A Mobility-Based Deployment Strategy for Edge Data Centers

Michele Girolami[a], Piergiorgio Vitello[b], Andrea Capponi[b], Claudio Fiandrino[c], Luca Foschini[d], Paolo Bellavista[d]

[a] *ISTI-CNR, Institute of Information Science and Technologies, Pisa, Italy*
[b] *University of Luxembourg, Luxembourg*
[c] *IMDEA Networks Institute, Madrid, Spain*
[d] *Dipartimento di Informatica: Scienza e Ingegneria University of Bologna, Bologna, Italy*

Abstract

The main objective of Multi-access Edge Computing (MEC) is to bring computational capabilities at the edge of the network to better support low-latency applications. Such capabilities are typically offered by Edge Data Centers (EDC). The MEC paradigm is not tied to a single radio technology, rather it embraces both cellular and other radio access technologies such as WiFi. Distributed intelligence at the edge for AI purposes requires careful spatial planning of computing and storage resources. The problem of EDC deployment in urban environments is challenging and, to the best of our knowledge, it has been explored only for cellular connectivity so far. In this paper, we study the possibility of deploying EDC without analysing the expected data traffic load of the cellular network, a kind of information rarely shared by network operators. To this purpose, we propose in this work CLUB, CLUstering-Based strategy tailored on the analysis of urban mobility. We analyze two experimental mobility data sets, and we analyze some mobility features in order to characterize their properties. Finally, we compare the performance of CLUB against state-of-the-art techniques in terms of the outage probability, namely the probability an EDC is not able to serve a request. Our results show that the CLUB strategy is always comparable with respect to our benchmarks, but without using any information related to network traffic.

Keywords: Edge Data Center; Multi-Access Edge Computing; Mobility; Mobile CrowdSensing

1. Introduction

The fifth generation (5G) mobile networks have significantly changed the way mobile wireless is built. 5G networks rely on Software-Defined Networking (SDN) and Network Function Virtualization (NFV) to support a full class of diverse services, including machine-to-machine and ultra low latency traffic. Thanks to SDN and NFV paradigm, radio access and core functions are virtualized and executed in EDCs in accordance to the MEC principle. MEC is standardized by the European Telecommunications Standards Institute (ETSI) [1] and aims at providing computing services closer to the end user [2]. Thus, it finds applicability in scenarios where locality and low-latency are essential [3]. MEC is not tied to a single radio technology, but embraces cellular and other radio access technologies such as WiFi. Furthermore, MEC is agnostic to the evolution of the mobile network itself. i.e., it can be deployed in 4G, or 5G networks.

Email addresses: `@isti.cnr.it` (Michele Girolami), `@uni.lu` (Piergiorgio Vitello), `@uni.lu` (Andrea Capponi), `@imdea.org` (Claudio Fiandrino), `@unibo.it` (Luca Foschini), `@unibo.it` (Paolo Bellavista)

EDCs are deployed close to the wireless element to provide radio coverage, e.g., base stations (BSs) inside an infrastructure owned by a mobile network operator (MNO) [4] or WiFi access points (APs). Indeed, let us stress that, while the literature on EDC focuses mostly on cellular-enabled edge nodes, also WiFi-enabled edge nodes have a strong potential and will be beneficial to a number of applications by bringing together high-bandwidth hotspots with processing and storage capabilities. MCS and city sensing are good examples of research areas where these WiFi edge nodes could act as data collection points for mobile nodes roaming across, by potentially executing also some filtering/processing operations over collected data [5]. Likewise, there are several examples of WiFi-enabled public spaces that would benefit from caching of video/applications contents to be delivered to local mobile nodes therein. More in general, focusing on both cellular and WiFi based deployments, the edge provides.

The edge provides computing functionalities that enable resource-constrained mobile devices to prolong battery lifetime [6] while enhancing and augmenting the performance of mobile applications [7]. This is key to supporting distributed intelligence at the edge. To date, edge computing research has mainly focused on resource management and allocation [8], trading power consumption and communication delays [9, 10]. Thus, it finds applicability in scenarios where locality and low-latency are essential [11].

Seminal works have mainly focused on the definition of architectural design principles [12, 13]. Emulation platforms for research in this area have only started to appear recently [14, 15, 16, 17], and little attention has been paid to the problem of resource deployment.

EDC deployment is a particularly interesting and challenging problem in the context of smart cities. Many factors influence citizens' mobility, including trip purposes (e.g. home-work commuting) and geographically imposed restrictions (e.g., temporary closed roads). Within a city, urban dynamics regulate the inter-dependency of land use and citizens'[1] movements [18], i.e. the locations they visit that determine mobility patterns. In the context of transportation research, such patterns are exploited to regulate traffic congestion, route planning, public safety and to allocate shared resources (e.g., bikes, scooters, taxis) [19]. These complex phenomena determine the characteristics of mobile data traffic [20, 21]. Recent studies identified a strong correlation between the urbanization tissue (i.e. land use) and average mobile data traffic volume per-user [22] as well as correlation between commuting patterns, city block structures and mobile cellular access data [23]. Although different mobile services exhibit different temporal behaviors, the resulting spatial patterns are uniform [24]. In particular, the amount of traffic flowing across BSs is key to define the computational demands of EDCs.

In this paper, we bring the research around the problem of EDCs deployment in smart cities one step forward. In our previous work, we focused on EDC deployment by assuming that the computing demand was entirely generated through requests carried by the mobile network [25]. We now generalize the problem by assuming to exploit WiFi connectivity too. To date, city-wide WiFi connectivity can be practically expected and in some cases, such as Luxembourg, is reality[2]. Mobile network operators have interest in deploying WiFi hotspots to support traffic offloading from cellular network [26, 27]. We retain as a fundamental assumption that EDCs can be deployed only at current BSs/APs sites, to re-use already deployed infrastructure (e.g., power supply, cabinets on roofs). By assuming that any technology can provide interconnection between UEs and EDCs, our methodology of EDC deployment is ready to serve *virtual edge operators*, i.e., actors in the market that are not owners of the infrastructure like mobile network operators.

In a nutshell, in this paper we propose CLUB, CLUstering-Based strategy for EDC deployment

[1] In the rest of the paper we will use the words *user*, *mobile user*, and *citizen* interchangeably.

[2] See: https://www.citywifi.lu/en/hotspot/

that is technology agnostic and deploys EDCs on the sole basis of user mobility aspects. More specifically, the CLUB considers the users' trajectories to identify highly-visited locations by adopting a spatial clustering approach. In turn, CLUB identifies the clusters' centroids and it performs a Voronoi-based tessellation to partition the region of interest in different sub-regions. Finally, CLUB selects for each sub-region a target location for the deployment of a set of EDCs. The proposed strategy only exploits information obtained from users' trajectories without considering any information related to the data traffic produced by users. This aspect is important as it greatly simplifies the CLUB-based deployment with respect to other approaches based on information extracted from network traffic. We benchmark the new strategy with state-of-the-art techniques [28], i.e., Distributed Deployment Algorithm (DDA) and Mobility-aware deployment algorithm (MDA). DDA deploys EDCs in such a way that they correspond to the centroid of a cluster composed of a set of BSs that all share a similar distance. In order to compare the implemented deployment strategies, we consider two mobility data sets, namely P-Mob based on trajectories extracted from the ParticipAct project [29, 30] and GPT-Mob based on information provided by the Google Popular Time Cloud service. We first analyze the two data sets in terms of mobility features and then we compare the resulting outage probability of CLUB against MDA and DDA.

Our results confirm that the CLUB deployment outperforms deployments like DDA. Furthermore, it provides slightly higher outages than MDA, while retaining the benefit of relying on a simpler mobility model and its wider applicability to all those scenarios where it is not possible to assume we can monitor network traffic. The paper is organized as follows: Section 2 surveys existing deployment strategies, Section 3 details the design of CLUB and MDA/DDA benchmark solutions and Section 4 describes the experimental session we carry out. Section 5 concludes the paper.

2. Background and Related Work

MEC allows resource-constrained mobile devices to offload computational workloads to nearby EDCs. Let us anticipate that in this paper we are not interested in the details of fog/edge architectures and protocols, for which we refer the interested reader to existing surveys [13, 31]. Differently from some recent works such as [32], [33] that delve into the technicalities of network-specific functions that can benefit from processing in EDCs in our work, we rather focus on the perspective of end-user applications. These works operate in cloud radio access network scenarios, where the baseband processing is outsourced from BS and moved in the cloud. Furthermore, we do not delve into technology, e.g. the specifications 3GPP enforced for the 5G core network and recent developments like network slicing [34]. Our objective is to deal with city-scale user mobility and exploit urban dynamics to deploy EDC devoted to the processing of end-user applications. In the following, we report the very few efforts found in the literature that share a similar approach with ours.

The paper [35] is seminal work that explored the problem of EDC deployment in smart cities by assessing the feasibility of leveraging three different infrastructures, i.e., cellular base stations, routers, and street lamps, and analyzing the potential city coverage if only a subset of these elements were upgraded to provide EDC capabilities. Other works that touched upon EDC deployment are [36, 37] and [38]. Unlike our contribution, both works disregard user mobility, which is now recognized to be a fundamental factor in smart cities because it influences the workload and computational demand of edge resources. Specifically, [36] proposes an optimal cloudlet placement and user allocation in Wireless Metropolitan Area Networks. By contrast, the objective of [37] is to identify the optimal placement of points of presence in the operator networks. In our approach, we target an EDC deployment that is only close to the access network, while [37] focuses on deploying EDCs in a layered architecture, i.e. both at access, aggregation, and core network. Unlike the above works, in [38] the authors aim at addressing the problem of assigning access points (APs) to EDC by considering as key metric the routing cost and cost of moving VMs. Specifically, the traffic demands that APs witness changes over time and with limited computing and storage capacity at EDC side, the optimal assignment might not always be the one with minimum latency between APs and EDCs. Unlike our contribution that is agnostic to the virtualization infrastructure, [38] focuses on virtual machines thereby neglecting the specifics of containerization that is the key technology for cloud-native next generation networks [39]. Finally, unlike the body of work in computation offloading, in the paper [40] the authors propose Comp-HO, an algorithm specifically designed for augmented reality in MEC environments that optimizes the joint problem signal strength and computational load so to minimize access delays and congestion at EDCs.

The closest effort to this work is our previous conference paper [25], where we proposed and compared two heuristics for EDC deployment, namely distributed deployment algorithm (DDA) and mobility-aware deployment algorithm (MDA). The objective of [25] however is to expose how by taking into account user mobility yields better deployment strategies than mobility unaware solutions. The shortcoming of such work is that it solely focuses on a single radio technology, i.e., the mobile cellular network. Both DDA and MDA assign a cost to the link that interconnects a BS and the EDC and a k-medoids algorithm is used to find the optimal BS where the new EDC is deployed. While in DDA the cost is based on geographical proximity (i.e., the objective is to ensure short links with low latency), in MDA the cost is weighted by mobility, i.e., the optimal BS for the deployment is selected by considering how many computing requests the neighboring BS have along time. With an iterative approach, MDA refines the location of new EDCs so that in any moment in time the total computing demand for each EDC is close to the average each EDC should have to minimize outages, i.e., the probability that a request can not be satisfied.

3. Edge Deployment Strategies

We now describe our reference scenario and the strategies we implemented to deploy edges in urban environments. In particular, we detail in Section 3.1 a possible MEC-based architecture, and we clarify the kinds of services that the edge nodes can locally provide. In particular, we state the problem we address, namely an efficient strategy to deploying edge nodes. Afterwards Sections 3.2 and 3.3 describe two alternative strategies for the edge deployment, namely the Clustering-Based strategy and DDA.

3.1. The Reference Scenario of the MEC Deployment

Our reference architecture is characterized by an urban environment densely populated. People visiting such environment are equipped with several types of smart devices connected to a broadband link (e.g. LTE, 5G/6G) or connected to a WiFi network.

Network operators provide to the end-users several kinds of services, ranging from traditional delay-tolerant applications, such as email or instant messaging, to low-latency services, such as gaming, video offloading and augmented reality. The kind of service required and the amount of devices simultaneously requiring such service, highly affect the overall network performance. As a general idea, low-latency services are higher demanding with respect to browsing a web

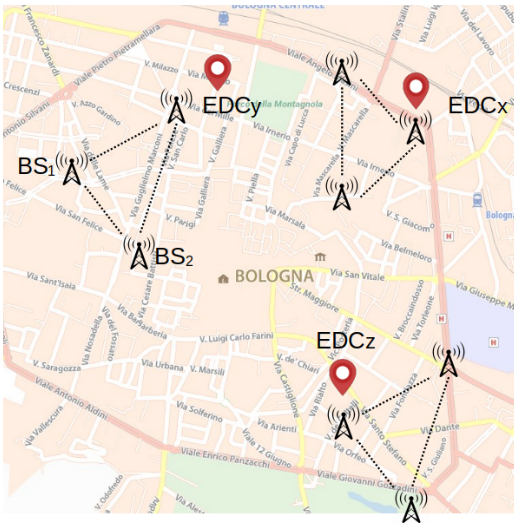


Figure 1: A graphical representation of clusters of BSs. Each cluster is proxied by an EDC, namely EDC_x, EDC_y and EDC_z. EDCs act as a local service provider for devices roaming in the nearby of the cluster (Map data copyrighted OpenStreetMap contributors and available from https://www.openstreetmap.org).

page or checking emails. As a consequence, a balance between the service's SLA and the outage probability has to be found. We model the behaviour of the end-users with the concept of *smart device activity* as the traffic generated by mobile devices. This depends on several factors, such as the mobile application used by end-users, its computational demand, and the usage duration. In this work, we describe the smart device activity. In this case, we set a fixed percentage on the total citizen walking time, to randomly generate the usage of a certain app or the period of not performing any activities with the smart device. Future steps will be to associate a specific user activity with more realistic corresponding app, such as video streaming when waiting a bus or augmented reality while walking.

We observe that, to better support the heterogeneity of the services we mentioned, the traditional cloud computing model is nowadays phased out. In particular, we explore the impact of migrating services from the Cloud to the edge of the network, so that to reduce the latency and to move the computation close to the place where services are accessed. Such migration is feasible if we extend a traditional network infrastructure with MEC architecture.

Traditionally, network operators provide connectivity to customers with a number of BSs. BS are deployed according to specific strategies. We assume that some of the BSs can also be provisioned with an EDC, a computational unit able to provide extra-services to nearby devices.

More specifically, we consider that an EDC acts as service provider for a set of BSs. Therefore, all the service requests coming from devices connected to $BS_1 \cdots BS_k$ are served by EDC_x as proxy for k BSs. We report in Figure 1 an example of our reference architecture, in which we show EDC_y proxying all the requests from devices connected to BS_1 to BS_3.

Assigning the EDCs to the BSs deployed in the environment is not an easy task being the underlying optimization problem NP-HARD (it can be mapped to the Quadratic Assignment Problem [10]). In fact, an inefficient deployment might incur in EDCs overbooked or, conversely, underused. This is the case of EDC deployed in highly populated areas where a high number of service's requests have to be served simultaneously.

We describe in Section 3.2 and 3.3 two alternative strategies for deploying EDCs in urban areas knowing in advance the location of the BSs. The two strategies differ with respect to the amount of context-information exploited during the deployment phase of the EDCs.

3.2. The Clustering-Based Strategy

The CLUstering-Based (CLUB) strategy we propose in this work implements a mobility-based deployment algorithm. CLUB does not exploit any information concerning the traffic generated by mobile devices, neither the location of the BSs, rather CLUB only analyzes the user mobility with a unsupervised clustering technique. The idea is to avoid exploiting knowledge about the incoming/outgoing BS's traffic, as this type of information is rarely available by network operators. Differently, CLUB exploits user mobility traces, commonly available as GPS trajectories. This kind of data set reports a timestamped sequence of GPS locations (e.g. according to a reference system, such as WGS84, EPSG4326) in the form: $[timestamp, latitude, longitude, user_{ID}]$. Such traces can be aggregated and processed so that to spot high-density locations. The identified locations can be used to deploy a set of EDC *close* to those places where people generally meet.

The CLUB strategy is implemented with 3 steps, as reported in Figure 2, in particular;

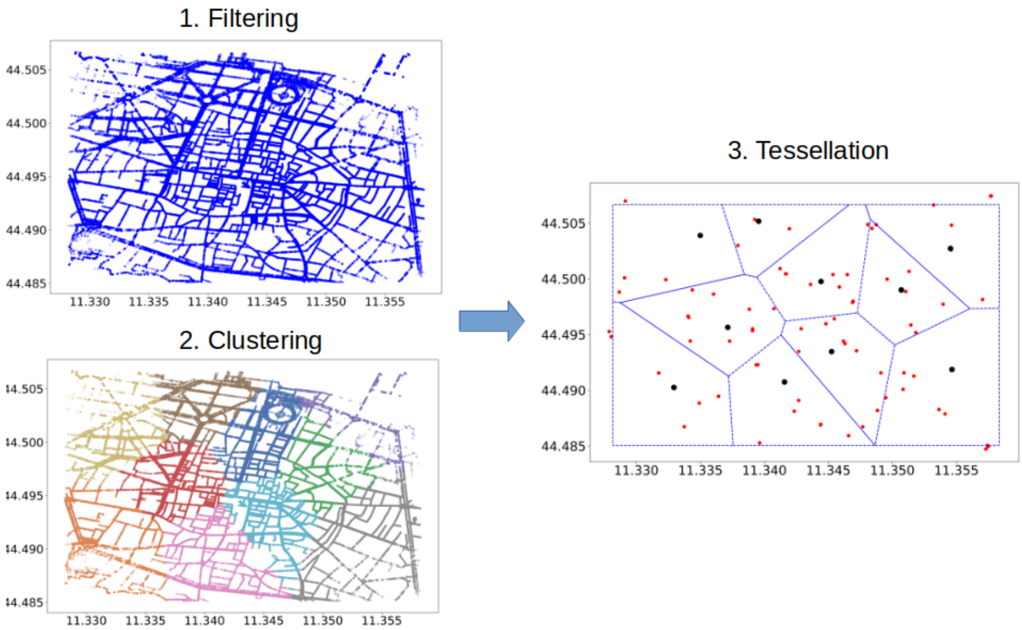


Figure 2: The processing pipeline used to deploy EDC with the CLUB strategy.

- data filtering;
- spatial clustering and identification of cluster's centroids;
- tessellation and EDC selection.

The first step consists of filtering the input data set cropping GPS points in the area of interest and restricting them to the reference time period. The output of the filtering is a set of GPS points within the bounding box B. The second step clusters the GPS points so that to identify high-density locations. For this purpose, we adopt the K-Means [41] clustering algorithm. K-Means identifies well separated clusters by minimizing the *inertia* criterion. In particular, K-Means requires to know the number of clusters to detect and then it clusters points so that to minimize the within-cluster sum-of-squares given by:

$$\sum_{i=0}^{n}(||x_i - \mu_j||^2), \tag{1}$$

where μ_j is the mean of the samples in the cluster and x_i is the i-th sample. The algorithm performs three steps:

- selecting k arbitrary points from the initial data set;
- assigning to each of the k points its nearest centroid;
- recomputing the centroids by analysing the input GPS points The centroid re-computation is interrupted when the difference between the previous locations of the centroids and the new computed locations is lower than a threshold.

In order to detect highly representative clusters, we configured K-Means so that to repeat the cluster detection with 10 iterations, this allows to shuffle the initial assignment of the k initial centroid and to obtain stable results. Moreover, we forced K-Means to initially select well separated clusters, as discussed in [42].

The result of the second step is a set of well-defined clusters, as shown with the colored map in Figure 2. We then compute the cluster's centroids, namely the cluster mid-locations. Centroids are used as input for the third step which corresponds to the tessellation. In particular, for each centroid c_i of cluster i, we identify the closest base station to c_i. The locations of the selected base stations $\hat{BS}$ are used for the deployment of k EDC. We finally associate to each of the k EDC a sub-set of base stations for which an EDC is responsible, similarly to what is implemented in the DDA strategy, see Section 3.3. This association is obtained with a Voronoi tessellation. The tessellation is obtained by using the k EDC as vertices, and it returns a set of k closed polygons. All the BS lying within polygon i are associated to the i-th EDC, as shown in Figure 2. The figure shows in red color the locations of the BS and in black color the locations of the selected EDC.

3.3. The State-of-the-art Strategies

DDA uses the k-medoids clustering algorithm to assign BSs to EDCs. While in the k-Means algorithm the centers of clusters are not necessarily input data points, the k-medoids method chooses the centroids among the input data. This is precisely what is needed to identify the location where to physically place the computing servers of the EDC. Specifically, DDA computes a cost based on the distances between the BSs that are EDC candidate locations and all the other BSs that will be assigned to it. This approach brings a significant shortcoming, i.e., under-utilization in some EDCs and overload in others. Under-utilization and overload occur when the EDC serves areas with low/high traffic and computing demand respectively. This is detrimental to outages, i.e., the probability that an EDC cannot serve a request.

MDA was developed precisely to overcome the above mentioned shortcoming of DDA. MDA captures the complex dynamics of a city (e.g., user mobility and social interactions) and use them as a weight to identify which areas of the city are potentially under-utilized or overloaded.

As DDA, MDA deploys EDCs among BSs by exploiting the k-medoids algorithm. However, unlike DDA, MDA assigns the EDCs among BSs by computing a cost based on the expected BSs' load along the day and the corresponding computational demand generated that EDCs have to sustain. MDA uses an iterative approach that computes the instantaneous computing load for each EDC. By comparing such value against the average load of all the EDCs, MDA favors allocation of incoming load to less loaded EDCs and strives to balance the load across all the EDCs thus minimizing outages.

4. Experimental Settings and Results

We detail in this section our experimental settings and obtained results. We consider 2 mobility data sets for testing the EDC deployment strategies. The data sets stress the strategies and they allow to validate the deployment strategies at different conditions. We describe in Section 4.1 the mobility data sets we use, while we report in Section 4.2 the obtained results.

4.1. Mobility Analysis with the Experimental Data Sets

We analyse 2 mobility data sets, called P-Mob (ParticipAct Mobility) and GPT-Mob (Google Popular Time Mobility). The analysis reported in this section provides a quantitative assessment of some mobility features, with the goal of characterizing the considered data sets from a mobility perspective. The reported analysis describes the nature of the data sets, in terms of geographic displacement, the followed trajectories and the trends of users in visiting the locations along with the time.

The data sets are generated by processing the information extracted from 2 real-world initiatives, namely ParticipAct [29] and Google Popular Time. P-Mob and GPT-Mob both reproduce the mobility of 100.000 users in an urban area centered in the Bologna city [3] for 24 hours. The type of mobility is pedestrian and the trajectories followed by users are generated according to a set of pre-defined origins and destinations. The distinguishing features of the data sets is how the origins of the users' trajectories are generated. We first describe how we built the data sets and then we compare some mobility features, so that to highlight similarities and differences of the data sets. [4]. It is worth to notice that P-Mob and GPT-Mob provide a complementary analysis of the EDC deployment strategies compared in this work. On the one hand, P-Mob mimics mobility from locations extracted from real-world GPS trajectories, but collected from a limited number of users. On the other hand, GPT-Mob is grounded on massive information provided by Google, hence the origins extracted are highly representative.

Concerning P-Mob, origins are extracted from the ParticipAct data set. ParticipAct is a real-world experiment whose data are collected with an Android-based mobile app. The experiment involved about 180 end-users, roaming in Emilia Romagna region for 18 months. The involved users were mainly students of the University of Bologna who accepted to join the experiment and to collect data. The number of involved users and of collected traces varies according to the time period, we refer [30, 44] to for an in-depth analysis of the ParticipAct experiment. For the purpose of this work, we analyzed 12 months of mobility, from January to December 2014. Origins of P-Mob are generated by super-imposing a mesh grid of 600 meters side and by counting the amount of GPS points fitting in each of the grid's cells. Such aggregation allowed us to identify those highly populated cells at different time hours and, in turn, to rank the preferential origins according to the cell popularity.

[3] longitude min: 11.32815834, longitude max: 11.35833366, latitude min: 44.4850610 and latitude max: 44.5066589
[4] Mobility analysis is obtained with the scikit-mobility python library [43]

Unlike P-MOB, the preferential origins of GPT-Mob are selected according to the information provided by the Google Popular Time service. Specifically, in alignment with the methodology used in our previous research [45], we fetched through the Google Cloud APIs information about the visiting patterns of several commercial activities such as: bars, restaurants and pubs. Data are extracted from the Bologna city center for a period of 2 months. The information collected allows to classify the popularity of such points of interest and, in turn, to rank them so that to select a set of preferential origins.

We first visualize the geographic distribution of points in P-Mob and GPT-Mob data sets in order to check if the two data set cover similar regions. The heatmap in Figure 3 shows that the coverage region is centered on the Bologna city center, and that visited locations are uniformly distributed in the considered bounding box. Even if the covered locations of the two data sets are similar, there exist some differences in terms of mobility features, as described in the next.

Figure 3: Geographic extension of the considered data sets. (Map data copyrighted OpenStreetMap contributors and available from https://www.openstreetmap.org).

We show in Figure 4 the time series of the number of visits aggregated on an hourly basis. The number of visits varies for the two data sets. In particular, with P-Mob visits have a stable trend for the whole day, while with the GPT-Mob we observe two peaks at 2 different time frames: 12.00 and 21.00. This behaviour reproduces a typical pattern of mobility in urban areas for working days. Figure 4 also shows the distribution of the number of visits for each of the locations detected. The two distributions are reported on the graph inset on a log scale. We observe that the two distributions reproduce a power-low trend, however the P-Mob data set reports higher probability values for the number of visits than that of the GPT-Mob data set. We now inspect the origins and destinations for the 2 data set. In particular, we detect the mobility *flows*. Flows are obtained by aggregating user's trajectories and by counting the number of trajectories from the same origin to the same destination. The higher the flow value from $a \rightarrow b$, the higher the number of users following such trajectory. The flow's origins and destinations are obtained by tessellating the area with $500m$-side tiles and by identifying the tile center as an origin or a destination.

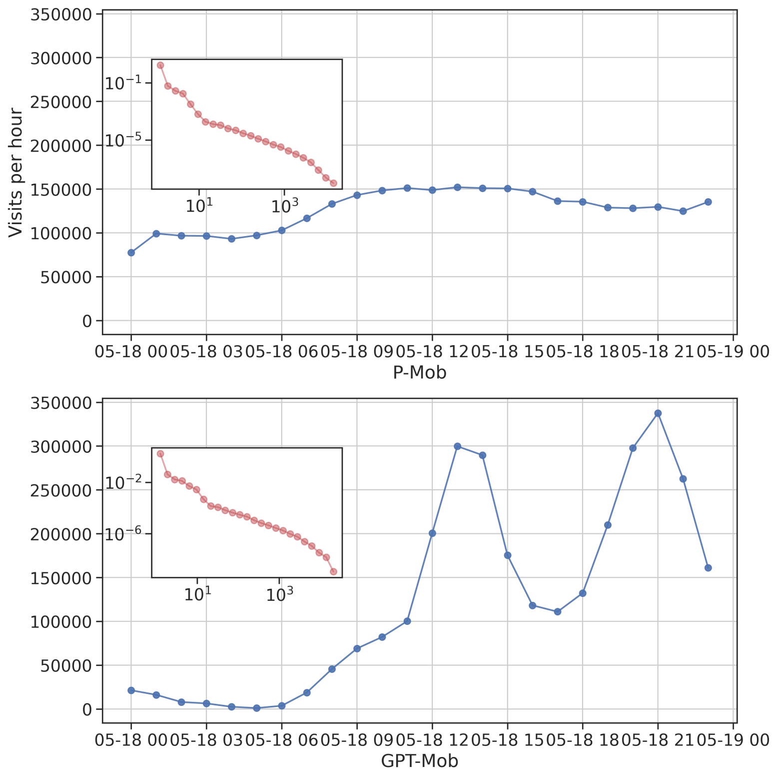
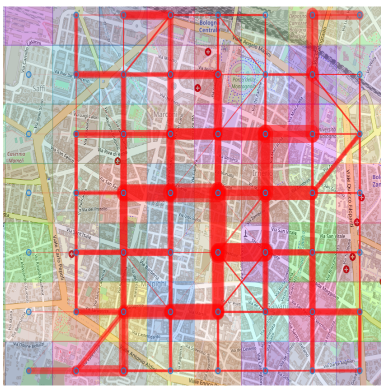
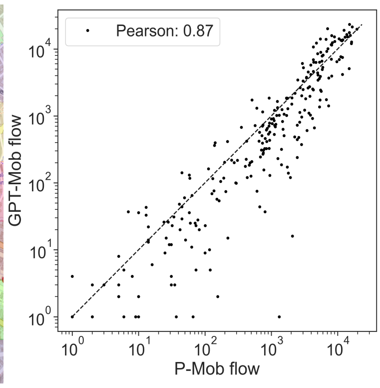


Figure 4: Visits per hour and distribution of visits per location of the experimental data sets.

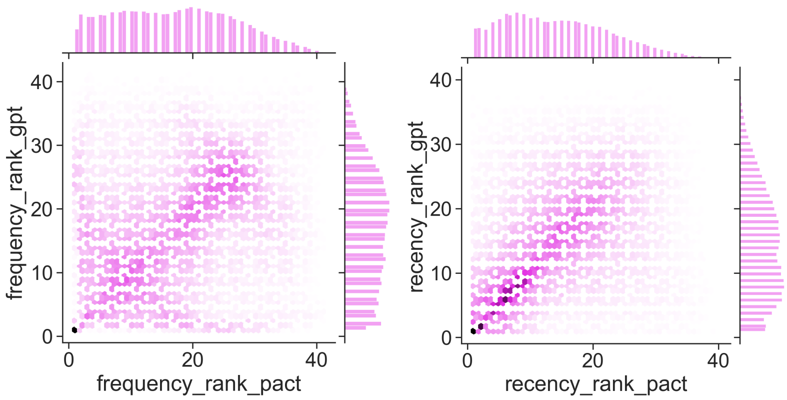


(a) Geographic representation of the P-Mob flows (Map data copyrighted OpenStreetMap contributors and available from https://www.openstreetmap.org).



(b) Correlation of the flow values

Figure 5: Flow analysis of the experimental data sets.



(a) Frequency rank.

(b) Recency rank.

Figure 6: Frequency and Recency of P-Mob and GPT-Mob data sets.

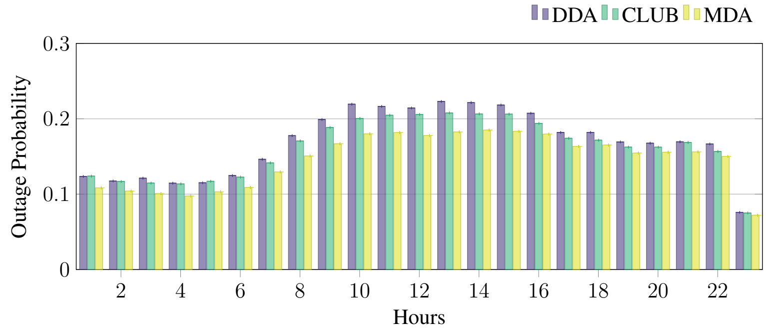
The result is a set of 264 routes for the P-Mob data set and 252 routes for GPT-Mob. We report in Figure 5a a geographic representation of the P-Mob flows. The Bologna area is split into 1241 tiles of 500 meter side, the red lines describe the flows, whose thickness is proportional to the flow value. From the figure it is possible to observe that some routes are more visited, as reported with the flows in the city center. A similar flow map is obtained for the GPT-Mob data set, as shown with the joint plot in Figure 5b. From the figure, it is possible to observe a clear correlation of the trajectory's flows for the 2 data sets, with a Pearson correlation of 0.87.

We further analyse how users of the 2 data set visit locations, to this purpose, we compute the frequency and the recency rank. The frequency rank f_l of location l measures how frequently l is visited, $f_l = 1$ means that location l is the most visited. Similarly, the recency rank r_l of location l measures how recently l is visited, $r_l = 1$ means that l is the last visited location. We compare the frequency and the recency ranks for the 2 data sets, as reported in Figure 6. Concerning the frequency rank in Figure 6a we observe a mismatch of rank in the interval $[0 - 20]$, while a more clear correlation for higher values of the frequency rank. Differently, the recency rank in Figure 6b is more tightly correlated in the 2 data sets.

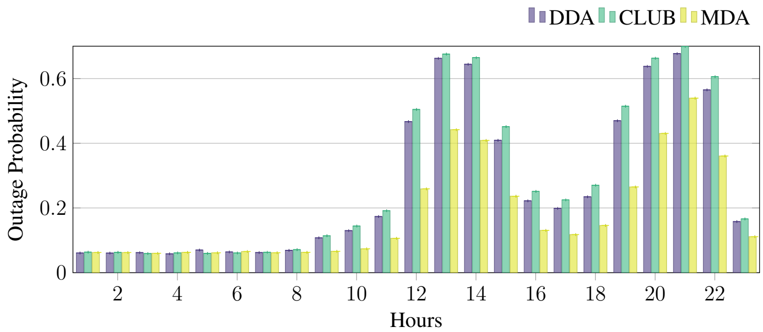
4.2. Performance of the Edge Selection Strategies

We compared the outage of our new proposed method (CLUB) with respect to the ones of the other two state-of-the-art algorithms, i.e., DDA and MDA (see Section 3.3), for the computed deployments. Figure 7 shows the per-hour outage probability in a working day for the different strategies with the number of servers fixed per EDC (set to 10) and different user mobility in the city. Figure 7(a) illustrates the outage probability for the P-Mob data set. As expected, the outage probability of MDA is the lowest and CLUB outperforms DDA during all hours. It should be noted that the difference between MDA and CLUB is minor, which confirms the goodness of our new deployment evaluated by CLUB. Indeed, CLUB requires much less information than MDA to define a deployment, since MDA requires data about the individual traffic loads of the BSs in a given area. Based on this result we can affirm that CLUB is a good compromise between outage performance and complexity.

Figure 7(b) shows the per-hour outage probability for the GPT-Mob data set. Interestingly, the results show a different behavior compared to the P-Mob. First, we can observe how the outage



(a) Outage probability based on P-Mob



(b) Outage probability based on GPT-Mob

Figure 7: Total Outage Probability in a working day.

probabilities show two clear peaks at lunch and dinner time. The reason is that the majority of the activities in the GPT data are restaurants and this provides a bias in modeling the mobility because it is strictly related to the one category of activities. As a consequence, the maximum value of outage probability reached is higher if compared to those obtained with the previous mobility model (up to 40%). Concerning the comparison between the approaches, MDA significantly outperforms the other strategies, while there is a change in the relation between CLUB and DDA, with the GPT mobility DDA having lower outage probabilities compared to CLUB, especially during peak hours.

5. Conclusions and Future Work

The advent of MEC promises to support new challenging application scenarios that require additional communications and computational capabilities at the edge of the network. EDCs are the cornerstone of these new widely diffused and complex edge deployment scenarios. While the MEC paradigm embraces all possible (wireless) communication technologies to cover the last mile, the problem of effectively deploying EDCs in smart cities scenarios has been mostly tackled for telco cellular infrastructure only, by often assuming that the whole infrastructure is under the control of the same operator. Effective EDC deployment is key to promote distributed intelligence at the network edge.

The paper presented our CLUstering-Based (CLUB) EDC deployment strategy that aims to remove those assumptions by making our solution ready-to-use in practical scenarios where MEC nodes could potentially host multiple connectivity types and where various mobile/wireless (virtual) operators cooperate for their management. In particular, obtained results, that benchmarked our solution with other existing ones in the literature, are promising and show that CLUB, by only using user mobility aggregated data, is able to compute EDC deployments that can grant outage probabilities close to the ones obtained by more informed algorithms, such as MDA, that require full knowledge of individual traffic loads. We analyze the mobility obtained from P-Mob and GPT-Mob data sets, build by using data gathered from two CrowdSensing initiatives [46], namely ParticipAct and Google Popular Time. We first analyze the mobility features of the 2 data set and then we compute the outage probability of a number of EDC selected according to CLUB, DDA and MDA. We observe that CLUB always provides comparable results with respect to DDA and MDA but at lower costs in terms of the kind of information exploited to provide an effective EDC deployment. The results show that CLUB is a good compromise between outage performance and complexity, thanks to this characteristic CLUB can be essential for EDC deployments in areas with lack of traffic loads data.

Those encouraging results are pushing us to further investigate and refine CLUB along two main ongoing research directions. On the one hand, we are investigating how to increase the effectiveness of CLUB considering also some additional user profiling information that might complement the only mobility traces, but without requiring a full control by the operator; for instance, that might include the information about the types of used mobile apps in a certain geographical area. That would allow to make more informed scheduling decisions, but also to enable new usage scenarios, such as the possibility to use CLUB to identify, in non-densely populated areas, opportunities for the network operators to evaluate the deployment of new EDCs at the edge. On the other hand, we are working to define the more complex problem of the multi-tenant management of EDCs by different and multiple virtual operators, thus considering not only the outage in terms of bandwidth, but also computing and memory requirements, as well their different (and possibly contrasting) goals, depending also on the type of delivered services.

6. References

[1] F. Giust, G. Verin, K. Antevski, J. Chou, Y. Fang, W. Featherstone, et al., MEC deployments in 4G and evolution towards 5G, ETSI White Paper (Feb 2018).

[2] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, D. Sabella, On multi-access edge computing: A survey of the emerging 5G network edge cloud architecture and orchestration, IEEE Communications Surveys Tutorials 19 (3) (2017) 1657–1681. doi:10.1109/COMST.2017.2705720.

[3] M. Satyanarayanan, The emergence of edge computing, Computer 50 (1) (2017) 30–39. doi:10.1109/MC.2017.9.

[4] C. Li, Y. Xue, J. Wang, W. Zhang, T. Li, Edge-oriented computing paradigms: A survey on architecture design and system management, ACM Comput. Surv. 51 (2) (Apr. 2018). doi:10.1145/3154815.

[5] Q. Zhu, M. Y. S. Uddin, Z. Qin, N. Venkatasubramanian, Data collection and upload under dynamicity in smart community internet-of-things deployments, Pervasive and Mobile Computing 42 (2017) 166–186. doi:https://doi.org/10.1016/j.pmcj.2017.10.003.
URL `https://www.sciencedirect.com/science/article/pii/S1574119217304868`

[6] C. Fiandrino, N. Allio, D. Kliazovich, P. Giaccone, P. Bouvry, Profiling performance of application partitioning for wearable devices in mobile cloud and fog computing, IEEE Access 7 (2019) 12156–12166. doi:10.1109/ACCESS.2019.2892508.

[7] P. Mach, Z. Becvar, Mobile edge computing: A survey on architecture and computation offloading, IEEE Communications Surveys Tutorials 19 (3) (2017) 1628–1656. doi:10.1109/COMST.2017.2682318.

[8] I. Sarrigiannis, K. Ramantas, E. Kartsakli, P.-V. Mekikis, A. Antonopoulos, C. Verikoukis, Online vnf lifecycle management in an mec-enabled 5g iot architecture, IEEE Internet of Things Journal 7 (5) (2020) 4183–4194. doi:10.1109/JIOT.2019.2944695.

[9] R. Deng, R. Lu, C. Lai, T. H. Luan, H. Liang, Optimal workload allocation in fog-cloud computing toward balanced delay and power consumption, IEEE Internet of Things Journal 3 (6) (2016) 1171–1181. doi:10.1109/JIOT.2016.2565516.

[10] J. Xu, B. Palanisamy, H. Ludwig, Q. Wang, Zenith: Utility-aware resource allocation for edge computing, in: Proc. of IEEE EDGE, 2017, pp. 47–54. doi:10.1109/IEEE.EDGE.2017.15.
[11] P.-V. Mekikis, K. Ramantas, A. Antonopoulos, E. Kartsakli, L. Sanabria-Russo, J. Serra, D. Pubill, C. Verikoukis, Nfv-enabled experimental platform for 5g tactile internet support in industrial environments, IEEE Transactions on Industrial Informatics 16 (3) (2020) 1895–1903. doi:10.1109/TII.2019.2917914.
[12] S. Sarkar, S. Chatterjee, S. Misra, Assessment of the suitability of fog computing in the context of Internet of Things, IEEE Transactions on Cloud Computing 6 (1) (2015) 46–59. doi:10.1109/TCC.2015.2485206.
[13] C. Mouradian, D. Naboulsi, S. Yangui, R. H. Glitho, M. J. Morrow, P. A. Polakos, A comprehensive survey on fog computing: State-of-the-art and research challenges, IEEE Communications Surveys Tutorials 20 (1) (2018) 416–464. doi:10.1109/COMST.2017.2771153.
[14] C. Fiandrino, A. B. Pizarro, P. J. Mateo, C. A. Ramiro, N. Ludant, J. Widmer, openLEON: An end-to-end emulation platform from the edge data center to the mobile user, Computer Communications 148 (2019) 17 – 26. doi:https://doi.org/10.1016/j.comcom.2019.08.024.
[15] Y. Zeng, M. Chao, R. Stoleru, EmuEdge: A hybrid emulator for reproducible and realistic edge computing experiments, in: Proc. of IEEE ICFC, 2019, pp. 1–6.
[16] R. Mayer, L. Graser, H. Gupta, E. Saurez, U. Ramachandran, EmuFog: Extensible and scalable emulation of large-scale fog computing infrastructures, in: Proc. IEEE Fog World Congress, 2017, pp. 1–6. doi:10.1109/FWC.2017.8368525.
[17] J. Hasenburg, M. Grambow, E. Grünewald, S.Huk, D. Bermbach, MockFog: Emulating fog computing infrastructure in the cloud, in: Proc. of IEEE ICFC, 2019, pp. 1–6.
[18] K. Jayarajah, A. Tan, A. Misra, Understanding the interdependency of land use and mobility for urban planning, in: Proc. ACM UbiComp, 2018, pp. 1079–1087. doi:10.1145/3267305.3274163.
[19] Z. Pan, Y. Liang, W. Wang, Y. Yu, Y. Zheng, J. Zhang, Urban traffic prediction from spatio-temporal data using deep meta learning, in: Proc. of ACM KDD, 2019, p. 1720–1730. doi:10.1145/3292500.3330884.
URL `https://doi.org/10.1145/3292500.3330884`
[20] J. Wang, J. Tang, Z. Xu, Y. Wang, G. Xue, X. Zhang, D. Yang, Spatiotemporal modeling and prediction in cellular networks: A big data enabled deep learning approach, in: Proc. of IEEE INFOCOM, 2017, pp. 1–9. doi:10.1109/INFOCOM.2017.8057090.
[21] X. Wang, Z. Zhou, F. Xiao, K. Xing, Z. Yang, Y. Liu, C. Peng, Spatio-temporal analysis and prediction of cellular traffic in metropolis, IEEE Transactions on Mobile Computing 18 (9) (2019) 2190–2202. doi:10.1109/TMC.2018.2870135.
[22] A. Furno, M. Fiore, R. Stanica, C. Ziemlicki, Z. Smoreda, A tale of ten cities: Characterizing signatures of mobile traffic in urban areas, IEEE Transactions on Mobile Computing 16 (10) (2017) 2682–2696. doi:10.1109/TMC.2016.2637901.
[23] H. Cao, J. Sankaranarayanan, J. Feng, Y. Li, H. Samet, Understanding metropolitan crowd mobility via mobile cellular accessing data, ACM Trans. Spatial Algorithms Syst. 5 (2) (Jul 2019). doi:10.1145/3323345.
URL `https://doi.org/10.1145/3323345`
[24] C. Marquez, M. Gramaglia, M. Fiore, A. Banchs, C. Ziemlicki, Z. Smoreda, Not all apps are created equal: Analysis of spatiotemporal heterogeneity in nationwide mobile service usage, in: Proc. ACM CoNeXT, 2017, pp. 180–186.
[25] P. Vitello, A. Capponi, C. Fiandrino, G. Cantelmo, D. Kliazovich, The impact of human mobility on edge data center deployment in urban environments, in: Proc. of IEEE GLOBECOM, 2019, pp. 1–6. doi:10.1109/GLOBECOM38437.2019.9013618.
[26] S. Andreev, O. Galinina, A. Pyattaev, K. Johnsson, Y. Koucheryavy, Analyzing assisted offloading of cellular user sessions onto d2d links in unlicensed bands, IEEE Journal on Selected Areas in Communications 33 (1) (2015) 67–80. doi:10.1109/JSAC.2014.2369616.
[27] R. Mahindra, H. Viswanathan, K. Sundaresan, M. Y. Arslan, S. Rangarajan, A practical traffic management system for integrated LTE-WiFi networks, in: Proc. of ACM MobiCom, 2014, p. 189–200. doi:10.1145/2639108.2639120.
[28] P. Vitello, A. Capponi, C. Fiandrino, G. Cantelmo, D. Kliazovich, Mobility-driven and energy-efficient deployment of edge data centers in urban environments, IEEE Transactions on Sustainable Computing (2021) 1–1doi:10.1109/TSUSC.2021.3056621.
[29] G. Cardone, A. Cirri, A. Corradi, L. Foschini, The participact mobile crowd sensing living lab: The testbed for smart cities, IEEE Communications Magazine 52 (10) (2014) 78–85. doi:10.1109/MCOM.2014.6917406.
[30] S. Chessa, M. Girolami, L. Foschini, R. Ianniello, A. Corradi, P. Bellavista, Mobile crowd sensing management with the participact living lab, Pervasive and Mobile Computing 38 (2017) 200 – 214. doi:https://doi.org/10.1016/j.pmcj.2016.09.005.
URL `http://www.sciencedirect.com/science/article/pii/S1574119216302127`
[31] Y. Mao, C. You, J. Zhang, K. Huang, K. B. Letaief, A survey on mobile edge computing: The communication perspective, IEEE Communications Surveys Tutorials 19 (4) (2017) 2322–2358.
[32] L. Chen, T. Nguyen, D. Yang, M. Nogueira, C. Wang, D. Zhang, Data-driven C-RAN optimization exploiting traffic

and mobility dynamics of mobile users, IEEE Transactions on Mobile Computing (2020) 1–1.

[33] S. Gonzalez-Diaz, A. Garcia-Saavedra, A. De La Oliva, X. Costa-Perez, R. Gazda, A. Mourad, T. Deiss, J. Mangues-Bafalluy, P. Iovanna, S. Stracca, P. Leithead, Integrating fronthaul and backhaul networks: Transport challenges and feasibility results, IEEE Transactions on Mobile Computing (2019) 1–1.

[34] V. Balasubramanian, F. Zaman, M. Aloqaily, I. A. Ridhawi, Y. Jararweh, H. B. Salameh, A mobility management architecture for seamless delivery of 5G-IoT services, in: Proc. of IEEE ICC, 2019, pp. 1–7. doi:10.1109/ICC.2019.8761658.

[35] J. Gedeon, J. Krisztinkovics, C. Meurisch, M. Stein, L. Wang, M. Mühlhäuser, A multi-cloudlet infrastructure for future smart cities: An empirical study, in: Proc. ACM EdgeSys, 2018, pp. 19–24. doi:10.1145/3213344.3213348.

[36] M. Jia, J. Cao, W. Liang, Optimal cloudlet placement and user to cloudlet allocation in wireless metropolitan area networks, IEEE Transactions on Cloud Computing 5 (4) (2017) 725–737. doi:10.1109/TCC.2015.2449834.

[37] J. Martín-Pérez, L. Cominardi, C. J. Bernardos, A. de la Oliva, A. Azcorra, Modeling mobile edge computing deployments for low latency multimedia services, IEEE Transactions on Broadcasting 65 (2) (2019) 464–474. doi:10.1109/TBC.2019.2901406.

[38] A. Ceselli, M. Fiore, M. Premoli, S. Secci, Optimized assignment patterns in mobile edge cloud networks, Computers & Operations Research 106 (2019) 246–259. doi:https://doi.org/10.1016/j.cor.2018.02.022.

[39] A. Pino, P. Khodashenas, X. Hesselbach, E. Coronado, S. Siddiqui, Validation and benchmarking of CNFs in OSM for pure cloud native applications in 5G and beyond, in: International Conference on Computer Communications and Networks (ICCCN), 2021, pp. 1–9. doi:10.1109/ICCCN52240.2021.9522356.

[40] P. Zhou, B. Finley, X. Li, S. Tarkoma, J. Kangasharju, M. Ammar, P. Hui, 5g mec computation handoff for mobile augmented reality (2021). arXiv:2101.00256.

[41] J. A. Hartigan, M. A. Wong, A k-means clustering algorithm, JSTOR: Applied Statistics 28 (1) (1979) 100–108.

[42] D. Sculley, Web-scale k-means clustering, in: Proceedings of the 19th International Conference on World Wide Web, WWW '10, Association for Computing Machinery, New York, NY, USA, 2010, p. 1177–1178. doi:10.1145/1772690.1772862.
URL `https://doi.org/10.1145/1772690.1772862`

[43] L. Pappalardo, F. Simini, G. Barlacchi, R. Pellungrini, scikit-mobility: a python library for the analysis, generation and risk assessment of mobility data (2019). arXiv:1907.07062.

[44] D. Belli, S. Chessa, L. Foschini, M. Girolami, A probabilistic model for the deployment of human-enabled edge computing in massive sensing scenarios, IEEE Internet of Things Journal 7 (3) (2020) 2421–2431. doi:10.1109/JIOT.2019.2957835.

[45] A. Capponi, P. Vitello, C. Fiandrino, G. Cantelmo, D. Kliazovich, U. Sorger, P. Bouvry, Crowdsensed data learning-driven prediction of local businesses attractiveness in smart cities, in: IEEE Symposium on Computers and Communications (ISCC), 2019, pp. 1–6. doi:10.1109/ISCC47284.2019.8969771.

[46] V. S. Dasari, B. Kantarci, M. Pouryazdan, L. Foschini, M. Girolami, Game theory in mobile crowdsensing: A comprehensive survey, Sensors 20 (7) (2020). doi:10.3390/s20072055.
URL `https://www.mdpi.com/1424-8220/20/7/2055`