inside the building), one teachers' offices building (with two zones inside the building), and one student housing (with three zones inside the building). The problem is that university A has different geospatial features, with two classrooms buildings (one of them with similar features to University B and the other one with the double size), one teachers' building (with similar features to university B) and no student housing. Consequently, the mobility model of university B is not directly applicable to university A.

Because of the hierarchical definition of the mobility model, the researchers can adapt the model of university B by removing the student housing, duplicating the classroom building in the building level of the hierarchy, and by increasing the size of this second duplicated classroom building by doubling the zones in the building-parts level. These adaptations are graphically represented in Fig. 3. Additionally, the figure also shows how the transition matrices are modified, by also duplicating or removing matrices and columns/rows of those matrices.

### 4. Proposed solution

This section presents the details of our proposed method that covers the requirements to address the research problem stated in the previous section (Section 3). Our method involves successive phases of data collection, the definition of the model, and generation of synthetic data (Fig. 4). By this, researchers are able to create synthetic traces with the mobility model obtained from these

<sup>1</sup> We use the terms user and device indistinctly along the paper. They refer to the concept of *thing*, in the context of the Internet of Things, an entity that requests a service or generates and transfers data.

<sup>2</sup> A handoff happens when a device changes the AP where it is connected to.

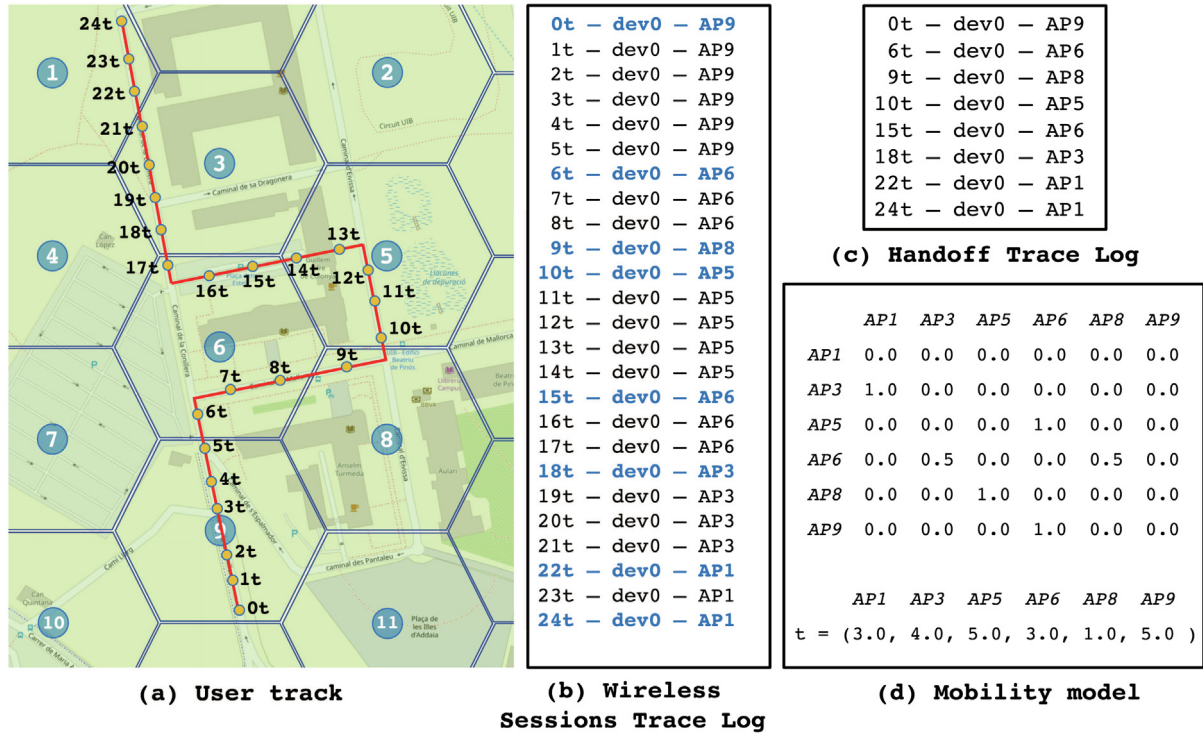


Fig. 1. Example of a user track and the corresponding mobility model.

phases. Synthetic traces of user mobility can be exported to simulation/emulation engines for a more realistic evaluation of, for example, fog environments.

The phases in the method we propose are based on the traditional problem of characterizing e-business workloads with Customer Behavior Model Graphs (CBMG) (Menasce and Virgilio, 2000). But we have extended this method to incorporate a geospatial hierarchy of the APs that: (a) facilitates the adaptation of the models between scenarios with different geospatial features; (b)

reduces the computational complexity of the generation of the model.

In summary, workload characterization with CBMG uses the web user requests stored in the web access logs to create users characterizations based on their web browsing. To do that, the user sessions are firstly identified, because the end of the user sessions is not registered in the weblogs. Subsequently, a clustering algorithm is executed to identify different types of users, and each user session is associated with one user type. Finally, one web tran-

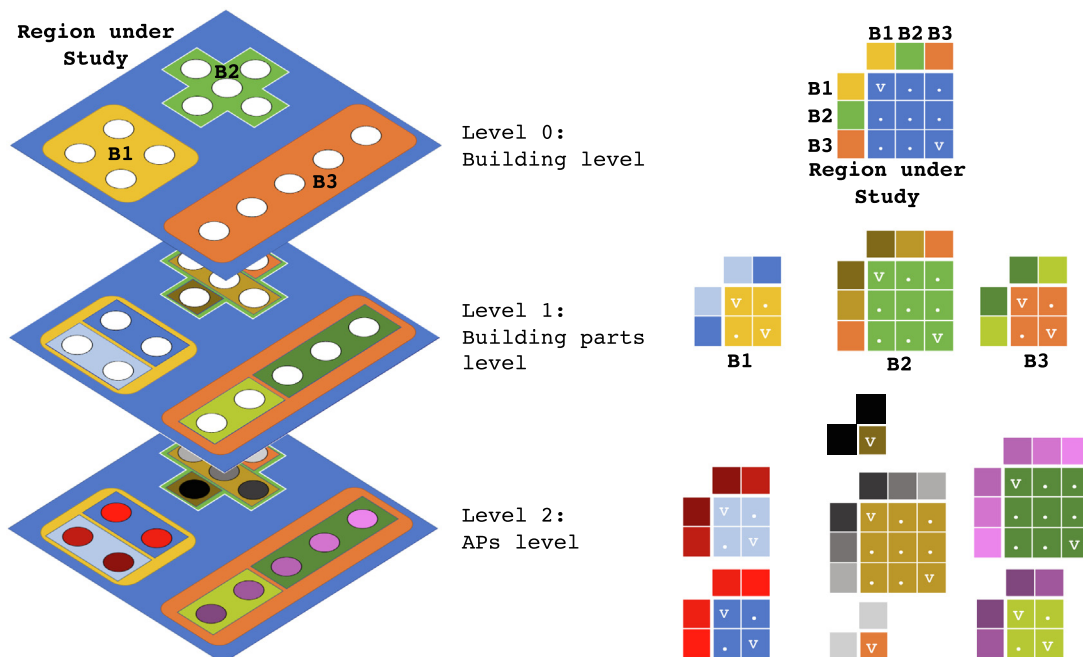


Fig. 2. Example of geospatial hierarchical modeling of user mobility.

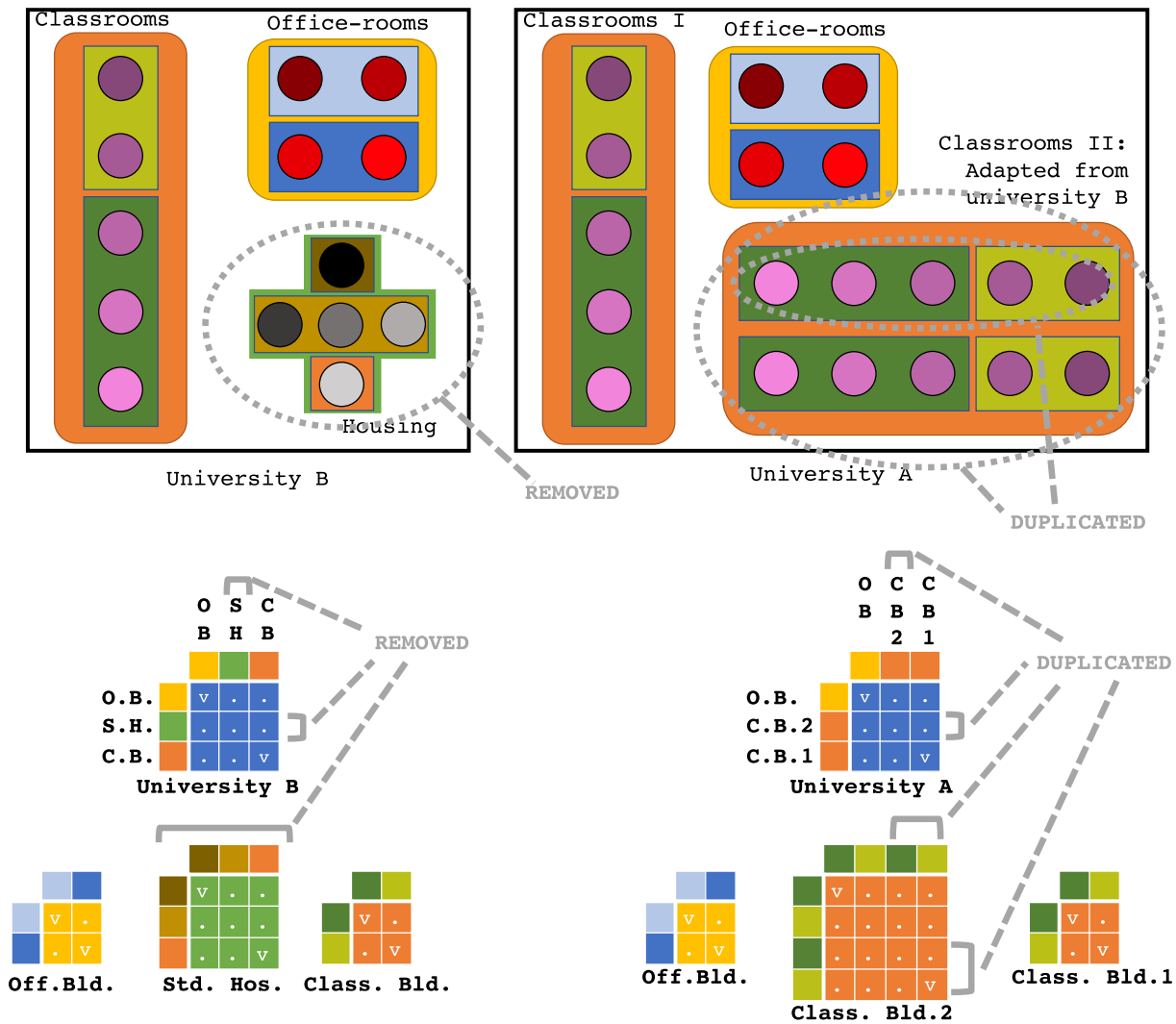


Fig. 3. Example of model adaptation between scenarios with different geospatial features.

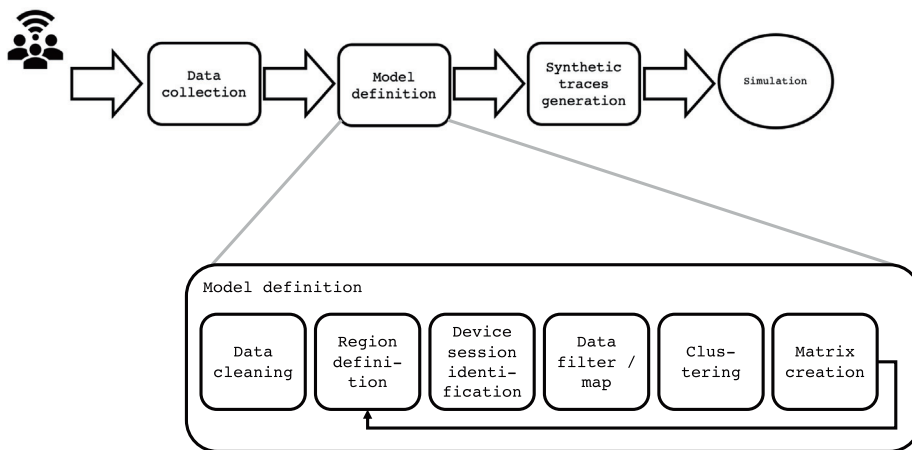


Fig. 4. Complete life-cycle of our proposed method for the hierarchical modeling of user mobility based on transition matrices.

sition matrix is generated for each user type using the set of user sessions of that user type.

We propose to revisit this method from e-business workload characterization (Kuruz et al., 2005) to be applied to the problem domain of user mobility in fog architectures. In our research problem, we deal with user connections to APs instead of web user requests.

One of the new challenges of applying or adapting this method to fog user mobility is the size of the problem. The number of web pages mapped to states is usually much smaller than the number of APs in a region under study. Our proposal solves this problem and the following subsections explain the adaptation of the life-cycle of this method (Fig. 4) for the user mobility modeling.

#### 4.1. First phase: data collection

Our problem domain only needs to characterize AP connection length and the handoffs to represent the user mobility. Consequently, the first phase, data collection, is in charge of monitoring the user mobility by collecting the wireless sessions trace logs that characterize the user tracks.

The most straightforward solution to gather the wireless sessions trace logs is to use already deployed Wi-Fi infrastructures. Several commercial alternatives are available in the market for the tracking of users using Wi-Fi infrastructure and the collection of the wireless sessions. Any of these alternatives allows us to gather data about the presence of users, and our problem requirements do not need an accurate location of the devices. The expected data gathered about the users is a log that stores and relates the user devices (the MAC address), the AP (the one that corresponds to the coverage area where the device is located in), and a timestamp.

The output of this phase is the wireless session trace log which contains periodical data of all the devices and APs in the area under study. Each line of this trace log includes a tuple of three elements  $\langle \text{time\_stamp}, \text{user\_id}, \text{ap\_id} \rangle$  stored periodically for all the devices and all the APs. By this, the wireless session of a given device can be generated and its handoffs can be also detected. From the point of view of a simulation of a fog infrastructure, a handoff is the only relevant event in the mobility of the users. A handoff involves that the service requests (or the data a device generates) come from a different AP and, probably, they also follow a different network route.

#### 4.2. Second phase: model definition

Using the wireless session trace logs, the goal of this phase is to generate a mobility model using this log. In the context of our work, we propose to define the user mobility through the connection time of the devices to a given APs (represented with a time vector) and the probability of a change of the AP coverage area (represented with a stochastic transition matrix) (Keramat Jahromi et al., 2016). This type of model has been previously implemented in other scenarios such as the characterization of e-business workloads with Customer Behavior Model Graphs (CBMG) (Menasce and Virgilio, 2000; Kuruz et al., 2005).

A straightforward use of the transition matrices could result in mapping each entry of the matrix with one AP where devices are connected to, considering the transitions as the probabilities of a device handoff to another AP coverage area. This first approximation is not suitable for medium-big scenarios with a high number of APs in the infrastructure. In those cases, the complexity of the modeling is increased, resulting even in an unsolvable problem. Consequently, we propose to group APs, instead of mapping one AP to one state, and to divide the problem scenario into zones and levels with different granularity.

For example, in the first level of a university campus, the region under study can be split into buildings, mapping each building with a zone (entry of the matrix) of the region under study. In successive iterations, the second levels (the buildings) are exploited by an isolated analysis of each of them, i.e., the modeling process is repeated for each building, which has to be split again into zones.

By this hierarchical definition, the zones of the region are modeled independently and, consequently, the model can be modified by, for example, removing or duplicating zones. This modification simply results in the modification (removing/duplicating) of the rows and columns of the transition matrices and the time vectors. If the final user of the model is interested in applying the results to other different (but with similar features) mobility scenarios, the adaptation of an existing model is easier because of the possibility of modifying the zones of the model.

Fig. 4 contains the steps we propose for the hierarchical modeling of the user mobility. In our hierarchical model, the analysis is split in successive levels of granularity to reduce the complexity, which results in the isolated repetition of some steps of this phase.

##### 4.2.1. Data cleaning

Data cleaning refers to the analysis of the data set to identify and remove samples that are not useful for mobility modeling. It is a very particular process that depends on each data set and the target of the study. For example, a user can be interested in removing: users with only one AP sample; samples associated with a specific AP because it is located outside the region under study, samples associated with a specific device, samples in a time interval (during the night), etc.

##### 4.2.2. Hierarchical region definition

Our proposal considers physical areas split into hierarchical zones. Each region is divided into zones, and the zones are recursively considered as sub-regions that are also divided into new zones. Each region corresponds to, or is modeled by, a pair of transition matrix and time vector. Similarly, the zones are related to the row and columns of the matrices/vectors.

Thus, we divide the characterization problem into levels in terms of geographic, topological, or other types of criteria. An independent modeling process needs to be carried out for each zone of a level recursively. This modeling process consists of the session identification, data filtering and mapping, clustering and transition matrix creation (Fig. 4).

The hierarchy starts with the higher level, where the region under study is divided into  $N$  zones, with  $N$  being a reasonable number, neither too small (which reduces the model accuracy) nor too large (which increases the model complexity). Each zone is mapped to one state and they include disjoint and geographically adjacent APs. Once the modeling process is carried out for the first level, the process is repeated independently for each particular zone. The hierarchy is constrained by several requirements: the hierarchy levels are defined until the level with a granularity of 1 AP per zone is reached; each new region that emerges from a zone only considers APs into the former zone, the new zones covers all the APs (surjective) and APs are mapped with only one zone (disjoint).

We include an example of hierarchical division in Fig. 5. The first level, level 0, divides the area under study into two zones which correspond to two buildings. The APs of each building are mapped to the corresponding zone. Once the model of level 0 is generated, the modeling process is repeated for each of the zones in level 0, i.e., a modeling process is executed for the first building (zone S0) and the second building (zone S1). These two new level 1 regions are split in terms of the wing of the buildings. This hierarchy is split in a reasonable number of levels until each zone only covers one AP, as in our example which requires three levels.

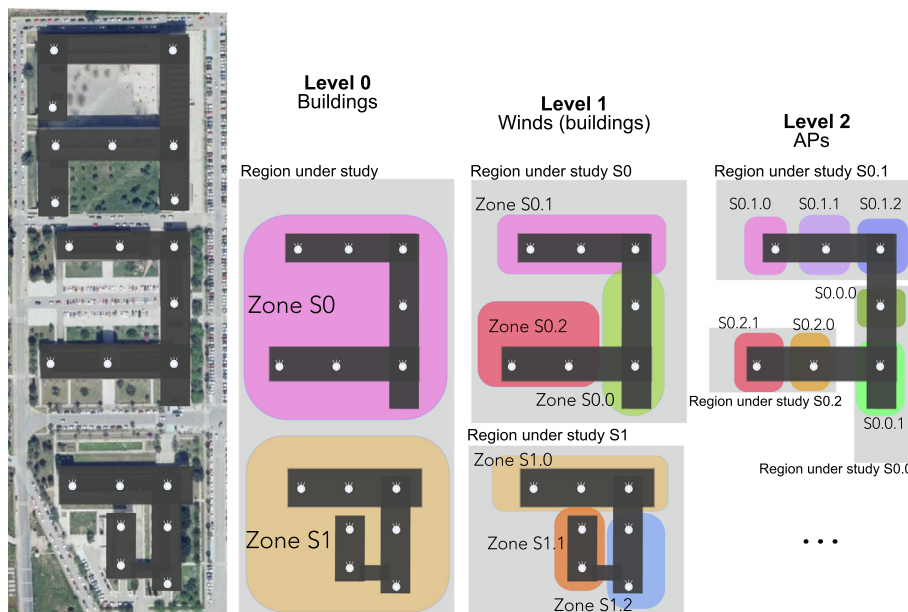


Fig. 5. Example of the hierarchical approach for mobility modeling.

The use of a hierarchical division of the region under study also facilitates the generalization of the resulting model. As we previously explained, the number of open mobility models is very reduced and the possibility to generalize the existing one is an important point. A hierarchical model easily allows to remove, duplicate, or modify some regions of the resulting model to adapt it to a different study case. Consider an open model that considers three buildings of a university campus, one building is a student housing, another one contains the teaching classrooms, and the teachers' office rooms are in the last one. If we need to adapt this model to a different campus, for example with three classroom buildings or without student housings, we can adapt it by duplicating the part corresponding to the classroom building or by removing the model of the housing building. This generalization can be also performed in terms of user types, popularity probabilities, etc.

#### 4.2.3. Device session identification

Once the data is loaded and cleaned, the next step is to identify the users. We assume that a MAC address corresponds to one user. Consequently, the selection of the samples of a user requires to filter by the MAC address. Not only the users need to be identified, but also the user sessions. A session is defined as each of the visits that a user does to the region under study during the time of the experiment. In other words, if a user leaves the region under study and comes back again, these two visits result in two different user sessions.

From the point of view of the infrastructure, a user session is defined as a device that enters the coverage perimeter defined by the levels of the study at a certain point in time, it is in the area for a specific period of time, and it leaves it. The Wi-Fi infrastructure does not register the disconnections in the data sets and, consequently, the session detection is conceptually determined when:

- The user changes between regions under study.
- None connections are detected after a certain period of time.

And these two criteria are carried out over the data set by detecting when:

- A sample of a user is associated with an AP that does not belong to any of the zones of the current area under study.

- The time between two successive connections of the same device is greater than a defined threshold value.

The definition of the threshold value for the user session identification implies carrying out a preliminary study to calculate the optimal threshold time for the data set that is being analyzed. For example, in our case study in Section 5, a range of defined threshold values are tested and, for all of them, the Euclidean distance between the number of sessions and the average connection time of these is calculated. The optimal threshold is determined by the smaller Euclidean distance, based on the idea that the best threshold is the one that balances the total number of sessions and the length of the sessions (Kurz et al., 2005).

#### 4.2.4. Data filter/map

Once the different user sessions are identified, not all their samples in the data set provide useful information. Consequently, the samples are filtered by keeping:

- The first sample of a consecutive subset of samples with the same AP identifier. This corresponds to the handoffs (AP changes) of a user session.
- The first and the last sample of each session. They identify the entry and exit points of the user session.

The result of this phase is a subset of samples that represents the session start, end and the AP changes. Fig. 1.c shows an example of the subset obtained from Fig. 1.b.

#### 4.2.5. User profiling: clustering

Human behavior is very different and it cannot be determined only with a general model, and the same pattern for all users is not realistic nor accurate. Consequently, it is important to split users with similar mobility patterns. The goal of user profiling or segmentation is to achieve a clear description of how the users in each cluster behave and move around the area under study. Clustering is the usual solution for this type of profiling and it is necessary to trade off the number of user types (as smaller as possible) and the suitability of the model (as accurate as possible) (Hardy, 1996).



We characterize the behavior of the users in terms of the percentage of time that they spend in a specific zone of the region under study. Consequently, each user session is characterized with a  $n$ -tuple probability vector, where  $n$  is the number of zones in the region under study. This vector represents the probability of the user being in each of the zones, calculated as the total time in each zone divided by the session length.

The selection of a clustering algorithm depends on many factors (Abbas, 2008). For example, in our case study, we implement the  $k$ -means algorithm because of its simplicity and its generalized use.

#### 4.2.6. Transition matrix and time vector creation

Once the users are split into different groups according to their behavior using a clustering process, a data model is generated for each of the clusters. This data model is made up of a transition matrix and a vector with the average stay length in each zone.

The transition matrix represents the transitions between zones, where each element  $a_{ij}$  of the matrix is the probability to go to a zone  $j$  when a user is in zone  $i$ . Each element  $a_{ij}$  is calculated as the total number of handoffs from  $i$  to  $j$  divided by the total number of handoffs for all the sessions of the user in a given cluster group. Additionally, the matrix is completed with two additional elements labeled as  $IN$  and  $OUT$ , that respectively represent the start and the end of a user session. In other words,  $a_{INj}$  represents the probability that the zone  $j$  is the starting point of a user. On the contrary,  $a_{iOUT}$  represents the probability that the zone  $i$  is the last zone of a user session. Consequently,  $a_{OUTj} = 0 \forall j$  and  $a_{iIN} = 0 \forall i$ .

The stay length vector indicates the average time that a given user type spends in each zone expressed in time units. Each element of the vector  $v_i$  is calculated as the sum of the time of all the visits of a user to a zone  $i$  divided by the total number of visits to that zone.

The data models provide a double utility. First, they serve to characterize the movement of the users of the different groups. They also are, if necessary, the input for the next phase, the synthetic generation of traces.

#### 4.3. Third phase: generation of the synthetic traces

The objective of this phase is to generate a synthetic data set using the data model obtained in the previous phase. The most important advantage of the generation of synthetic traces is that the resulting model can be extrapolated to scenarios with other features (for example, changing the number of users, building, removing some user type, etc.). Additionally, the use of synthetic traces also guarantees additional anonymization of sensitive information.

The synthetic data is also a three elements tuple:  $\langle \text{time\_stamp}, \text{user\_id}, \text{ap\_id} \rangle$ , in front of the original  $\langle \text{time\_stamp}, \text{device\_id}, \text{ap\_id} \rangle$  tuple. The user identifier for synthetic tuples is auto-incrementally generated during the synthetic generation.

Each user trace is generated randomly using the mobility model. Each time synthetic users are generated, we first randomly determine the user types they belong to, using either weights determined manually or by the popularity distribution of the real data set. Using the probabilities of the transition matrix of that user group, the entry point (zone) is determined. The time in that zone is determined with the stay period vector and randomly generated following an exponential distribution. The second zone in the user trace is again determined with the probabilities of the transition matrix. This is repeated subsequently until the zone  $OUT$  is chosen randomly.

The times of each sample,  $\langle \text{time\_stamp}, \text{device\_id}, \text{ap\_id} \rangle$ , are mapped to the simulation time, which starts in time 0 and are incremented in simulation time units. The generation of the users

is determined by increasing the time of their initial sample with the average time between user arrivals. This time can be manually fixed or we can use the one obtained from the real data set. An exponential distribution is used for the generation of these user arrivals.

## 5. Case study: the campus of the University of the Balearic Islands

We apply our hierarchical decomposition method in the campus of the University of the Balearic Islands (UIB). UIB is a small-medium size university with approximately 14,000 students, 1,000 teachers and 800 administrative staff. The Wi-Fi infrastructure of the university is composed of 425 APs, distributed along with the 18 buildings in the campus, which are located in a secluded area of the city, specific for the campus.

In the remainder of this section, we detail the specific aspects to apply the method proposed in Section 4 to the case of the UIB. At the end of the section, some of the results obtained for the mobility model are also commented.

### 5.1. Data collection

The APs deployed in the UIB are from the company Aruba Networks. These APs include the ALE technology (Analytics and Location Engine) which collects location and mobility data of the user devices (Gouin-Vallerand and Rousseau, 2019). Consequently, we easily implement the data collection phase by using the methods provided by ALE.

We first use the method `access_points` to collect the list and data of all the APs deployed in the Wi-Fi infrastructure of the campus. Secondly, ALE offers data about the location of the devices/users with three main methods: `proximity`, `presence` and `station`. We choose `proximity`, since it registers the MAC addresses of all the detected user devices, whether or not they are logged to the Wi-Fi network. Method `proximity` also associates the user device with the closest AP. Fig. 6 shows an example of the JSON file returned by the method `proximity`. In this example, data about only one device and one AP are represented, but a usual execution returns all the devices for all the APs.

Note that our API vendor only offers real-time data. Users interested in analyzing historical data need to store the data persistently, for example, in a database. We implemented a Python script which requests the method `proximity`, parses the returned JSON file, and stores into a database a 3-tuple  $\langle \text{time\_stamp}, \text{user\_id}, \text{ap\_id} \rangle$  for each device in the coverage areas. ALE is configured to return the one-way hashed MAC addresses to protect the privacy of the users. The hash function is *salted* for periods of one day to avoid the matching of traces with real users. The script is executed each minute, so the database stores the device-AP connections with an accuracy of 1 min. The database stores the hashed MAC address of the user device as the device identifier, and the MAC address of the AP as the AP identifier.

For this study, we gathered the Wi-Fi probes with a minute frequency during one week, November 9th-15th 2020, obtaining 122 MB of raw data. In this period, partial mobility restrictions were applied due to COVID-19 pandemic. The presence was reduced to approximately half of the usual number of users. Additionally, we conduct several studies, by dividing the data of this week in different time subsets. More concretely, 9 studies are considered: one for each single day of the week, one for the five working days, and one for the weekend. We chose one of them, the experiment of Monday 9th November, to be analyzed in the results of this paper. The selected experiment is a suitable example for any of the working days. If the reader is interested in the other results,

```
{ "Proximity_result": [{
  "msg": {
    "sta_eth_mac": {
      "addr": "FCF8AECC8E20"
    },
    "associated": true,
    "hashed_sta_eth_mac": "DBB3D96B0AA540A2839B39071C3154760775AD50"
    "ap_name": "9c:1c:12:c0:19:5a"
    "radio_mac": {
      "addr": "B45D50F94E30"
    },
    "target_type": "TARGET_TYPE_STATION"
  },
  "ts": 1521133688,
}]}
```

Fig. 6. Example of device tracking log obtained with a commercial tool (Aruba Analytics and Location Engine).

Table 1  
Division into zones of the region under study (lv0, -, -)

Zone id.	Number of APs	Zone id.	Number of APs
bldg_AT <sup>1</sup>	36	bldg_JO	63
bldg_CEP	14	bldg_MA	43
bldg_CJ	7	bldg_MEN	8
bldg_CL	1	bldg_MO	52
bldg_CTI	9	bldg_RES	18
bldg_EIV	11	bldg_RL	42
bldg_GC	35	bldg_SCT	27
bldg_IE	7	bldg_SE	8
bldg_ITD	16	bldg_SL	28

<sup>1</sup> Zone selected as the region under study for level 1.

Table 2  
Division into zones of the region under study (lv1, bldgAT, -).

Zone id.	Number of APs	Zone id.	Number of APs
basement_fl	3	1st_fl_East	6
0_fl_North	7	1st_fl_North	7
0_fl_East <sup>1</sup>	6	2nd_fl	3

<sup>1</sup> Zone selected as the region under study for level 2.

they can be accessed in the data repository associated with this work <sup>3</sup>.

## 5.2. Model definition

### 5.2.1. Data cleaning

In the data cleaning process we remove the samples of sessions formed by a single connection. Thus we avoid samples of passersby who simply pass near the campus but do not make any route within it.

### 5.2.2. Region definition

Using the hierarchical definition of our method, we define three levels in the division of the university campus. At the initial level (level 0), the entire campus is considered, dividing it into as many zones as buildings. Table 1 shows the 18 zones, with a range of APs per zone between 1 and 63.

At the second level (level 1), we study the movements individually for each building. We split each building into floors and

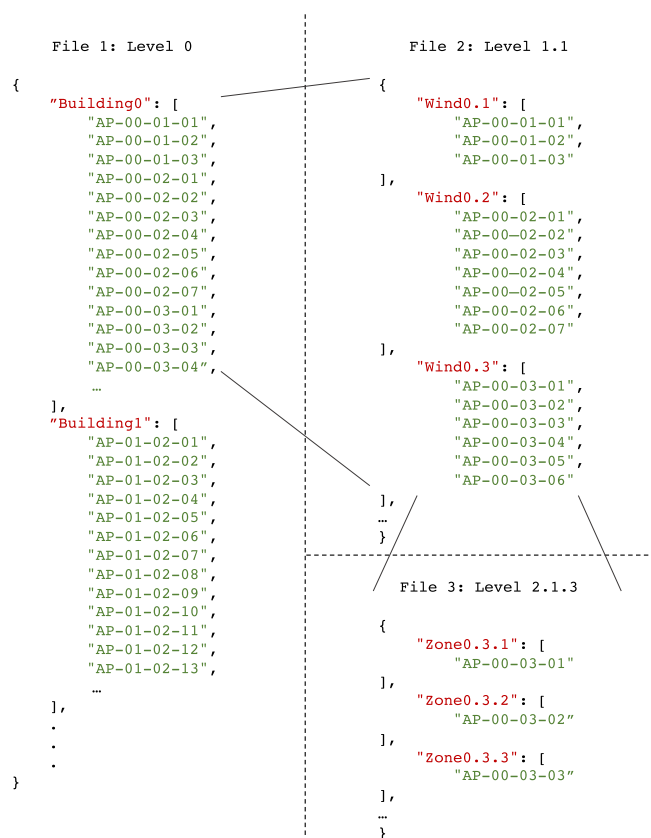


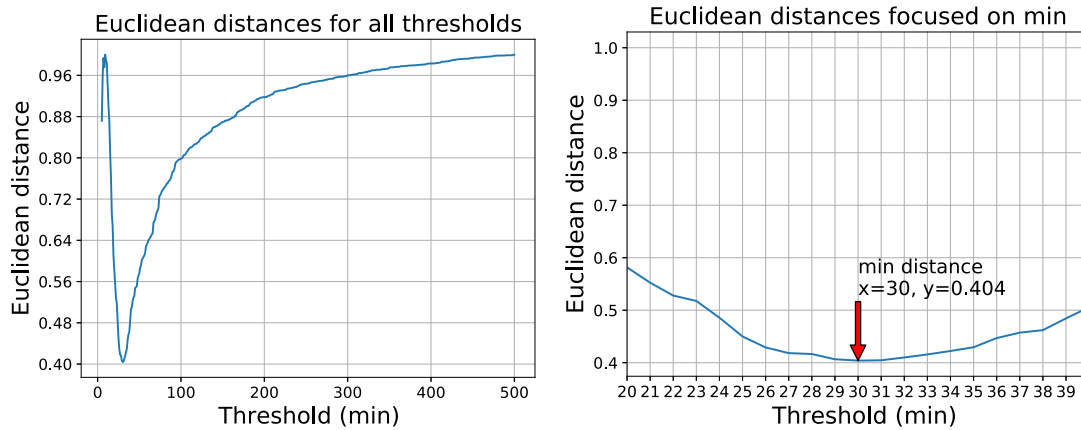
Fig. 7. Example of three JSON files for the hierarchical level definition.

wings. Finally, at the last level (level 2), each floor-wing is divided into zones that only includes one AP.

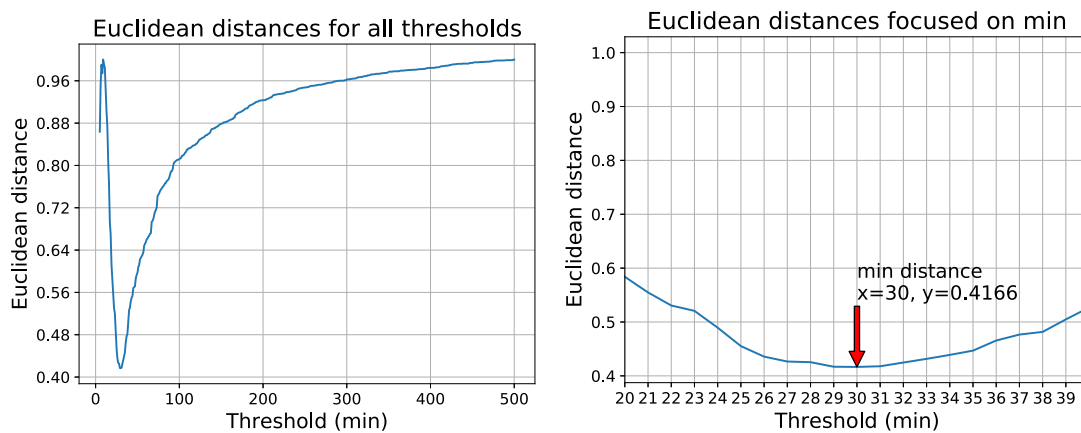
Each region under study has a 3-tuple identifier based on the level, the building acronym and the floor-wing,  $\langle level, building, wing \rangle$ . For example,  $\langle level_0, -, - \rangle$  refers to the analysis of level 0 that corresponds to the movement of users between buildings,  $\langle level_1, buildingA, - \rangle$  refers to the analysis of mobility of the users between wings of Building A, or  $\langle level_2, buildingA, northSecondFloor \rangle$  refers to the analysis of the user movement between the APs in the north wing of the second floor of Building A.

The high number of regions by level inhibits a detailed description of each of them in this article. For this reason, we only explain one region from each level. More concretely, the explained regions

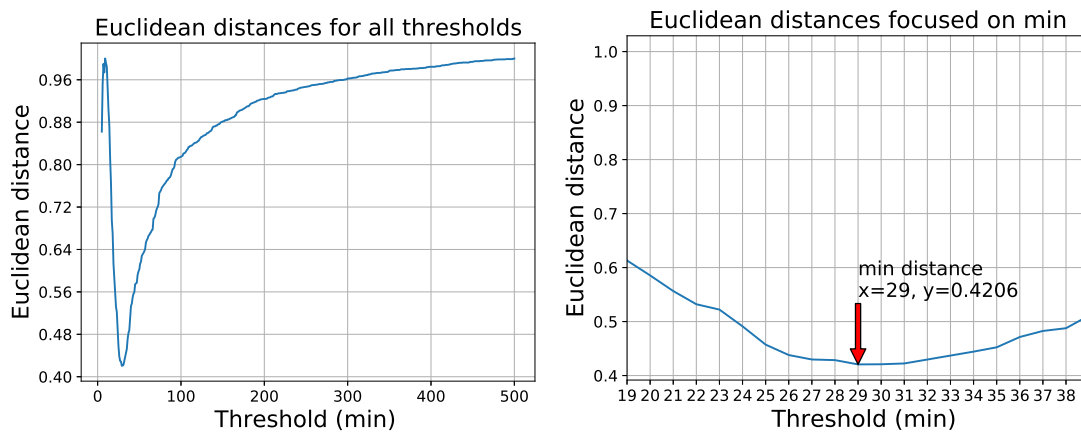
<sup>3</sup> <https://github.com/ocsix6/mobility-analysis-corporate-wifi>



(a) Threshold calculation for  $\langle lv0, -, - \rangle$



(b) Threshold calculation for  $\langle lv1, bldgAT, - \rangle$



(c) Threshold calculation for  $\langle lv2, bldgAT, 0flE \rangle$

Fig. 8. Euclidean distances between number of user sessions and average session times.

under study are  $\langle lv0, -, - \rangle$ ,  $\langle lv1, bldgAT, - \rangle$ , and  $\langle lv2, bldgAT, 0flE \rangle$ , all of them from the period corresponding to Monday 9th November 2020. The results for the other regions under study are published in the data repository <sup>4</sup>.

Table 1 shows the number of APs in each zone of the region under study  $\langle lv0, -, - \rangle$ . At this level, one zone is created for each of the buildings on the campus.

Table 2 shows the number of APs in each of the zones defined in the building *bldg\_AT*, used as an example of the region under study for level 1. It is a four-storey building and two wings (north and east), with an approximate occupation of 800 students and 100 teachers where, due to the architecture of the building, we delimit

<sup>4</sup> <https://github.com/ocsix6/mobility-analysis-corporate-wifi>

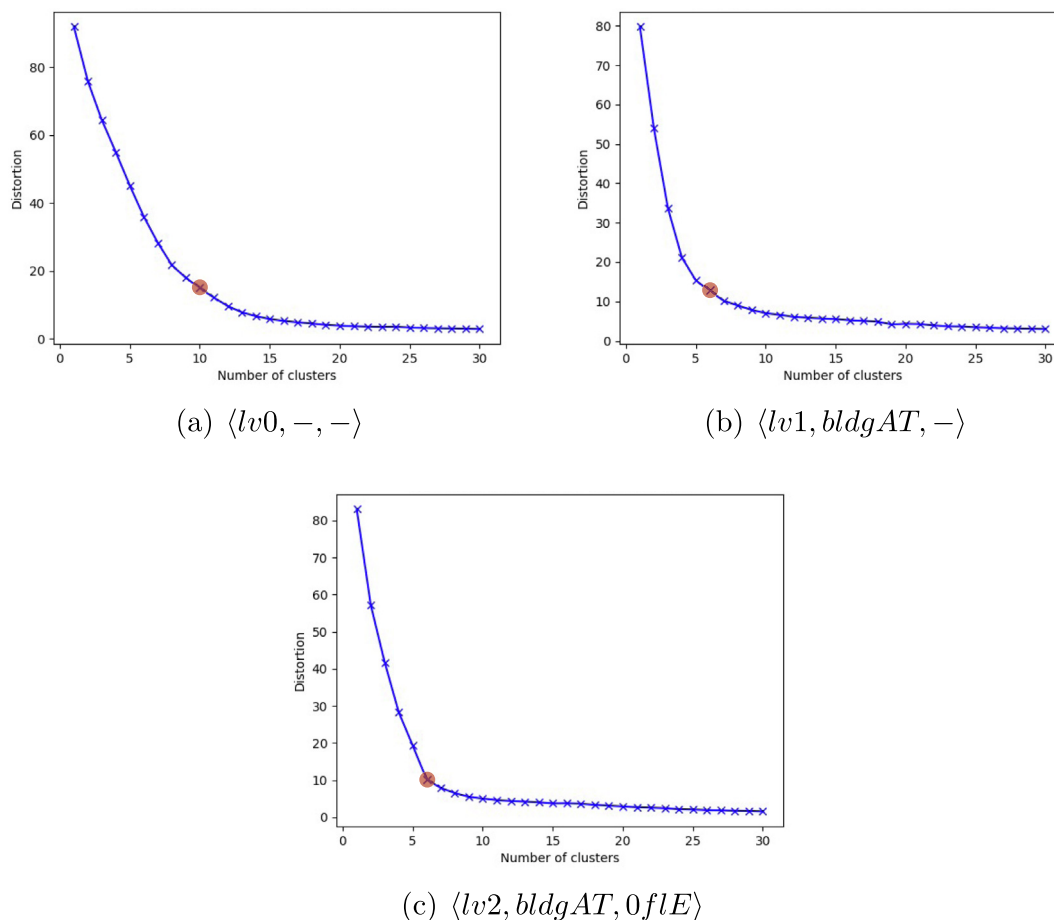


Fig. 9. Elbow method to detect the optimal number of clusters considering the distortion metric and with a maximum number of 30 clusters.

one single zone for the basement and the second floor, and two zones –one for each wing– for the ground floor and the first floor.

Finally, the wing *0.fl.East* is selected as an example of the region under the study of level 2. This is the deepest level in the region hierarchy and each APs is defined as an isolated zone. Thus, 6 zones are considered, one for each AP.

We implement the definition of each of the levels using a JSON file that specifies the APs included in each zone. Fig. 7 shows the source code of the JSON files corresponding to the three areas under study explained in this section.

### 5.2.3. Device session identification

As we commented in Section 4.2.3, the user disconnections are not registered in the trace logs and user sessions may be determined manually. An iterative process, with an incremental threshold value, calculates the number of sessions and the average connection time over all the users for each threshold value. The desired threshold is the one that reduces both the number of sessions and the average session length. Consequently, we use the Euclidean distance because it indicates the threshold value which balances both metrics. The threshold value depends on each data set and, consequently, the process of threshold definition must be repeated for each region under study.

Fig. 8 represents the evolution of the Euclidean distance as the threshold value is increased. Three plots are presented, one for each region included in this section.

For  $\langle lv0, -, - \rangle$ , the smaller Euclidean distance results in a threshold of 30 min. This threshold corresponds to the case of 11,520 user sessions with 4,883.7 s of average session time. In the example for level 1,  $\langle lv1, bldgAT, - \rangle$ , the threshold is also

defined in 30 min, with 11,570 user sessions and 4,737.0 s of average time. Finally,  $\langle lv2, bldgAT, 0flE \rangle$  results in a threshold of 29 min with 12,599 users and 4,252.7 s of session time.

Once the threshold value is selected, the user sessions can be recalculated considering the criteria explained in Section 4.2.3.

### 5.2.4. Data filter/map

The data filtering is straightforward, and it only requires to loop through each user session, taking into account the threshold value of the previous phase. The process keeps the first sample in the session, the first sample of a new AP subset, and the last sample in the user session. All the other samples are removed.

### 5.2.5. User profiling: clustering

In our case study, we implement the k-means algorithm for the clustering of the user sessions. Note that the user mobility is characterized by their zone probability vector, the vector that indicates the probability for a given user of being in a given zone of the region under study. Consequently, we first calculate those vectors for each user session.

In the case of the k-means algorithm, similarly to some other clustering methods, the number of clusters is an input value of the algorithm. Thus, a previous study of the suitable number of clusters needs to be performed. We calculate it through the elbow method (Syakur et al., 2018). The objective behind this is to minimize both opposite metrics of intergroup distance<sup>5</sup> and the number

<sup>5</sup> We measure the intergroup distance using the distortion metric. The distortion is calculated as the average of the squared distances from the cluster centers of the respective clusters.

of clusters. The elbow method determines the optimal number of clusters  $k$  like the number of clusters from which the improvement, in terms of intergroup distance, obtained by increasing this number of clusters is so small that it does not justify the additional cost (complexity) of increasing the number of clusters. This is graphically determined in the *elbow* of the curve, hence the name of the method (Fig. 9).

Fig. 9 shows the *elbows* for the three regions under study that we present in this section. We can see that the optimal number of clusters is 10 for  $\langle lv0, -, - \rangle$ , 6 for  $\langle lv1, bldgAT, - \rangle$ , and 6 for  $\langle lv2, bldgAT, 0flE \rangle$ . Once the optimal number of clusters is defined, the  $k$ -mean algorithm is executed and the clusters for each region are generated.

### 5.2.6. Transition matrix and time vector creation

The final step is the generation of the transition matrices and the time vectors. In this step, we obtain one pair of matrix–vector for each user cluster, i.e., ten matrices and vectors for  $\langle lv0, -, - \rangle$ , six for  $\langle lv1, bldgAT, - \rangle$ , and six for  $\langle lv2, bldgAT, 0flE \rangle$ , because the total numbers of user profiles (clusters) for each case are 10, 6, and 6.

Instead of representing these 22 matrices/vectors, we only show one pair of matrix–vector from each of the three regions. The other ones are available in the open repository. Eq. 1 shows the transition matrix for one of the 10 user profiles generated in  $\langle lv0, -, - \rangle$ , and Eq. 2 the average stay time in minutes for each building of that region. Eq. 3 and Eq. 4 respectively show the transition matrix and the time vector of one of the 6 user profiles in  $\langle lv1, bldgAT, - \rangle$ . Finally, Eq. 5 and Eq. 6 correspond to the mobility model of one of the six profiles in  $\langle lv2, bldgAT, 0flE \rangle$ .

$$\begin{matrix}
 & \text{IN} & \text{OUT} & \text{bldg\_AT} & \text{bldg\_CEP} & \text{bldg\_CJ} & \text{bldg\_GC} & \text{bldg\_IE} & \text{bldg\_ITD} & \text{bldg\_JO} & \text{bldg\_MA} & \text{bldg\_MO} & \text{bldg\_RES} & \text{bldg\_RL} & \text{bldg\_SCT} & \text{bldg\_SL} \\
 \text{IN} & \left[ \begin{array}{cccccccccccccc}
 0 & 0 & 0.0033 & 0 & 0 & 0.0033 & 0.0042 & 0 & 0.0025 & 0.0017 & 0.0042 & 0.9791 & 0 & 0 & 0.0017 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0.364 & 0 & 0 & 0 & 0 & 0 & 0.091 & 0 & 0.273 & 0 & 0.273 & 0 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0.333 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0667 & 0.4 & 0.0667 & 0.0667 & 0.0667 & 0.0667 \\
 0 & 0.2069 & 0 & 0 & 0.0345 & 0 & 0 & 0 & 0.069 & 0 & 0 & 0.69 & 0 & 0 & 0 & 0 \\
 0 & 0.5 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 \\
 0 & 0.364 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0909 & & 0 & 0.545 & 0 & 0 & 0 & 0 \\
 0 & 0.182 & 0.273 & 0 & 0 & 0 & 0.091 & 0 & 0.091 & 0 & 0 & 0.364 & 0 & 0 & 0 & 0 \\
 0 & 0.5 & 0 & 0 & 0 & 0.143 & 0 & 0 & 0 & 0 & 0 & 0.286 & 0.0714 & 0 & 0 & 0 \\
 0 & 0.952 & 0.0033 & 0.25 & 0.0008 & 0.0057 & 0.019 & 0.0025 & 0.0041 & 0.0041 & 0.0041 & 0 & 0.0025 & 0 & 0 & 0 \\
 0 & 0.2 & 0 & 0 & 0 & 0.2 & 0 & 0 & 0 & 0 & 0.2 & 0.4 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
 \end{array} \right] \\
 \end{matrix}
 \tag{1}$$

$$\begin{matrix}
 & \text{bldg\_AT} & \text{bldg\_CEP} & \text{bldg\_CJ} & \text{bldg\_GC} & \text{bldg\_IE} & \text{bldg\_ITD} & \text{bldg\_JO} & \text{bldg\_MA} & \text{bldg\_MO} & \text{bldg\_RES} & \text{bldg\_RL} & \text{bldg\_SCT} & \text{bldg\_SL} \\
 \text{IN} & \left[ \begin{array}{cccccccccccc}
 53.32 & 40.48 & 1.01 & 27.46 & 14.07 & 5.62 & 17.2 & 28.5 & 19.89 & 89.59 & 77.22 & 9.08 & 9.17 & ] \\
 \end{array} \right] \\
 \end{matrix}
 \tag{2}$$

$$\begin{matrix}
 & \text{IN} & \text{OUT} & \text{1st\_fl\_East} & \text{1st\_fl\_North} & \text{0\_fl\_East} & \text{0\_fl\_North} & \text{2nd\_fl} \\
 \text{IN} & \left[ \begin{array}{cccccc}
 0 & 0 & 0.0364 & 0.0182 & 0.0364 & 0 & 0.909 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0.4 & 0 & 0 & 0 & 0 & 0.6 \\
 0 & 0.286 & 0 & 0 & 0 & 0 & 0.714 \\
 0 & 0.25 & 0 & 0 & 0 & 0 & 0.75 \\
 0 & 0.5 & 0 & 0 & 0 & 0 & 0.5 \\
 0 & 0.568 & 0.243 & 0.081 & 0.081 & 0.027 & 0 \\
 \end{array} \right] \\
 \end{matrix}
 \tag{3}$$

$$\begin{matrix}
 & \text{1st\_fl\_East} & \text{PIN} & \text{0\_fl\_East} & \text{0\_fl\_North} & \text{2nd\_fl} \\
 \text{IN} & \left[ \begin{array}{cccc}
 6.92 & 15.03 & 14.93 & 6.23 & 51.85 \\
 \end{array} \right] \\
 \end{matrix}
 \tag{4}$$

$$\begin{matrix}
 & \text{IN} & \text{OUT} & \text{AP-OO-03-01} & \text{AP-OO-03-02} & \text{AP-OO-03-03} & \text{AP-OO-03-04} & \text{AP-OO-03-05} & \text{AP-OO-03-06} \\
 \text{IN} & \left[ \begin{array}{cccccc}
 0 & 0 & 0.0058 & 0.953 & 0.035 & 058 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0.75 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 \\
 0 & 0.825 & 0.0164 & 0 & 0.104 & 0.027 & 0.016 & 0.011 & 0 \\
 0 & 0.333 & 0 & 0.667 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0.5 & 0 & 0.333 & 0.167 & 0 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \end{array} \right] \\
 \end{matrix}
 \tag{5}$$

$$\begin{matrix}
 & \text{AP-OO-03-01} & \text{AP-OO-03-02} & \text{AP-OO-03-03} & \text{AP-OO-03-04} & \text{AP-OO-03-05} & \text{AP-OO-03-06} \\
 \text{IN} & \left[ \begin{array}{cccccc}
 4.6 & 32.4 & 6.35 & 8.58 & 0 & 0 & ] \\
 \end{array} \right] \\
 \end{matrix}
 \tag{6}$$

Additionally, the chord diagrams of the transition matrices are generated to improve the interpretability of the models. Fig. 10 presents the chord diagrams of the three examples presented in the previous paragraph, where the arcs represent the flow of the users between zones of a region, and the sizes of the arcs represent the frequency (probability) of the flows.

### 5.3. Generation of the synthetic traces

Once the user mobility model is obtained, it can be incorporated in a simulation/emulation tool. There are two alternatives, the incorporation of the data of the model (matrices and vectors) or the creation of a synthetic trace log that is read by the simulator.

Our mobility model leaves the simulator designer the responsibility of defining the user arrival rate and the popularity of the user types. Thus, the simulator designer is able to change the *workload* of the system in terms of the number of users.

We implemented a Python script that randomly generates users and assigns them to a user type. For each new random user arrival, the user track is generated with the transition matrix and the time of the handoffs is determined with the time vector.

### 5.4. Result analysis

The analysis of the results consists of twofold validations that test the correctness of the obtained model and its complexity.

The first, in Section 5.4.1, checks if the model generated with our method reflects accurately the movement of the users. For this case, the comparison with other previous works is not required, because we can compare the resulting model with the real raw data. In any case, we do not either compare our method with other research works because of our hierarchical definition, that is not studied in any previous works. Comparison with non-hierarchical models is not possible because the transition matrices have different shapes (different numbers of matrices and matrix sizes) and meaning.

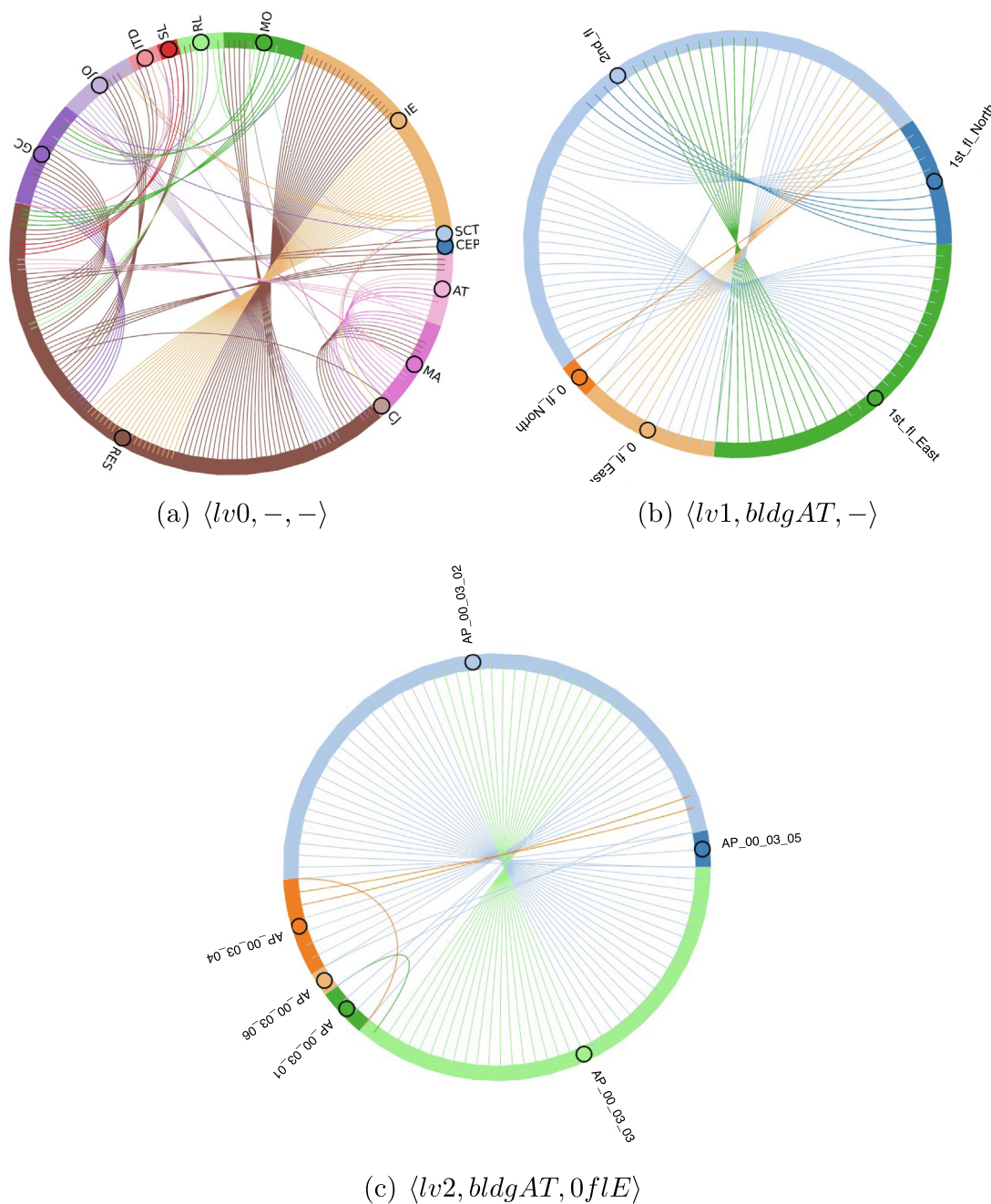


Fig. 10. Chord diagrams for three examples of the regions under study.

Secondly, in Section 5.4.2, the complexity and extrapolation of the resulting hierarchical model is evaluated and compared with a previous proposed non-hierarchical model (the Origin–Destination matrix) (Barbosa-Filho et al., 2018).

5.4.1. Mobility model analysis

We have designed an experiment to analyze the quality of our model generation that consists of obtaining a second mobility model considering the synthetic traces generated with the first model. The greater the statistical similarity between both models, the best accuracy our proposal shows. We compute, and analyze, this statistical similarity with the mean square error (RMSE) over the transition matrices and the time vectors of each user.

The complete process for the analysis of the case study is: (i) the gathering of real log traces; (ii) the generation of the mobility

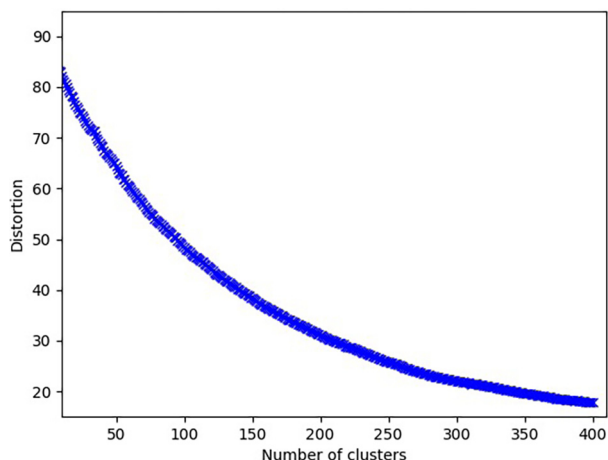
model of those traces; (iii) the creation of synthetic traces from the mobility model obtained; and (iv) the generation of the mobility model of these latter synthetic traces. Finally, the RMSE is calculated between the models obtained in (ii) and (iv).

The user types in (ii) and (iv) are mapped and the RMSE is calculated between the matrix/vector in each user type of (ii) and its corresponding matrix/vector in (iv). Table 3 shows the average and maximum value of the RMSE for both the transition matrix and the time vector. Additionally, the error percentages are also included between brackets.

The error levels are smaller in the case of the transition matrix, whose average values range between 7.7% and 1.5% and they are always smaller than 9.8%. In the case of the time vector, the average error level ranges between 4.5% and 16.8%, and the maximum RMSE is 20.08%. Note that the second region under study,

**Table 3**  
Maximum and mean value of the RMSE for the matrices and vectors of each region under study.

Region	Transition matrix		Time vector (s.)	
	avg. RMSE	max. RMSE	avg. RMSE	max. RMSE
$\langle lv0, -, - \rangle$	0.025 (2.5%)	0.057 (5.7%)	25.449 (4.5%)	62.207 (11.1%)
$\langle lv1, bldgAT, - \rangle$	0.077 (7.7%)	0.098 (9.8%)	25.634 (16.8%)	31.651 (20.8%)
$\langle lv2, bldgAT, OfE \rangle$	0.015 (1.5%)	0.038 (3.8%)	6.268 (5.6%)	17.826 (16.0%)



**Fig. 11.** Elbow method to detect the optimal number of clusters considering the distortion metric and with a maximum number of 400 clusters.

$\langle lv1, bldgAT, - \rangle$ , is the one with highest differences between both mobility models, and the case that increases the general error levels.

In general terms, we can conclude that the obtained model is more accurate in terms of the flow of the users between zones than in the time spent in a given zone.

**5.4.2. Hierarchical design analysis**

Our mobility characterization method is based on hierarchical modeling to minimize the impact of large scale scenarios. This hierarchical subdivision of the problem reduces the computational cost of subsequent analysis' tasks and even enables its distributed processing. To answer the RQ1, which states that our model reduces the complexity of the resulting model, and the execution time to obtain it, we have compared our hierarchical model with a non-hierarchical model, such as the one defined in previous related works, the Origin–Destination matrix (Barbosa-Filho et al., 2018).

Fig. 11 shows the number of optimal clusters for a non-hierarchical process in which each AP of our case study is mapped with a different zone. This is equivalent to one-level modeling with 425 zones. It is observed that the optimal number of clusters is much higher in the non-hierarchical modeling, resulting in 158 clusters, in opposition to the maximum number of 10 clusters across all the levels and regions in the hierarchical modeling. A smaller number of clusters reduces the complexity of the model and the execution times of the modeling process, and we have observed a very significant reduction from 158 to 10 clusters.

Moreover, the elbow point of the non-hierarchical model (Fig. 11) is less clear than the ones in the hierarchical model (Fig. 9). The graph in the non-hierarchical case has a slope, or gradient, closer to  $-1.0$ . This means that an increase in the number of

clusters could improve the distortion without a high increase of the cost/complexity of the clustering, for this non-hierarchical model.

As it has been commented in the previous paragraph, a higher number of clusters involves higher execution times. Table 4 shows the execution times for the three examples of the case study and the non-hierarchical modeling. The total execution time is presented along with the desegregated time for the calculation of the optimal number of clusters, and the clustering algorithm. As it is observed, the execution time has an important improvement for the case of hierarchical modeling. In general terms, the common execution time for each region of our case study is around 25 s, except for the case of the level 0, which increases up to 198 s because it is the only one with more than 10 zones. Considering that the hierarchy modeling of the campus defined 18 areas of level 1 and 74 areas of level 2, the execution time for all the regions under study is around 2500 s, five times shorter. Additionally, the execution could be parallelized very easily.

Finally, the goodness of the non-hierarchical modeling is measured with the RMSE (Table 5). If we compare the RMSE with the ones of the hierarchical modeling (Table 3), it is observed that, although the average RMSE is better in the case of the transition matrices of the non-hierarchical modeling, the maximum value of the RMSE is similar to the worse case in the hierarchical modeling. On the contrary, the error level for the time vector is better for the case of the non-hierarchical modeling.

To sum up, the hierarchical modeling shows similar error levels in the transition matrices, but with a worse behavior in the time vector. The great benefit of a hierarchical method is in the complexity both in the number of clusters and the execution time of the modeling.

The second research gap that our proposal deals with is to allow the extrapolation or adaptability of the resulting models between scenarios with different geospatial features. To answer to this second research question (RQ2), we have extrapolated the mobility model obtained in this case study. Remember that the analysis of the results of our case study are focused on one building (bldgAT) in a university campus with 18 buildings. We have extrapolated this real scenario to a fictitious one with only 3 buildings with similar features to bldgAT. To do that, we replicate the model of bldgAT, and we adapt each of those three replicated transition matrices to create buildings with different number of winds and floors.

The new modified model is used to generate synthetic traces. An additional modeling process is applied to compare the results with the former model of bldgAT. The results show that the average

**Table 4**  
Execution times for the modeling process.

Region	Execution time (s.)		
	Total	Elbow	Clustering
Non-hierarchical	14119.22	13816.04	30.64
$\langle lv0, -, - \rangle$	197.95	157.86	1.97
$\langle lv1, bldgAT, - \rangle$	25.48	3.22	0.17
$\langle lv2, bldgAT, OfE \rangle$	23.89	2.74	0.15

**Table 5**  
Maximum and average value of the RMSE for the non-hierarchical modeling.

	Transition matrix		Time vector (s.)	
	avg. RMSE	max. RMSE	avg. RMSE	max. RMSE
Non-hierarchical	0.019 (1.9%)	0.088 (8.8%)	4.981 (0.6%)	25.475 (3.1%)

RMSE are 2.85% and 4.31% respectively for the transition matrix and the time vector of *bdgAT*.

## 6. Conclusions

We have proposed a method for user mobility characterization based on a hierarchical definition of regions to (RQ1) reduce the complexity of creating a mobility model based on transition matrices, and to (RQ2) increase the adaptability of the obtained models between scenarios with different geospatial features. The applicability of the method has been tested in a case study located in the campus of the University of the Balearic Islands.

The case study was analyzed in terms of the RMSE between the model obtained from the real data traces and the model obtained from the synthetic traces created with the former model. Additionally, a non-hierarchical study of the campus was also performed.

By comparing the results between the hierarchical and the non-hierarchical study, we observed that the complexity of the modeling process is much lower in the first one, which answered the RQ1. Although the non-hierarchical model showed smaller values for the RMSE, both studies showed acceptable error levels. Additionally, RQ2 was answered by extrapolating the obtained hierarchical model to a scenario with different geospatial features. The resulting model was used to create synthetic mobility which showed low RMSE values.

We propose three future research lines that emerge from the proposal of our method: the integration of geospatial hierarchical mobility models in a fog simulator, such as *iFogSim* (Gupta et al., 2017) or *YAFS* (Lera et al., 2019); the extension of other mobility models, instead of the ones that use a transition matrix, with a geospatial hierarchical design; and, finally, the adaptation of the proposed method to other problem scenarios, such as the prediction of the user behavior.

In terms of the case study, an interesting future work is to apply our method to a case study where the user mobility data is gathered with a different wireless technology, for example a 5G network instead of a Wi-Fi. This would prove that our method is not influenced by the underlying network technology that is used to gather user mobility data. The main difficulty to carry it on in the future is to get access to these types of data, which are only available in the infrastructures of telecommunication companies.

## Data statement

The data generated in this study and the source code of the scripts are public available in the repository <https://github.com/oc-six6/mobility-analysis-corporate-wifi>.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgements

Funding: Project TIN2017-88547-P supported by the Spanish Government (MCIN/ AEI /10.13039/501100011033/) and the European Commission (FEDER Una manera de hacer Europa).

## References

- L. Eeckhout, K. de Bosschere, H. Neefs, Performance analysis through synthetic trace generation, in: 2000 IEEE International Symposium on Performance Analysis of Systems and Software. ISPASS (Cat. No.00EX422), 2000, pp. 1–6. doi:10.1109/ISPASS.2000.842273.
- Ogundoyin, S.O., Kamil, I.A., 2021. Optimization techniques and applications in fog computing: An exhaustive survey. *Swarm Evolut. Comput.* 66. <https://doi.org/10.1016/j.swevo.2021.100937>. URL: <https://www.sciencedirect.com/science/article/pii/S2210650221000985> 100937.
- A. Brogi, S. Forti, C. Guerrero, I. Lera, 2020. How to place your apps in the fog: State of the art and open challenges, *Software: Practice Exp.* 50 (5) 719–740. arXiv: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/spe.2766>, doi: <https://doi.org/10.1002/spe.2766>. <https://onlinelibrary.wiley.com/doi/abs/10.1002/spe.2766>.
- Zonoozi, M., Dassanayake, P., 1997. User mobility modeling and characterization of mobility patterns. *IEEE J. Sel. Areas Commun.* 15 (7), 1239–1252. <https://doi.org/10.1109/49.622908>.
- Zhang, H., Dai, L., 2019. Mobility prediction: A survey on state-of-the-art schemes and future applications. *IEEE Access* 7, 802–822. <https://doi.org/10.1109/ACCESS.2018.2885821>.
- Noulas, A., Scellato, S., Lathia, N., Mascolo, C., 2012. Mining user mobility features for next place prediction in location-based services. In: 2012 IEEE 12th International Conference on Data Mining, pp. 1038–1043. <https://doi.org/10.1109/ICDM.2012.113>.
- M.S. Quessada, R.S. Pereira, W. Revejes, B. Sartori, E.N. Gottsfritz, D.D. Lieira, M.A. da Silva, G.P.R. Filho, R.I. Meneguette, Itsmei: An intelligent transport system for monitoring traffic and event information, *International Journal of Distributed Sensor Networks* 16 (10) (2020) 1550147720963751. arXiv: <https://doi.org/10.1177/1550147720963751>, doi:10.1177/1550147720963751. URL: <https://doi.org/10.1177/1550147720963751>.
- Secchi, S., Raad, P., Gallard, P., 2016. Linking virtual machine mobility to user mobility. *IEEE Trans. Netw. Serv. Manage.* 13 (4), 927–940. <https://doi.org/10.1109/TNSM.2016.2592241>.
- Bittencourt, L.F., Diaz-Montes, J., Buyya, R., Rana, O.F., Parashar, M., 2017. Mobility-aware application scheduling in fog computing. *IEEE Cloud Comput.* 4 (2), 26–35. <https://doi.org/10.1109/MCC.2017.27>.
- Huang, Z., Wang, P., Zhang, F., Gao, J., Schich, M., 2018. A mobility network approach to identify and anticipate large crowd gatherings. *Transp. Res. Part B: Methodol.* 114, 147–170. <https://doi.org/10.1016/j.trb.2018.05.016>. URL: <https://www.sciencedirect.com/science/article/pii/S0191261517310986>.
- Hoogendoorn, S.P., Bovy, P.H.L., 2005. Pedestrian travel behavior modeling. *Networks Spatial Econ.* 5 (2), 193–216. <https://doi.org/10.1007/s11067-005-2629-y>.
- E. Meijles, M. de Bakker, P. Groote, R. Barske, Analysing hiker movement patterns using gps data: Implications for park management, *Computers, Environment and Urban Systems* 47 (2014) 44–57, progress in Movement Analysis – Experiences with Real Data. doi: 10.1016/j.compenvurbsys.2013.07.005. URL: <https://www.sciencedirect.com/science/article/pii/S0198971513000665>.
- I. Lera, T. Pérez, C. Guerrero, V.M. Eguíluz, C. Juiz, Analysing human mobility patterns of hiking activities through complex network theory, *PLOS ONE* 12 (5) (2017) 1–19. doi:10.1371/journal.pone.0177712. URL: <https://doi.org/10.1371/journal.pone.0177712>.
- M. Luca, G. Barlacchi, B. Lepri, L. Pappalardo, A survey on deep learning for human mobility, *ACM Comput. Surv.* 55 (1). doi:10.1145/3485125. URL: <https://doi.org/10.1145/3485125>.
- Barbosa-Filho, H., Barthelemy, M., Ghoshal, G., James, C., Lenormand, M., Louail, T., de Menezes, R.P., Ramasco, J.J., Simini, F., Tomasini, M., 2018. Human mobility: Models and applications. *Phys. Rep.* 734, 1–74.
- A. Mimouna, I. Alouani, A. Ben Khalifa, Y. El Hillali, A. Taleb-Ahmed, A. Menhaj, A. Ouahabi, N.E. Ben Amara, 2020. Olimp: A heterogeneous multimodal dataset for advanced environment perception, *Electronics* 9 (4). doi:10.3390/electronics9040560. URL: <https://www.mdpi.com/2079-9292/9/4/560>.
- H. Xie, E. Tanin, L. Kulik, P. Scheuermann, G. Trajcevski, M. Fanaeepour, Euler histogram tree: A spatial data structure for aggregate range queries on vehicle trajectories, in: Proceedings of the 7th ACM SIGSPATIAL International Workshop on Computational Transportation Science, IWCTS '14, Association for Computing Machinery, New York, NY, USA, 2014, p. 18–24. doi: <https://doi.org/10.1145/2674918.2674921>. URL: <https://doi.org/10.1145/2674918.2674921>.
- Y. Xu, S.-L. Shaw, Z. Zhao, L. Yin, Z. Fang, Q. Li, Understanding aggregate human mobility patterns using passive mobile phone location data – a home-based approach, *Transportation* 42. doi: <https://doi.org/10.1007/s11116-015-9597-y>.
- L. Pappalardo, G. Barlacchi, R. Pellungrini, F. Simini, Human mobility from theory to practice: data, models and applications, in: Companion Proceedings of The 2019 World Wide Web Conference, WWW '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 1311–1312. doi:10.1145/3308560.3320099. URL: <https://doi.org/10.1145/3308560.3320099>.



- V. Kulkarni, B. Garbinato, 20 years of mobility modeling & prediction: Trends, shortcomings & perspectives, in: Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, SIGSPATIAL '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 492–495. doi: <https://doi.org/10.1145/3347146.3359110>. URL: <https://doi.org/10.1145/3347146.3359110>.
- S.M. King, F. Nawab, K. Obraczka, A survey of open source user activity traces with applications to user mobility characterization and modeling (2021). arXiv:2110.06382.
- Wang, J., Kong, X., Xia, F., Sun, L., 2019. Urban human mobility: Data-driven modeling and prediction. SIGKDD Explor. Newsl. 21 (1), 1–19. <https://doi.org/10.1145/3331651.3331653>.
- Solmaz, G., Turgut, D., 2019. A survey of human mobility models. IEEE Access 7, 125711–125731. <https://doi.org/10.1109/ACCESS.2019.2939203>.
- Thornton, F., McNamara, K.E., Farbotko, C., Dun, O., Ransan-Cooper, H., Chevalier, E., Lkhagvasuren, P., 2018. Human mobility and environmental change: a survey of perceptions and policy direction. *Popul. Environ.* 40, 239–256.
- A. Hess, K.A. Hummel, W.N. Gansterer, G. Haring, Data-driven human mobility modeling: A survey and engineering guidance for mobile networking, *ACM Comput. Surv.* 48 (3). doi: <https://doi.org/10.1145/2840722>. URL: <https://doi.org/10.1145/2840722>.
- Becker, R., Cáceres, R., Hanson, K., Isaacman, S., Loh, J.M., Martonosi, M., Rowland, J., Urbaneck, S., Varshavsky, A., Volinsky, C., 2013. Human mobility characterization from cellular network data. *Commun. ACM* 56 (1), 74–82.
- T.S. Azevedo, R.L. Bezerra, C.A.V. Campos, L.F.M. de Moraes, An analysis of human mobility using real traces, in: 2009 IEEE Wireless Communications and Networking Conference, 2009, pp. 1–6. doi: <https://doi.org/10.1109/WCNC.2009.4917569>.
- T. Wu, R.M. Rustamov, C. Goodall, Distributed learning of human mobility patterns from cellular network data, in: 2017 51st Annual Conference on Information Sciences and Systems (CISS), 2017, pp. 1–6. doi: <https://doi.org/10.1109/CISS.2017.7926085>.
- Thuillier, E., Moalic, L., Lamrous, S., Caminada, A., 2018. Clustering weekly patterns of human mobility through mobile phone data. *IEEE Trans. Mob. Comput.* 17 (4), 817–830. <https://doi.org/10.1109/TMC.2017.2742953>.
- Toch, E., Lerner, B., Ben Zion, E., Ben-Gal, I., 2019. Analyzing large-scale human mobility data: a survey of machine learning methods and applications. *Knowl. Inform. Syst.* 58. <https://doi.org/10.1007/s10115-018-1186-x>.
- Traunmueller, M.W., Johnson, N., Malik, A., Kontokosta, C.E., 2018. Digital footprints: Using wifi probe and locational data to analyze human mobility trajectories in cities. *Comput. Environ. Urban Syst.* 72, 4–12. <https://doi.org/10.1016/j.compenvurbysys.2018.07.006>. URL: <https://www.sciencedirect.com/science/article/pii/S0198971517305914>.
- Uras, M., Cossu, R., Atzori, L., 2019. Pma: a solution for people mobility monitoring and analysis based on wifi probes. In: 2019 4th International Conference on Smart and Sustainable Technologies (SpliTech), pp. 1–6. <https://doi.org/10.23919/SpliTech.2019.8783040>.
- Oliveira, L., Schneider, D., De Souza, J., Shen, W., 2019. Mobile device detection through wifi probe request analysis. *IEEE Access* 7, 98579–98588. <https://doi.org/10.1109/ACCESS.2019.2925406>.
- C. Balzotti, A. Bragagnini, M. Briani, E. Cristiani, Understanding human mobility flows from aggregated mobile phone data, *IFAC-PapersOnLine* 51 (9) (2018) 25–30, 15th IFAC Symposium on Control in Transportation Systems CTS 2018. doi: <https://doi.org/10.1016/j.ifacol.2018.07.005>. URL: <https://www.sciencedirect.com/science/article/pii/S2405896318307213>.
- S. Hoteit, G. Chen, A.C. Viana, M.C. Fiore, Spatio-Temporal Completion of Call Detail Records for Human Mobility Analysis, in: Rencontres Francophones sur la Conception de Protocoles, l'Évaluation de Performance et l'Expérimentation des Réseaux de Communication, Quiberon, France, 2017. URL: <https://hal.archives-ouvertes.fr/hal-01516717>.
- W. Gao, G. Cao, Fine-grained mobility characterization: Steady and transient state behaviors, in: Proceedings of the Eleventh ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc '10, Association for Computing Machinery, New York, NY, USA, 2010, p. 61–70. doi:10.1145/1860093.1860103. URL: <https://doi.org/10.1145/1860093.1860103>.
- D. Ashbrook, T. Starner, Learning significant locations and predicting user movement with gps, in: Proceedings. Sixth International Symposium on Wearable Computers, 2002, pp. 101–108. doi: <https://doi.org/10.1109/ISWC.2002.1167224>.
- Y. Chon, H. Shin, E. Talipov, H. Cha, Evaluating mobility models for temporal prediction with high-granularity mobility data, in: 2012 IEEE International Conference on Pervasive Computing and Communications, 2012, pp. 206–212. doi:10.1109/PerCom.2012.6199868.
- W. Mathew, R. Raposo, B. Martins, Predicting future locations with hidden markov models, in: Proceedings of the 2012 ACM Conference on Ubiquitous Computing, UbiComp '12, Association for Computing Machinery, New York, NY, USA, 2012, p. 911–918. doi:10.1145/2370216.2370421. URL: <https://doi.org/10.1145/2370216.2370421>.
- A. Asahara, K. Maruyama, A. Sato, K. Seto, Pedestrian-movement prediction based on mixed markov-chain model, in: Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, GIS '11, Association for Computing Machinery, New York, NY, USA, 2011, p. 25–33. doi:10.1145/2093973.2093979. URL: <https://doi.org/10.1145/2093973.2093979>.
- S. Gambs, M.-O. Killijian, M.N. n. del Prado Cortez, Next place prediction using mobility markov chains, in: Proceedings of the First Workshop on Measurement, Privacy, and Mobility, MPM '12, Association for Computing Machinery, New York, NY, USA, 2012. doi:10.1145/2181196.2181199. URL: <https://doi.org/10.1145/2181196.2181199>.
- N.A. Amirrudin, S.H.S. Ariffin, N.N.N.A. Malik, N.E. Ghazali, User's mobility history-based mobility prediction in lte femtocells network, in: 2013 IEEE International RF and Microwave Conference (RFM), 2013, pp. 105–110. doi:10.1109/RFM.2013.6757228.
- Yan, X.-Y., Han, X.-P., Wang, B.-H., Zhou, T., 2013. Diversity of individual mobility patterns and emergence of aggregated scaling laws. *Sci. Rep.* 3 (1), 2678. <https://doi.org/10.1038/srep02678>.
- Yu, S.-Z., Kobayashi, H., 2003. A hidden semi-markov model with missing data and multiple observation sequences for mobility tracking. *Signal Process.* 83 (2), 235–250. [https://doi.org/10.1016/S0165-1684\(02\)00378-X](https://doi.org/10.1016/S0165-1684(02)00378-X). URL: <https://www.sciencedirect.com/science/article/pii/S016516840200378X>.
- W. Zhu, C. Zhang, S. Yao, X. Gao, J. Han, A spherical hidden markov model for semantics-rich human mobility modeling, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 32, 2018.
- Fülöp, P., Imre, S., Szabó, S., Szálka, T., 2009. Accurate mobility modeling and location prediction based on pattern analysis of handover series in mobile networks. *Mobile Inform. Syst.* 5 (3), 255–289.
- Xie, J., Wan, Y., Kim, J.H., Fu, S., Namuduri, K., 2013. A survey and analysis of mobility models for airborne networks. *IEEE Commun. Surveys Tutor.* 16 (3), 1221–1238.
- Menascé, D.A., Virgilio, A.F.A., 2000. *Scaling for E Business: Technologies, Models, Performance, and Capacity Planning*. Prentice Hall PTR, USA.
- C. Kurz, C. Guerrero, G. Haring, Extending tcp-w to allow for fine grained workload specification, *WOSP '05*, Association for Computing Machinery, New York, NY, USA, 2005, p. 167–174. doi:10.1145/1071021.1071039. URL: <https://doi.org/10.1145/1071021.1071039>.
- Keramat Jahromi, K., Zignani, M., Gaito, S., Rossi, G.P., 2016. Simulating human mobility patterns in urban areas. *Simul. Model. Pract. Theory* 62, 137–156. <https://doi.org/10.1016/j.simpat.2015.12.002>. URL: <https://www.sciencedirect.com/science/article/pii/S1569190X15001732>.
- A. Hardy, On the number of clusters, *Computational Statistics and Data Analysis* 23 (1) (1996) 83–96, classification. doi: 10.1016/S0167-9473(96)00022-9. URL: <https://www.sciencedirect.com/science/article/pii/S0167947396000229>.
- O.A. Abbas, Comparisons between data clustering algorithms, *Int. Arab J. Inform. Technol. (IAJIT)* 5 (3).
- C. Gouin-Vallerand, S. Rousseau, An indoor navigation platform for seeking internet of things devices in large indoor environment, *GoodTechs '19*, Association for Computing Machinery, New York, NY, USA, 2019, p. 108–113. doi:10.1145/3342428.3342652. <https://doi.org/10.1145/3342428.3342652>.
- Syakur, M.A., Khotimah, B.K., Rochman, E.M.S., Satoto, B.D., 2018. Integration k-means clustering method and elbow method for identification of the best customer profile cluster. *IOP Conf. Ser.: Mater. Sci. Eng.* 336. <https://doi.org/10.1088/1757-899x/336/1/012017>.
- H. Gupta, A. Vahid Dastjerdi, S.K. Ghosh, R. Buyya, ifogsim: A toolkit for modeling and simulation of resource management techniques in the internet of things, edge and fog computing environments, *Software: Practice Exp.* 47 (9) (2017) 1275–1296. arXiv: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/spe.2509>, doi: <https://doi.org/10.1002/spe.2509>. <https://onlinelibrary.wiley.com/doi/abs/10.1002/spe.2509>.
- Lera, I., Guerrero, C., Juiz, C., 2019. Yafs: A simulator for iot scenarios in fog computing. *IEEE Access* 7, 91745–91758. <https://doi.org/10.1109/ACCESS.2019.2927895>.