2018 Third ACM/IEEE Symposium on Edge Computing

From Cell Towers to Smart Street Lamps: Placing Cloudlets on Existing Urban Infrastructures

Julien Gedeon*, Michael Stein*, Jeff Krisztinkovics*, Patrick Felka†, Katharina Keller†,
Christian Meurisch*, Lin Wang* and Max Mühlhäuser*

* Telecooperation Lab, Technische Universität Darmstadt, Germany
{gedeon, stein, krisztinkovics, meurisch, wang, max}@tk.tu-darmstadt.de
† Institute for Information Systems, Goethe-Universität Frankfurt am Main, Germany
{felka, kakeller}@wiwi.uni-frankfurt.de

***Abstract*—Cloudlets are small-scale offloading units for low-latency demands, offering a unique opportunity for emerging smart city applications such as autonomous driving or augmented reality. While previous works have investigated the general concept of cloudlets, little attention has been directed to the question of *where* to actually place cloudlets on existing infrastructure in a city. Due to cloudlets' heterogeneity in this context, their placement remains challenging.**

In this paper, we first provide a thorough analysis of a city-wide cloudlet infrastructure deployed on three types of existing infrastructures that act as wireless access points: cellular base stations, commercial off-the-shelf routers, and smart lamp posts. Based on real-world data for the access point locations in a major city and movement traces of two mobile applications, we analyze multiple coverage metrics to gain insights on the practicability of leveraging these infrastructures for a city-scale deployment of cloudlets. As a second major contribution, we propose a novel placement strategy that takes into account the heterogeneity in terms of communication ranges, resources, and costs associated with each type of cloudlet. Our strategy enables the tradeoff between deployment cost and quality of service as required for different deployment scenarios. The effectiveness of our strategy is confirmed through real-world-trace-based evaluation.

I. Introduction

The proliferation of advanced mobile applications such as those based on virtual reality (VR) or augmented reality (AR) has imposed very stringent resource requirements on the mobile devices [1]. Although those devices are becoming more powerful, their capability of handling the advanced applications is still restricted due to size and energy constraints. Cloud-based solutions [2], [3], while addressing the resource limit issue, fail to entirely fulfil the needs due to concerns over latency and traffic volume. Furthermore, they lack support for mobility [4] and context awareness [5]. These drawbacks have a severe impact for a number of upcoming applications (including AR and VR applications), where large-volume data such as video streams need to be processed in real-time [6]. Recently, a technology trend—labeled as fog computing [7]–[9] or edge computing [10]–[14]—has emerged that aims to bring storage [15] and computing [16] capabilities closer to mobile users, leveraging existing

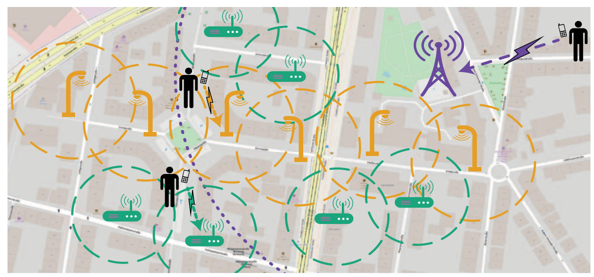


Figure 1. A multi-cloudlet urban infrastructure consisting of upgraded cell towers (*purple*), routers (*green*), and smart street lamps (*orange*).

devices to reduce latencies and core network utilization. As a key driver of this trend, cloudlets [17] have been proposed. Cloudlets are well-connected micro data centers at the edge of the network, serving as offloading targets for data and computations from resource-constrained mobile devices. Research in this direction has addressed various problems that often relate to runtime issues, e.g., offloading mechanisms [18] and programming models [19]. However, little attention has been paid to the question on where to deploy cloudlets on a city-scale.

Since establishing new infrastructures to host cloudlets for their widespread coverage is costly, we suggest to place cloudlets on three types of infrastructures present in every city: cellular base stations, commercial off-the-shelf routers, and street lamps. Mobile users can then leverage these cloudlets for offloading. This general idea is visualized in Figure 1. We furthermore believe that exploiting existing infrastructure is an important enabler for future smart city applications [20]–[23] that provide services to citizen. Example services include environmental monitoring, traffic management and optimization, emergency response, and AR games.

To show the feasibility of this approach, we first conduct an analysis of the coverage that can be achieved when only a subset of these infrastructures are upgraded to host cloudlets. For this, we use four different metrics of coverage (spatial, point, path, and time coverage) and investigate

978-1-5386-9445-9/18/$31.00 ©2018 IEEE
DOI 10.1109/SEC.2018.00021

how they affect the resulting coverage. Second, instead of randomly choosing a certain number of access points to upgrade, we turn our attention to placement strategies that aim to minimize placement costs and maximize the available offloading capabilities for users. Existing algorithms from the domain of wireless sensor networks (WSNs) that optimize coverage often do not consider heterogeneity in terms of cost and resources—something characteristic of our problem domain—and therefore cannot be applied to make reasonable placement decisions for the deployment of cloudlets. To the best of our knowledge, this paper is the first to close this gap by proposing a joint optimization of coverage and costs for the placement of heterogeneous cloudlets. For both our coverage analysis and the evaluation of placement strategies, we use extensive real-world data from Darmstadt, a major German city. The datasets contain the locations of access points in the city as well as user traces from two mobile applications.

In summary, the contributions of this paper are threefold:

- *Understanding.* We study the particular characteristics and stakeholders of a city-wide, heterogeneous cloudlet landscape on three existing urban infrastructures (namely cell towers, routers, and street lamps).
- *Coverage Analysis.* To examine the (partial) benefit of the different cloudlet-capable infrastructures, we perform a comprehensive coverage analysis on real-world data from a representative city. Given location data of access points, we analyze what degree of coverage can be achieved by using a certain percentage of all available access points. For this purpose, we define four different coverage metrics.
- *Placement Strategy.* Based on the findings of the coverage analysis, we propose `GSCORE`—a novel strategy for placing cloudlets in urban spaces that considers the infrastructural heterogeneity wrt. costs and quality of service (i.e., communication ranges and available resources). The evaluation shows that `GSCORE` outperforms the baseline strategies in different scenarios.

The remainder of this paper is organized as follows. First, we provide background information and review related work in Section II. Next, Section III introduces the considered multi-cloudlet architecture. Our real-world datasets are described in Section IV. In Section V, we perform our coverage analysis of urban cloudlets. We formulate the placement problem and propose an algorithm to place heterogeneous cloudlets in Section VI. Finally, Section VII concludes the paper and discusses future work.

II. Background and Related Work

In this section, we provide background information and review related work in the domains of cloudlets, computation offloading, coverage, and the placement of cloudlets.

A. Cloudlets and Edge Computing

Up until recently, offloading computations was mostly done through what is known as *Mobile Cloud Computing* (MCC) [2], [3], i.e., by making use of cloud computing infrastructures [24]. To counter the drawbacks of MCC, researchers have made efforts to push computations closer to the mobile end users by providing lightweight computing entities close-by. As a pioneering idea to realize this, the notion of cloudlets has first been introduced by Satyanarayanan [17] as a concept to provide small-scale data centers that can be leveraged by nearby devices. Initially based on virtual machine technology, performance considerations have since then shifted the practical implementation of cloudlets towards more lightweight virtualization technologies, such as containers [25]–[27] or unikernels [28]–[30]. Liu et al. [27] present an edge computing platform that is based on customized Docker containers instantiated on standard routers. Fesehaye et al. [31] analyze the impact of using cloudlets with regard to latency and throughput from a user perspective compared to cloud computing. Others have pointed out the benefits of using cloudlets to reduce the energy consumption [32]. Besides hosting cloudlets on dedicated infrastructures, Chen et al. [33] and Chi et al. [34] also suggest the use of ad-hoc cloudlets, i.e., cloudlets that are hosted on the mobile devices themselves and interact with other nearby devices. The concept of cloudlets has been used to realize various kinds of applications, including caching big data [35], providing cognitive assistance [6], [36], and enabling AR applications [37]. Pang et al. [38] survey the current state and future challenges of cloudlet-based mobile computing.

B. Computation Offloading

Several frameworks for offloading computations from mobile devices have been proposed. MAUI [18] is based on a profiler that decides on a method-level granularity where parts of the applications are offloaded to. The main focus of their work is on energy-awareness, aiming to maximize the lifetime of mobile devices. The authors of CloneCloud [24] introduce a partitioning mechanism that enables devices to offload parts of the execution to device clones in the cloud. Ding et al. [39] present *MADNet*, an energy-aware offloading architecture for mobile phones. A special case of devices to offload to are privately owned routers that have either been used to discover surrogates [40] or to perform computations on their own [41]. Besides offloading computations, the authors in [15] extend the notion of edge cloudlets and consider them as micro-storage units at the edge of the network. Providing offloading capabilities at the radio access network (RAN) has been investigated in [42], with the special case of so-called femtocells [43], which are less expensive to deploy and operate. Visions for offloading infrastructures also include the use of drones to host cloudlets [44]. In contrast to our work, none of the existing works have considered street lamps as locations for cloudlets.

C. Coverage

The problem of coverage has been studied extensively in the context of WSNs, as analyzed in various surveys [45]–[48]. In general, coverage describes how well an area of interest can be monitored [45], [46]. There are several scenario-specific definitions of coverage, such as sweep coverage [49] or barrier coverage [50]. In a way similar to cloudlets, participatory sensing requires volunteers to contribute. In this domain, Gedeon et al. [51] have examined the spatial and temporal coverage of moving sensors in cities, some of which are carried around by people. In our urban cloudlet scenario, coverage refers to the quality of service that can be delivered by the network. Similar to our definitions of coverage that will be introduced in Section III-B, Fan et al. [52] propose different definitions of coverage, namely area, point, and path coverage. We alter these definitions to fit the scenario of mobile users in the city that wish to perform offloading. Examining the coverage of edge cloudlets has been done by considering only one type of cloudlet [40], or focusing on temporal [53] or point coverage [54] only.

D. Cloudlet Placement

While there is abundant research on the placement of (virtualized) computing resources, both for homogeneous environments like data centers [55], [56] and in the context of cloudlets and edge computing [57]–[59], the question of where to place cloudlets on available heterogeneous infrastructures has seldom been examined. Two works [60], [61] study the placement of cloudlets in wireless metropolitan area networks (WMANs) and jointly propose solutions for the user-to-cloudlet allocation problem, but they do not consider the costs of cloudlets. Xu et al. [60] present a greedy heuristic to minimize the average access delay of mobile users to a cloudlet. Jia et al. [61] devise two algorithms to minimize the response time: *Heaviest-AP First (HAF)* and *Density-Based Clustering (DBC)*. The former places cloudlets to the access points where user workloads are the heaviest, while the latter places cloudlets according to user-dense regions. Caselli et al. [62] focus on the planning of a cloudlet network that consists of cellular base stations only. Similarly, the authors in [63] analyze a large dataset of cell tower locations in the US. Without considering the costs or computing resources, they investigate the distance reduction to data centers when cell towers of a certain category —classified according to the estimated residential population— are upgraded with micro data centers.

Yao et al. [64] investigate the cost-aware deployment of cloudlets that are heterogeneous with respect to costs and resource capacities. They adopt a greedy strategy that iteratively chooses cloudlets with minimum unit cost of resources. Compared to our model, they make assumptions that are not realistic, e.g., that there is no spatial overlap in the deployment of cloudlets and that the entire area is covered by access points. Even though the authors consider heterogeneous cloudlets, they are not linked to real-world infrastructure. In contrast, we consider three different types of infrastructure, each with specific characteristics. Bulut et al. [65], [66] have studied the deployment of WiFi access points. They do however assume that access points can freely be placed anywhere. In contrast to that, we assume that we cannot influence the placement of the access points but instead have to choose a subset of the existing ones.

Table I
CHARACTERISTICS OF ACCESS POINT TYPES

	Cellular Base Stations	**Routers**	**Street Lamps**
Density	low	high	medium
Ownership	Mobile network provider	ISP, businesses or private	Municipal
Access technology	3G/4G	WiFi	WiFi
Communication range	high	low	low-medium
Computational resources	high	low-medium	low
Costs	high	low-medium	low

III. A MULTI-CLOUDLET ENVIRONMENT

We now introduce our cloudlet environment, namely the different types of cloudlets and our definitions of coverage for the later analysis.

A. Types of Cloudlets and Stakeholders

For our urban scenario, we consider cloudlets to be hosted on three types of infrastructure: cellular base stations, routers, and street lamps. Mobile users in the vicinity of these cloudlets can then make use of them to offload data and computations. The different types of access points are heterogeneous in a number of ways. First, due to different wireless access technologies, their communication ranges vary. Second, due to the physical space available for hardware installations at the access points, the computing resources at the cloudlets vary. Lastly, we have different stakeholders that own or operate the infrastructure. This leads to different business models and hence, varying costs of using the cloudlets. We summarize the characteristics of each type of cloudlet in Table I. The use of existing infrastructure as well as future infrastructures, such as lamp posts in the context of smart cities, allows a cost-effective placement of cloudlets. Moreover, this allows a smooth transition when replacing existing access technologies with emerging ones (e.g., moving from 4G to 5G). Our scenario makes use of the heterogeneous landscape and takes advantage of it in two ways. First, our scenario builds on a realistic hardware landscape that already exists in most urban areas, which makes our approach transferable to the real world. Second, we use this heterogeneity for a targeted optimization of the placement of cloudlets. With the implementation of different cloudlet types, several stakeholders are involved, especially in the context of smart cities.

1) Cellular Base Stations: Since every major city today is entirely covered with cellular reception, cellular base stations are a good location to deploy cloudlets if we want to assure that they can reach a large number of users. Another advantage is their high reliability [67] and good interconnection with backhaul networks. However, the access latency might be subject to a high variance. Offloading to cellular base stations is commonly referred to as Mobile Edge Computing (MEC) [21], [68] and motivated by the fact that offering computing and storage resources at the extreme edge of the network is a future business opportunity for service providers and network operators. This trend is further going to be fueled by the advent of 5G networks [69] and the deployment of femtocells [43]. Additionally, at most cellular stations, there is enough physical space available to install massive computing resources in the form of server-grade hardware.

2) Routers: Next, we consider commercial off-the-shelf WiFi routers. Unsurprisingly, their density in urban areas is very high. It is important to note that these devices are either privately owned, are public access points, or belong to businesses. The latter often already offer their customers free WiFi access, while other projects promote the open sharing of ones WiFi (e.g., Freifunk[1] in Germany). We argue that going one step further—from providing network access towards computational capabilities—is a natural progression. To allow easy access, computing resources can either be located on the routers themselves or one network hop away in the local network connected to the router.

3) Street Lamps: Besides service providers, businesses, and private citizens, municipalities also have an inherent interest to enable services that lead to smarter cities. For this reason, we envision cloudlets to be placed on lamp posts. Upgrading lamp posts to host cloudlets might seem to incur a huge investment at first. However, municipalities around the world are currently in the process of updating their street lighting, mostly due to energy considerations. According to the *Humble Lamppost* project[2], 75% of lamp posts in Europe are over 25 years old and consume between 20 and 50% of a city's energy budget. Therefore, investments to upgrade lamp posts to LED-based lighting with additional functionalities, such as sensory capabilities and network connectivity, will amortize in only a few years. Consequently, a number of commercial products are already available, e.g., the SM!GHT[3] lamp by the German company *EnBW*. We argue that in view of this trend, installing additional hardware to provide computing resources is a negligible investment. From the perspective of users moving on a city street, cloudlets on lamp posts would have the advantage of a less obstructed communication range compared to routers located in buildings.

[1]https://freifunk.net (accessed: 2018-05-17)
[2]https://eu-smartcities.eu/initiatives/78/description (accessed: 2018-08-23)
[3]https://smight.com/en/ (accessed: 2018-05-17)

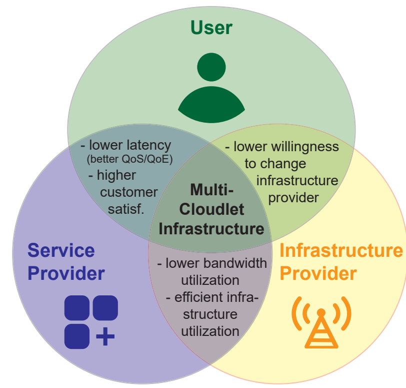
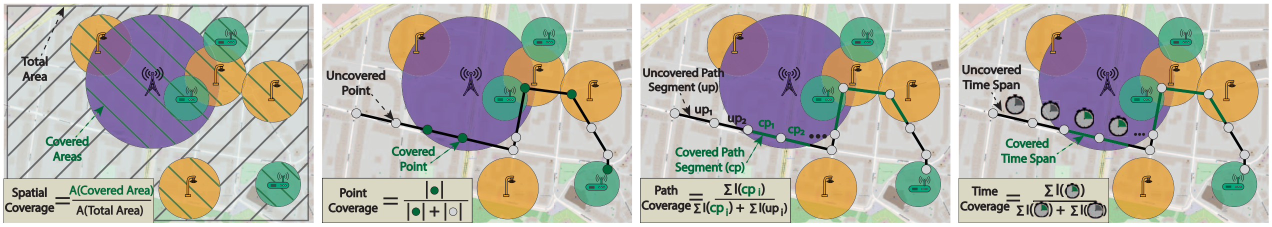


Figure 2. Stakeholders and their benefits in a multi-cloudlet scenario.

To conclude this section, the placement of cloudlets brings advantages on the side of infrastructure providers, such as a more efficient utilization of the network. To transfer these benefits to an application or service, our scenario considers the service provider as a second stakeholder. The service provider is responsible for the placement of cloudlets, which, for example, can be based on service usage. Only if a targeted placement of cloudlets takes place by the service provider (e.g., by taking into account demands and cost considerations), a significant saving effect can be realized by the infrastructure provider (e.g., lower utilization of the backhaul link). Furthermore, due to lower latency and higher throughput of the targeted cloudlet placement, higher service quality can be achieved. This leads us to the last stakeholder in our scenario, the user. Both providers are interested in providing their users with the best possible service. The infrastructure provider, such as the mobile network provider, has the role of a trailblazer providing the connection to the Internet. Thus, his focus is on factors such as network throughput and latency to provide valuable service quality to their users. In an urban cloudlet scenario, these interests cannot be assessed detached from the underlying service, which in turn highlights the role of the service provider. However, on the part of the user, targeted placement of cloudlets by the providers should result in higher service quality and user satisfaction. Figure 2 illustrates the tension between the stakeholders in our scenario.

B. Coverage Definitions

The term *coverage* is most widely used in the context of WSNs [70]. In this domain, coverage denotes how well an area of interest can be monitored by sensors. Coverage is therefore a metric for the quality of service the network can deliver. While many definitions of coverage exist, not all of them are applicable to the application domain of cloudlets. For instance, barrier coverage [50] denotes the singular detection of a target inside an area. It is not a



(a) Spatial coverage (b) Point coverage (c) Path coverage (d) Time coverage

Figure 3. Coverage metrics.

sensible metric for our application domain because we want mobile users to have a continuous connection to cloudlets and not just at a single point in time. Instead, for our analysis of urban cloudlet coverage, we consider the following four coverage metrics:

Spatial coverage: This is defined as the ratio between the union of the communication ranges of available cloudlets and the total size of the area (see Figure 3(a)). Consequently, spatial coverage gives only an indication on how well an area is covered by cloudlets and does not consider user locations.

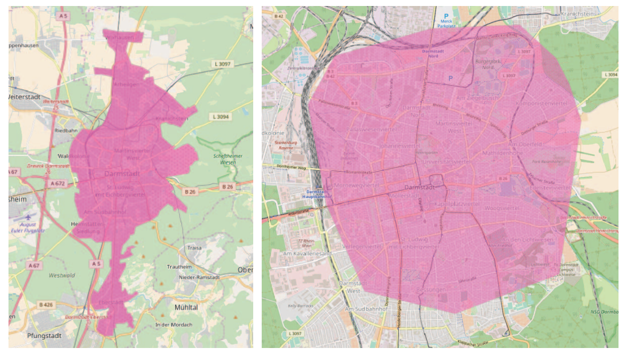
Point coverage: Point coverage indicates how many recorded location points of a mobile user are within the communication range of a cloudlet, as depicted in Figure 3(b). This coverage metric can therefore be used to model if user requests at distinct points can be served by a cloudlet.

Path coverage: Since users also move between the distinct points at which their position is recorded, path coverage takes into account the entire path length when computing the coverage ratio (see Figure 3(c)). This allows to model use cases where users need continuous connectivity to a cloudlet, e.g., when continuously processing video streams.

Time coverage: The different segments, i.e., the individual distances between two consecutive points in a user's path, might have different travel times. This metric works in a similar way as path coverage, but instead of the length of the path considers its duration (see Figure 3(d)).

IV. Datasets

We investigate the placement of cloudlets in the city of Darmstadt, Germany, a major city with a population of about 150 000. To do so, we use real-world data for both the location of access points and the traces of mobile users as described hereinafter. While the official administrative boundary of the city is depicted in Figure 4(a), we restrict our analysis in the remainder of the paper to the inner city area (spanning an area of $14.57 km^2$) as shown in Figure 4(b) because most of the access point data gathered lies within that boundary. This is especially true for the routers, which were collected by volunteers. Furthermore, the inner city area allows us to study the interplay between all three types of infrastructure, not all of which might be available with the same density in more rural areas.



(a) Administrative city boundary (b) Inner city area

Figure 4. City areas of Darmstadt.

A. Access Point Locations

In total, we collected the locations of nearly 50 000 access points throughout the city for the different types of access points. We now provide a description of how this data was obtained.

1) Cellular Base Stations: The *Bundesnetzagentur* (Federal Network Agency) is the regulating body in Germany in charge of authorizing and supervising the operation of radio installations. All transmitting stations, including cell towers, can be viewed through their website[4]. However, the website does not provide a feature to export the data. Thus, we performed a manual crawl using the network panel of the Google Chrome browser developer tool. We issued a query of all the cell towers within the city and parsed the resulting JSON data that contains their GPS locations.

2) WiFi Routers: We followed a wardriving approach to collect information about WiFi networks in the city and used this data to estimate the position of routers within the city. Using an Android application, volunteers walked around the city and collected the signals from available WiFi access points. We used the raw data from two volunteering campaigns, conducted in March 2016 and February 2018. In total, 27 participants —mostly students— were involved.

[4]http://emf3.bundesnetzagentur.de/karte/ (accessed: 2018-05-17)

Table II
NUMBER OF ACCESS POINTS COLLECTED

	Cellular Base Stations	Routers	Street Lamps
Total	205	34 699	14 331
Density per km^2	1.7	284.0	117.3
Inner City	66	31 974	5608
Density per km^2	4.5	2194.5	384.9

The positions of the access points were then estimated via multilateration from multiple measurements of the same access point's RSSI. By doing a lookup on the MAC addresses, we eliminated all manufacturers that do not produce routers. While this data might include some wrong data and uncertainty regarding the exact positions of the access points, we argue that overall this gives a reasonable estimation of the available routers to place cloudlets. More importantly, the data was collected while walking through the city and not inside buildings or private locations, therefore reflecting the usage context of a mobile user who wishes to perform opportunistic offloading.

3) Street Lamps: We obtained a database dump of the position of all street lighting in Darmstadt from *e-netz südhessen GmbH*[5], the company in charge of managing the city's electrical infrastructure. The dataset includes different types of street lighting, such as lights hung via cables over streets, but we only include fixed lamp posts for our further analysis of cloudlet coverage and placement, as they provide enough space and a safe enclosure to install additional hardware for cloudlets.

To conclude the description of the access point datasets, Table II summarizes the number and density of each access point type for the different city areas.

B. Mobility Traces

To be able to analyze the different types of coverage that take into account the user's position, i.e., point, path and time coverage (see Section III-B), we need realistic mobility traces that reflect where in the city we have demands for offloading. For our analysis, we use data from two mobile applications, Kraken.me and Ingress. Additionally, we include artificially generated mobility traces from a simulation tool. The three datasets differ with respect to the mobility patterns they represent. In addition, they feature locations both inside buildings and outside. We believe that combining them in our analysis can therefore be used to model offloading use cases for various applications. For example, Kraken.me maps the daily activities of users, i.e., a large amount of time users are at home or work, while Ingress directs users to specific locations in the city.

1) Kraken.me traces: Kraken.me [71] is a tracking framework that records users' activities and gathers data from various soft and hard sensors on mobile devices in order to provide personal assistance. During the development, a user study was conducted for several weeks using Android phones. Participants of the study were mostly students and university research staff. For our evaluations, we use a stripped dataset that only contains the timestamped positions along with a unique user ID.

[5]https://www.e-netz-suedhessen.de/ (accessed: 2018-05-17)

Table III
MOBILE APPLICATION TRACES

	Kraken.me	Ingress	CrowdSenSim
Users	205	1401	2499
Data points	437 417	520 409	431 001
Paths	11 930	47 915	44 150

2) Ingress game data: Ingress[6] is a popular mobile AR game and the predecessor of *Pokémon Go*. Players visit portals at physical locations in the city. Each player needs to visit and interact with multiple portals, which leads to a constant movement of the player in the real world. Consequently, the users' positions are recorded implicitly by the interaction at the portals. In total, there are 724 portals located in the inner city area of Darmstadt. The current state of the game and player activity is visible on the Ingress Intel Map website[7]. We built a crawler based on Python and *Selenium*, a tool that automates browsers, and requested changes in the game state every second. It is important to note that changes include the position updates from players at portals. Because the user locations are only recorded at the portals and not between, the data is more coarse-grained in terms of temporal resolution compared to the Kraken.me data. Due to the nature of the game, users are directed to the portals. However, the positions are also a good indicator for offloading demands related to other applications, since portals are often located at points of interest in the city.

3) Generated mobility traces: Lastly, to extend the number of available data points for our analysis, we artificially generated mobility traces by using *CrowdSenSim* [72], a discrete-event simulator for mobile crowd sensing. The simulator can generate user traces in urban areas where users roam around the city and randomly take turns onto streets. We set the simulation parameters such that several simulations are carried out for 7 days with 2500 users. The minimum and maximum travel times per path were set to 30 minutes and 720 minutes, respectively.

For each dataset, we performed a basic filtering of the data, such as removing data points with obviously erroneous positions or timestamps. We further defined threshold values for the minimum distance between two points (5m), the total spatial extent of a path (2000m^2), and a time threshold for the start of a new path (5 minutes). The resulting number of distinct users, data points, and paths are summarized for the inner city boundary in Table III.

[6]https://www.ingress.com/ (accessed: 2018-05-17)
[7]https://www.ingress.com/intel (accessed: 2018-05-17)

V. Coverage Analysis of Urban Cloudlets

In this section, we analyze the coverage of urban cloudlets when only a certain percentage of access points are upgraded to host those cloudlets. We perform the coverage analysis according to the four metrics we defined in Section III-B.

A. Spatial Coverage

First, we only investigate spatial coverage for the individual access point types without considering mobility traces of users. Figure 5 shows the results for cellular base stations (Figure 5(a)), routers (Figure 5(b)), and street lamps (Figure 5(c)). We assume a unit-disk model for the communication ranges and show the results for different realistic communication ranges for each type of cloudlet. For each step of 10 percent, the corresponding number of access points is randomly chosen. Besides access points located inside the inner city boundary, we also include access points whose communication ranges span across that boundary. Each experiment is run five times. While the resulting plots also display the corresponding error bars, they are very small for routers and lamps, since their communication ranges are much smaller and, thus, overlaps that impact the gain in coverage are unlikely. From the results, we can observe the general trend that a rather small fraction of upgraded access points is sufficient to provide good spatial coverage. This is especially true for routers because of their sheer number. Assuming a rather conservative communication range of 40m, already 20 percent of routers lead to almost 60 percent spatial coverage. For street lamps, the same fraction results in about 30 percent coverage. The increase in coverage for routers and street lamps is slower from a certain point on because with increasing numbers we get more spatial overlap in the communication range and, thus, less gain in overall coverage. In comparison, there are far less cell towers; they however have a much greater communication range. Figure 5(a) shows two consequences of this. First, adding more cell towers keeps increasing the overall coverage more significantly compared to routers and lamps and second, intersections in the communication ranges lead to high values in the error bars for small percentages.

We expect the overall coverage ratio to be even better with the following analyses that are based on mobility traces. Here, we will examine the coverage when combining different types of access points.

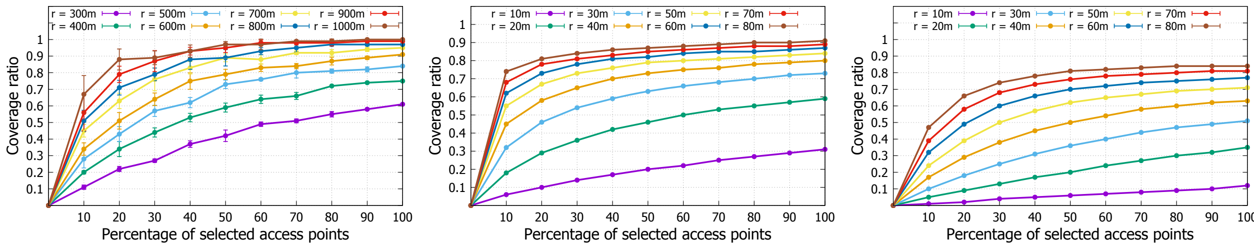
B. Point, Path, and Time Coverage

Next, we consider the mobility traces described in Section IV-B and evaluate the point, path, and time coverage (as defined in Section III-B) of the datasets. Since users are not evenly distributed in the city, this analysis allows for more realistic insights on cloudlet coverage, especially since users tend to change their location frequently. The datasets give realistic estimations on where offloading capacities will be required in the future. For instance, upcoming versions of the Ingress game might require more sophisticated processing for AR that cannot be handled by the mobile device itself. To reflect different deployment and business models, we define different scenarios with a varying number of access points of each type available. As an example, by incentivizing private individuals to provide computing capabilities at their home routers, the number of these devices that are available will increase. Similarly, network operators and municipalities are likely to have different cost constraints and willingness to upgrade their infrastructure. Subsidies might be another way to influence this. We define six different scenarios as described in Table IV with the relative and absolute number of access points per type. Assuming again a unit-disk model for the communication ranges, we select them randomly between the following ranges. For cellular base stations, the communication ranges are set between 300 and 1000 meters. Some works suggest an average communication range between 50 and 60 meters for WiFi routers [73], [74]. However, in our urban scenario, this might vary greatly (e.g., due to obstacles or different building structures); therefore, for cloudlets on routers and street lamps, we choose a range between 10 and 80 meters. Figure 6 plots the results for the three datasets. In each of the plots, we evaluate the point, path, and time coverage per scenario. The individual bars are stacked to represent the combined coverage we obtain from multiple types of access points. The stacking represents the additional coverage we gain by adding the subsequent type of access point. We assume that as the first type of access point, street lamps will be chosen, since—giving the underlying business model of municipalities providing services to their citizen—they will incur the lowest costs for users. Furthermore, because most of our location traces are not inside buildings but outside, street lamps are likely to be closest and therefore the best-connected cloudlets for users. The next part of the bar represents how much coverage routers add to points, paths, or time spans not covered by those lamps. Since our model assumes cell towers to be the most expensive type, they are used last to fill the gap that cannot be covered by other types of access points.

Table IV
Evaluation Scenarios

Scenario#	Cellular Base Stations	Routers	Street Lamps
SC1	75% (68)	10% (3224)	25% (1433)
SC2	75% (68)	25% (8060)	10% (573)
SC3	50% (45)	25% (8060)	25% (1433)
SC4	50% (45)	50% (16 120)	5% (286)
SC5	25% (22)	25% (8060)	10% (573)
SC6	25% (22)	10% (3224)	50% (2867)

From the results, we can make a number of interesting observations. Surprisingly, the variance between the different types of coverage (i.e., path, point, and time coverage) is very small. While there are variances in the datasets with respect to the distance and difference in times between the data

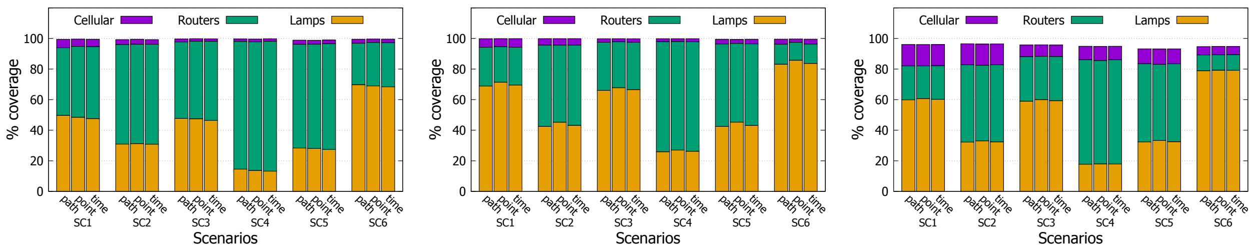


(a) Cellular base stations

(b) Routers

(c) Street lamps

Figure 5. Spatial coverage for different access point types.



(a) Kraken.me

(b) Ingress

(c) CrowdSenSim

Figure 6. Coverage analysis of path, point, and time coverage for the mobility traces.

points, this seems to average out for the whole dataset. For datasets with coarser temporal resolution, such as Ingress, we can still see a lower time coverage compared to the other metrics. Overall, this result shows that even if we only measure the coverage at single locations, on a global scale, users will most likely also have a connection to a cloudlet for most of the time along their path. We believe this provides an interesting insight for the planning of city-scale cloudlet infrastructures in the case where there is only limited data available to estimate the required demands. For instance, data protection laws might restrict the linking of entire user paths with their timestamps. Our analysis shows that regardless of what kind of data is available, each of the three coverage metrics can be used to estimate the resulting coverage for mobile users with offloading demands.

Looking at the overall coverage across the datasets, we see that the coverage is higher compared to the previous spatial coverage analysis because spatial coverage also includes areas that are less likely to be populated by people. This is validated by the fact that for the CrowdSenSim dataset (Figure 6(c)), which are generated movement traces rather than real ones, the overall coverage is lower compared to Kraken.me (Figure 6(a)) and Ingress (Figure 6(b)). In general, when combining the different rather small percentages for the individual types, we get high overall coverage ratios. For instance, for the first scenario (SC1) of the Kraken dataset, selecting only 25% of street lamps leads to almost 50% of coverage for that type alone. Adding routers, which are present in much greater number, the coverage surpasses 90%. This result holds true across all investigated datasets and coverage metrics. Obviously, the percentage of lamps that is selected first has the highest impact on the distribution of access point types to their contribution to the coverage. Routers always are able to fill up the coverage to over 90%, except for the artificially generated traces because those traces are strictly restricted to streets and therefore, fewer routers might be able to reach them. We can further observe that cell towers, which are likely to be most expensive and distant to the user, are still useful in filling coverage gaps and should still be considered in view of alternatives, such as local processing on the mobile device itself or cloud offloading.

To conclude this section, we have seen that our proposed approach of upgrading existing urban infrastructures to host cloudlets is feasible in practice and provides a high coverage to mobile users, even if only a subset of the infrastructure is upgraded. In the future, mobile applications might require more sophisticated computations that require local processing at the edge. We believe that a city-scale cloudlet deployment will be able to fill this gap.

VI. Placement Strategies for Urban Cloudlets

In the previous sections, we analyzed how well urban cloudlets can cover an area or mobile users, given that only a subset of access points is upgraded to host cloudlets. Now, we turn our attention to the question which access points should be upgraded to host cloudlets. We consider the access points to be heterogeneous in terms of costs, communication ranges, and resources they provide. This heterogeneity refers both to difference between the types of access points as well as within one type of access point. This is motivated by the differences in the underlying infrastructures and business models, as motivated in Section III-A. It is obvious that randomly selecting a certain percentage of access points—as we have done for the general coverage analysis in Section V—can lead to suboptimal results, either regarding the incurred costs or the QoS from a user's perspective. On the other hand, solving this problem in an optimal way is computationally hard and practically unfeasible, especially in dynamic edge computing environments, where available computing resources fluctuate. In this section, we will therefore present an algorithm for the cloudlet placement on urban infrastructures that is both cost-aware and at the same time tries to maximize the quality of service. We first define our model, then present and compare our approach with random placement as well as a placement strategy that greedily tries to minimize the overall costs.

A. The Model

We consider a set $AP = \{ap_1, \dots ap_n\}$ of n access points located in a 2-dimensional plane. Each access point $ap \in AP$ is of one of the types $\in \{celluar, router, lamp\}$ and has a unit-disk communication range of radius r_{ap}. If an access point is chosen to be upgraded to host a cloudlet, it can provide a certain amount of resources R_{ap}, which for instance can be modeled as the available CPU cycles of the cloudlet hardware. Using computing resources incurs a variable cost of $CVar_{ap}$ per unit of resources. In addition to the variable costs, fixed costs $CFix_{ap}$ have to be paid when an access point is upgraded. This could either be the cost of upgrading hardware or fixed costs for running the cloudlet for a certain amount of time, e.g., the costs for energy. We introduce a binary decision variable $x_{ap} \in \{0, 1\}$ to model the placement of cloudlets on access points. $x_{ap} = 1$ if a cloudlet is placed on access point $ap \in AP$, 0 otherwise. From the mobility traces, we have m user locations, denoted as $U = \{u_1, \dots, u_m\}$. Each user location requests a workload w_u. We further define $d(ap, u)$ as the Euclidean distance between an access point ap and a user location u. We characterize the association of a user to a cloudlet-enabled access point by $y_{u,ap} \in \{0, 1\}$. If user u offloads the computations to a cloudlet present at access point ap, $y_{u,ap} = 1$, otherwise $y_{u,ap} = 0$.

A placement P is therefore defined as the assignment of the variables x_{ap} and $y_{u,ap}$. Placements are subject to a number of constraints. First, we consider that we want to make placement decisions for K cloudlets, i.e.,

$$\sum_{ap \in AP} x_{ap} = K, K \in \mathbb{N}. \tag{1}$$

Obviously, users can only make use of a cloudlet at an access point if they are within its communication range and the access point has been equipped with a cloudlet, hence,

$$d(ap, u) \le r_{ap} \forall u \in U, \forall ap \in AP : y_{u,ap} = 1 \tag{2}$$

and

$$x_{ap} \ge y_{u,ap} \forall u \in U, \forall ap \in AP. \tag{3}$$

We further assume that user demands cannot be fragmented, i.e., all workload demand from one user is offloaded to exactly one cloudlet and cannot be divided:

$$\sum_{ap \in AP} y_{u,ap} = 1, \forall u \in U. \tag{4}$$

Placement decisions also need to consider the resource constraints on the cloudlets. Because user-to-cloudlet assignments should not overload the cloudlet, we have

$$\sum_{u \in U} y_{u,ap} \cdot w_u \le R_{ap}, \forall ap \in AP. \tag{5}$$

To evaluate how good a placement decision is, we take into account two factors: the costs and the overall quality of service. Costs include the fixed cost for deploying a cloudlet as well as the variable cost for each unit of resources that is offloaded. Hence, the total costs of a placement can be formulated as

$$\begin{aligned} C(P) = &\underbrace{\sum_{ap \in AP} CFix_{ap} \cdot x_{ap}}_{Fixed\ Costs} + \\ &\underbrace{\sum_{ap \in AP} \sum_{u \in U} y_{u,ap} \cdot CVar_{ap} \cdot w_u}_{Variable\ Costs} . \end{aligned} \tag{6}$$

We model the quality of service as the ratio of how much user demand can be offloaded to the cloudlets, i.e.,

$$Q(P) = \frac{\sum_{u \in U} \sum_{ap \in AP} y_{u,ap}}{m}. \tag{7}$$

Compared with our previously introduced definitions of coverage (see Section III-B), this is a variant of point coverage. However, for a point to be covered, in addition to the connectivity to a cloudlet, its computational demands must be met, i.e., there must be a cloudlet with enough (remaining) computing resources in range. Referring to the

Table V
NOTATION

AP	Set of access points
n	Total number of access points
R_{ap}	Available resources after upgrading the access point
r_{ap}	Radius of the unit-disk communication range
$CFix_{ap}$	Fixed cost for deploying a cloudlet at the access point
$CVar_{ap}$	Variable cost for using one unit of resources
x_{ap}	Binary decision variable to indicate cloudlet deployment
U	Set of user locations
m	Total number of user locations
w_u	Workload requested at user location
$d(ap, u)$	Euclidean distance between access point and user
$y_{u,ap}$	Binary decision variable for user-cloudlet assignment

results obtained from the comparison of point, path, and time coverage in Section V, we argue that this notion of point coverage will in practice also lead to users being connected to that cloudlet for the entire time along their path. Given these definitions, the overall utility of a placement is defined as

$$Utility(P) = \alpha \cdot \frac{\max(C(P)) - C(P)}{\max(C(P)) - \min(C(P))} + (1-\alpha) \cdot Q(P), \tag{8}$$

where $\alpha \in [0, 1]$ is a parameter to trade costs against quality of service. Note that we negate the cost factor $C(P)$ in order to represent lower costs by a higher utility value. Since $\min C(P)$ and $\max C(P)$ are constant, this part of the utility function is still linear. Table V summarizes the notation of our model.

B. Problem Statement

Given the above definitions, we state the cloudlet placement problem as follows. Place K cloudlets on access points such that

$$\begin{aligned} \max \quad & Utility(P) \\ \text{s.t.} \quad & (1), (2), (3), (4), (5) \\ & x_{ap} \in \{0, 1\}, \\ & y_{u,ap} \in \{0, 1\}, \ \forall ap \in AP, \ \forall u \in U. \end{aligned}$$

Our goal is therefore to maximize the offloading ratio, i.e., the number of users that will be able to offload computations to cloudlets while making cost-aware placement decisions for cloudlets on the access points. This problem can be modeled as a variant of the *facility location* or *k-Median* problem, both of which have been proven to be NP-hard [75]. In practice, this means we cannot find an optimal solution in a reasonable amount of time. However, in view of the dynamics in a real-city network, e.g., due to user mobility or changes in demands, we need to be able to quickly adapt to those changes.

C. The Approach

To make the cloudlet placement problem more tractable, we propose `GSCORE` (Grid-Score), a cloudlet placement algorithm described in this section. Instead of considering single user requests or make a decision on a global scale, the algorithm performs cloudlet placements locally. We divide the area to be covered by cloudlets into grids $G = \{g_1, \ldots, g_j\}$ with uniform edge length gs. Based on the user locations and the request size of each user, we can then compute the total size of the requests per grid $w_g = \sum w_u$ for every user u located in that grid. In reality, the request sizes of the grids might be estimated by measurements from network providers that are able to estimate the number of users and the offloading traffic they generate. Our algorithm operates solely on the knowledge of the individual grid cells with their associated grid sizes. For each grid cell, a local decision is made to place a certain number of cloudlets on the available access points in that grid. First, we make a decision on where to place cloudlets and later assign the individual user requests to the cloudlets to evaluate the system utility as defined earlier.

Cloudlet Placement: The pseudocode of `GSCORE` is shown in Algorithm 1. Its main loop iterates over the grid cells until the desired number K of cloudlets have been placed (lines 1-27). The cells are traversed in decreasing order of requests sizes, i.e., we begin with the cells that have the highest request sizes. Next, for each of the access points available in that cell, a score is computed (lines 4-12).

Algorithm 1 `GSCORE`

1: **while** $\sum_{i=0}^{n} x_i < K$ **do**
2: $\quad g_h \leftarrow G.getHighestRequestSize()$
3: $\quad S = \emptyset$
4: $\quad$ **for** $ap \in AP$ located in g_h **do**
5: $\qquad cr \leftarrow \frac{CFix_{ap} + CVar_{ap} \cdot R_{ap}}{R_{ap}}$
6: $\qquad factor_{area} \leftarrow \frac{|A(ap) \cap A(g_h)|}{|A(g_h)|}$
7: $\qquad factor_{capacity} \leftarrow \frac{R_{ap}}{w_{g_h}}$
8: $\qquad factor_{cr} \leftarrow \frac{max(cr) - cr}{max(cr) - min(cr)}$
9: $\qquad factor_{QoS} \leftarrow \frac{factor_{area} + factor_{capacity}}{2}$
10: $\qquad score_{ap} \leftarrow \alpha \cdot factor_{cr} + (1-\alpha) \cdot factor_{QoS}$
11: $\qquad S \leftarrow S \cup \{score_{ap}\}$
12: $\quad$ **end for**
13: $\quad numToPlace \leftarrow \lfloor \ln(\frac{w_{g_h}}{\bar{w_g}}) + \ln(g_s) + \frac{K}{|G|} \rfloor$
14: $\quad placedCap \leftarrow 0$
15: $\quad$ **for** $k \in [0, numToPlace]$ **do**
16: $\qquad score_{ap} \leftarrow max(S)$
17: $\qquad x_{ap} = 1$
18: $\qquad placedCap \leftarrow placedCap + R_{ap}$
19: $\qquad S \leftarrow S \setminus \{score_{ap}\}$
20: $\qquad AP \leftarrow AP \setminus \{ap\}$
21: $\quad$ **end for**
22: $\quad$ **if** $(w_{g_h} - placedCap) > 2 \cdot \bar{w_g}$ **then**
23: $\qquad w_{g_h} \leftarrow w_{g_h} - (placedCap \cdot \ln(w_{g_h}))$
24: $\quad$ **else**
25: $\qquad w_{g_h} \leftarrow w_{g_h} - placedCap$
26: $\quad$ **end if**
27: **end while**

The score reflects the tradeoff between cost considerations and quality of service. The cost factor $factor_{cr} = \frac{max(cr) - cr}{max(cr) - min(cr)}$ (line 8) normalizes the costs-to-resource ratio of the access point; hence, access points with higher resources at the same costs will be ranked higher. To normalize this metric, we assume an upper bound in the sense that each access point's capacity will be fully utilized. The factor for the quality of service $factor_{QoS} = \frac{factor_{area} + factor_{capacity}}{2}$

reflects how much of the grid area is covered by the communication range and what ratio of the grid's request demands can be satisfied by that access point. Note that we again only consider these factors on a grid cell level, i.e., a router with a larger communication range that covers an entire cell might have the same value for $factor_{area}$ as a cell tower, even though the latter in reality spans over multiple grid cells. Similarly, at this point, we completely disregard whether there will actually be users within the range of this access point. Doing so would greatly increase the complexity since it would require iterating over every individual data point. By selecting appropriate grid sizes in the evaluation, we will show that this approach is a reasonable approximation. Both factors are weighted according to the desired α for the calculation of the access point score (line 10).

From our raw user data, we could observe that the number of users per grid—and hence the generated request sizes—are not uniformly distributed. Instead, we see few grid cells with substantially higher request sizes than the average. This will result in many access points being placed in those grids, even if they are not enough to satisfy the total user demands of that grid. At the same time, this reduces the number of access points that could be placed to easily satisfy a greater offloading ratio in other grids. To mitigate this behavior, we compute the number of cloudlets to be placed in a grid cell, as shown in line 13. This formula normalizes the impact of grids with exceptionally high request sizes by taking $\ln(\frac{w_{g_h}}{\bar{w}_g})$. In addition, we also factor in the size of the grid (in the sense that we allow more cloudlets to be placed in larger grids) and the ratio of K to the number of grid cells. According to this function, the according number of cloudlets with the highest score will be added to the grid cell (lines 15-21). After having placed the corresponding number of cloudlets in a grid cell, its workload demand is adjusted in the following way: We assume each cloudlet will be used to full capacity. In addition, we again take into account the characteristics of the request size distribution to ensure that grids with lesser workload will also be iterated over. Hence, if the workload of a grid remains larger than two times the average workload, we adjust the new workload request estimation of the grid by multiplying the placed capacity of the cloudlets with the logarithm of the original request demand (line 23).

User-to-Cloudlet Assignment: To compute the utility value (see Equation (8)), we now assign the requests of individual users with the following strategy. Since the fixed costs have already been determined by the placement strategy, for each request, we choose the cloudlet with the lowest variable costs per resource unit that is within the range of the user.

D. Evaluation

We now compare our proposed approach with the following alternative strategies for cloudlet placement:

Random (`RND`): This approach randomly selects K access points where cloudlets are placed on. Obviously, the distribution of the K selected access points with respect to their type will follow the one of the dataset, meaning that we will have few expensive locations (i.e., cell towers), and a high number of routers and street lamps. They will however not necessarily be located in areas where the coverage has a high impact on the QoS, i.e., areas with a large number of users. Instead, cloudlets are likely to be spread evenly throughout the city.

Greedy-Cost-Aware (`GC`): This strategy tries to minimize the overall costs by iteratively selecting the access points with the lowest overall costs, defined as the sum of fixed and variable costs, assuming the placed cloudlet will be used to full capacity. Similar to `RND`, this will disregard the geographic distribution of user workloads and might penalize choices that have higher costs but a good costs-to-resource ratio. We therefore expect this approach to perform worse than `RND` in terms of the delivered quality of service. However, for very cost-restricted deployment models, this will lead to the insight of how much offloading is possible.

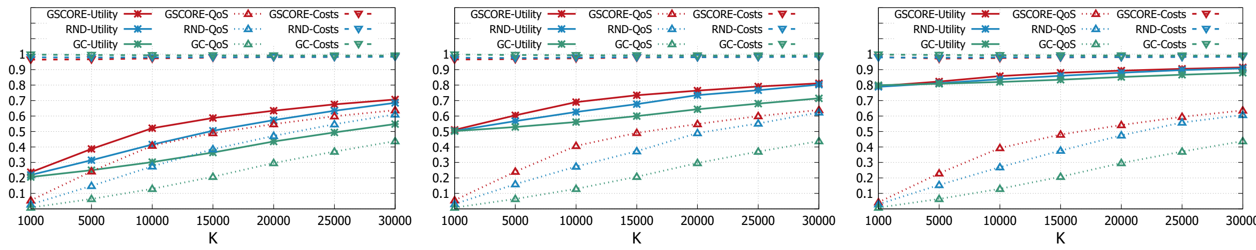
1) Setup: We built a simulation tool in Ruby to evaluate our placement strategy. The base data is stored in a PostgreSQL database with the spatial extender PostGIS enabled. We use the values listed in Table VI as our experimental settings for the modeling of access point attributes. The values reflect the heterogeneity of our access point infrastructure and their deployment characteristics as described in Section III. It is important to note that even within one type of access point, we consider the values for the communication range, resources, and costs to be variable.

Table VI
EVALUATION PARAMETERS

	Cellular Base Stations	Routers	Street Lamps
Communication range (m)	random (300,1000)	random (10,70)	random (20,80)
Resources	random (2000,5000)	random(5,100)	random(5,50)
Fixed cost	random (1000, 10 000)	random(1,100)	100
Variable cost	random(5,10)	random(1,5)	1

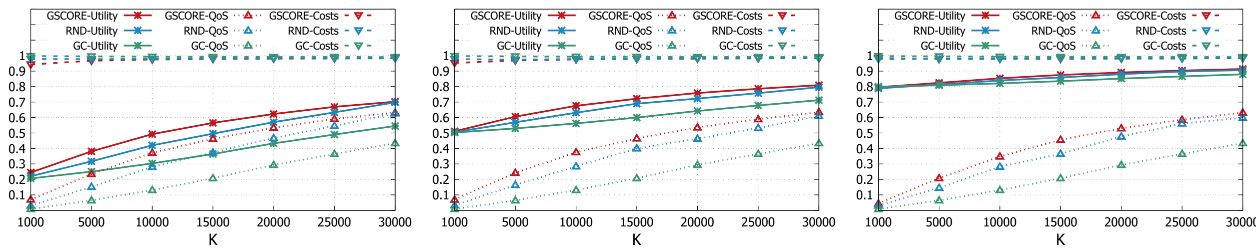
(a) Grid size of 50m (b) Grid size of 100m

Figure 7. Grid sizes for evaluation.



(a) $\alpha = 0.2$ (b) $\alpha = 0.5$ (c) $\alpha = 0.8$

Figure 8. Placement evaluation for a grid size of 50m.



(a) $\alpha = 0.2$ (b) $\alpha = 0.5$ (c) $\alpha = 0.8$

Figure 9. Placement evaluation for a grid size of 100m.

The only exception are the costs for using street lamps, which we assume to be fixed because they are operated by a single stakeholder, the municipality. To model the workload that users want to offload, we take the data points from all three datasets. Even though they were captured over a period of time, for the evaluation, we assume they jointly represent demand spots of mobile users throughout the city at a single point in time. Each data point is assigned a requested workload of 1 or 2 units. We conduct our experiments with two different grid sizes of 50 meters and 100 meters for their edge lengths. Heatmaps of these two setups that visualize the total number of requests per grids are shown in Figures 7(a) and 7(b). Darker red squares represent grid cells with higher demands, while blue ones are areas with lower demands.

2) Results: Figures 8 and 9 display the results of our placement strategy for grid sizes of 50m and 100m, respectively. For each grid size, we evaluate the placement strategies with values 0.2, 0.5, and 0.8 for α. Recall that $\alpha = 0.5$ equally weighs costs and quality of service for the overall utility, while lower values put more emphasis on the quality of service and vice-versa. For K, we use values from 1000 to 30 000. Besides the overall utility, the plots include the two components of the utility function, i.e., values for the costs and QoS. As defined in Equation (8), higher values denote lower costs and better QoS. Overall, we see that our proposed `GSCORE` algorithm surpasses `RND` and `GC` in each evaluated scenario for both the overall utility and the QoS part of the utility function. Even though barely visible in the graphs, `GSCORE` leads to a very small increase in the cost factor (in most cases around 1-2%) compared to the other strategies. However, the gain in terms of QoS when using `GSCORE` to place cloudlets is much higher. Take as an example the results for $K = 10\,000$ and $\alpha = 0.2$. Irrespective of the grid size, `GSCORE` achieves a QoS value that is three times higher compared to `GC`. Therefore, the essential benefit that `GSCORE` provides is that it trades a small fraction of cost increase for a much greater increase in the quality of service. Consequently, for smaller values of α, the gain in the overall utility when using `GSCORE` is higher. Since it takes into account the values of α for the scoring, `GSCORE` can be tuned to adapt to different deployment and business models for cloudlets.

Regarding the different grid sizes, we observe only a small difference between grid sizes of 50m and 100m. For a smaller grid size, `GSCORE` gains a little in the overall utility; however, smaller grid sizes result in iterating over more grids, and hence, a greater computational overhead to make the placement decision. We leave the exploration of this tradeoff for future work and plan to further investigate how other grid sizes perform.

As we can also see, for larger values of K, the difference between `GSCORE` and `RND` becomes smaller because it is more likely that there is a great overlap in the two chosen subsets of access points. This can be seen in the result plots if we look at K values of 30 000 and compare this value with the total number of access points in the inner city

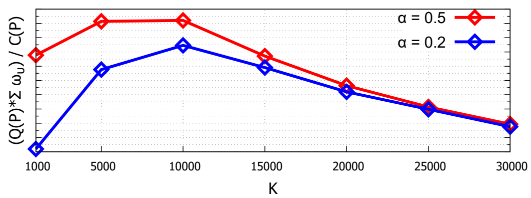


Figure 10. Choosing K by quality-to-cost ratio.

boundary (37 648). The results for small values of $K = 1000$ behave similarly. In that case, there are so few cloudlets available that only a small fraction of all user demands can be met, irrespective of the employed placement strategy. GSCORE performs best for input sizes between $K = 5000$ and $K = 20\,000$ or in other words, between 13% and 53% of all available access points. We therefore believe this placement strategy can be viable in practice, since in a real-world deployment, one would neither upgrade very few (because there would be no substantial gain for users) nor nearly all access points (because of practical limitations in terms of costs). Furthermore, we expect GSCORE to perform even better compared to RND if the environment is more heterogeneous than in our evaluation setup, e.g., if cloudlets on street lamps are owned by different operators and therefore have different costs associated with them. Note that our notion of placing K cloudlets can easily be adapted to match other constraints, such as the total costs. Instead of K denoting the number of access points, K could for example model the maximum allowed fixed costs of the deployment.

To further investigate practical insights on how to choose a suitable K, we analyze the quality-to-cost ratio for different values of K. Instead of using the normalized utility (see Equation (8)), we consider the absolute costs $C(P)$ and for the quality of service, we weight the absolute workload values by our QoS part of the utility. Hence, the ratio is given as $\frac{Q(P)\cdot\sum w_u}{C(P)}$. Figure 10 plots the value for this ratio for different values of K. For this example, we use the placement decision that GSCORE outputs for a grid size of 100m and the same parameters as described before. As $alpha$, we use 0.2 and 0.5. We omit α = 0.8 because this shows a very clear trend towards very low values of K. From the plot, we can observe that if we use the absolute costs and workloads that will be offloaded as the metric, there is a sweet spot for the value of K at 10 000. Very low values of K are not a good choice since we need a certain threshold of cloudlets deployed to be able to cover a certain city area (regardless of the available or used resources). On the other hand, we see that for very large numbers of cloudlets, the additional costs incurred surpass the gain in quality of service. For the practical deployment, we can therefore conclude that medium-sized values of K are most beneficial. It is worth noticing that these are also the ranges of K where our proposed algorithm performs best.

VII. Discussion and Future Work

In this paper, we have examined the placement of different types of cloudlets in an urban space using existing access point infrastructures, namely, cell towers, routers, and street lamps. We first studied the coverage that we can achieve by selecting only a subset of all access points to be upgraded. We did so by first considering spatial coverage only and then used mobility traces to evaluate point, path, and time coverage. The results of this analysis enable different stakeholders (e.g., municipalities and network operators) to estimate the number of cloudlets required to achieve a certain degree of coverage. As a second contribution, we presented GSCORE, a placement strategy that operates on local grids and is able to trade a small portion of cost savings against a substantially higher gain in the number of users that can offload. Furthermore, this tradeoff can be adjusted to model different underlying business and incentives models. We showed that this strategy is overall more beneficial than randomly upgrading a certain number of access points or choosing the ones with the lowest costs.

This work opens up some future research directions. While our placement strategy considers cloudlet heterogeneity in the domain of costs, resources, and communication ranges, we did not consider the available bandwidth and latency. Depending on the user location, the available bandwidth and latency to one particular cloudlet might vary significantly. In future deployments, we could envision that users collect measurements of network conditions and this data to be included as a factor in our QoS model. Our algorithm performs a static placement decision. In view of changes in user workload, user mobility, or the availability of the cloudlets themselves, future work should investigate the adaptation of cloudlet placements at runtime. Furthermore, besides user mobility, data mobility is another factor to be considered in future work, especially for the user-to-cloudlet assignment. Taking user and data mobility into account might prevent unnecessary or costly migrations of services.

Our evaluation has shown that a cost-based placement leads to a lower quality of service and overall lower utility compared to a random or coverage-based placement approach. Therefore, a cost-based placement also leads to lower network traffic savings on the part of the infrastructure provider. Because of the complex interplay between different stakeholders, new business models and incentive mechanisms are required for a joint optimization.

Acknowledgement

We thank our shepherd, Padmanabhan Pillai, and the anonymous reviewers for their valuable comments and suggestions. This work has been cofunded by the German Research Foundation (DFG) and the National Nature Science Foundation of China (NSFC) joint project under Grant No. 392046569 (DFG) and No. 61761136014 (NSFC), and as part of the Collaborative Research Center 1053 - MAKI (DFG).

REFERENCES

[1] E. Cuervo, K. Chintalapudi, and M. Kotaru, "Creating the perfect illusion: What will it take to create life-like virtual reality headsets?" in *Proceedings of the 19th International Workshop on Mobile Computing Systems & Applications (HotMobile 2018)*, 2018, pp. 7–12.

[2] N. Fernando, S. W. Loke, and W. Rahayu, "Mobile cloud computing: A survey," *Future Generation Computer Systems*, vol. 29, no. 1, pp. 84–106, 2013.

[3] D. T. Hoang, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: architecture, applications, and approaches," *Wireless Communications and Mobile Computing*, vol. 13, no. 18, pp. 1587–1611, 2013.

[4] S. Yi, C. Li, and Q. Li, "A Survey of Fog Computing: Concepts, Applications and Issues," *Proceedings of the 2015 Workshop on Mobile Big Data (Mobidata)*, pp. 37–42, 2015.

[5] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Communications Surveys and Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017.

[6] K. Ha, Z. Chen, W. Hu, W. Richter, P. Pillai, and M. Satyanarayanan, "Towards wearable cognitive assistance," in *Proceedings of the 12th International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '10. ACM, 2014, pp. 68–81.

[7] J. Gedeon, J. Heuschkel, L. Wang, and M. Mühlhäuser, "Fog computing: Current research and future challenges," in *1. GI/ITG KuVS Fachgespräche Fog Computing*, 2018, pp. 1–4.

[8] S. Yi, Z. Hao, Z. Qin, and Q. Li, "Fog Computing: Platform and Applications," in *Proceedings of the Third IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb)*, 2015, pp. 73–78.

[9] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, pp. 13–16, 2012.

[10] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[11] A. Chandra, J. Weissman, and B. Heintz, "Decentralized edge clouds," *IEEE Internet Computing*, vol. 17, no. 5, pp. 70–73, 2013.

[12] M. Satyanarayanan, "The emergence of edge computing," *IEEE Computer*, vol. 50, no. 1, pp. 30–39, 2017.

[13] M. Satyanarayanan, P. Simoens, Y. Xiao, P. Pillai, Z. Chen, K. Ha, W. Hu, and B. Amos, "Edge analytics in the internet of things," *IEEE Pervasive Computing*, vol. 14, no. 2, pp. 24–31, 2015.

[14] B. Varghese, N. Wang, S. Barbhuiya, P. Kilpatrick, and D. S. Nikolopoulos, "Challenges and opportunities in edge computing," in *Proceedings of the 2016 IEEE International Conference on Smart Cloud (SmartCloud)*, 2016, pp. 20–26.

[15] J. Gedeon, N. Himmelmann, P. Felka, F. Herrlich, M. Stein, and M. Mühlhäuser, "vStore: A context-aware framework for mobile micro-storage at the edge," in *Proceedings of the 9th International Conference on Mobile Computing, Applications and Services (MobiCASE)*, 2018, pp. 165–182.

[16] H. Chang, A. Hari, S. Mukherjee, and T. V. Lakshman, "Bringing the cloud to the edge," in *Proceedings of the IEEE INFOCOM Workshops*, 2014, pp. 346–351.

[17] M. Satyanarayanan, P. Bahl, R. Cáceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Computing*, vol. 8, no. 4, pp. 14–23, 2009.

[18] E. Cuervo, A. Balasubramanian, D.-K. Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl, "MAUI: Making smartphones last longer with code offload," in *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '10. ACM, 2010, pp. 49–62.

[19] K. Hong, D. J. Lillethun, U. Ramachandran, B. Ottenwälder, and B. Koldehofe, "Mobile fog: a programming model for large-scale applications on the internet of things," in *Proceedings of the 2nd ACM Workshop on Mobile cloud computing (MCC)*, 2013, pp. 15–20.

[20] C. Perera, Y. Qin, J. C. Estrella, S. Reiff-Marganiec, and A. V. Vasilakos, "Fog Computing for Sustainable Smart Cities: A Survey," *ACM Computing Surveys*, vol. 50, no. 3, pp. 32:1–32:43, 2017.

[21] T. Taleb, S. Dutta, A. Ksentini, M. Iqbal, and H. Flinck, "Mobile edge computing potential in making cities smarter," *IEEE Communications Magazine*, vol. 55, no. 3, pp. 38–43, 2017.

[22] J. M. Schleicher, M. Vögler, S. Dustdar, and C. Inzinger, "Enabling a smart city application ecosystem: Requirements and architectural aspects," *IEEE Internet Computing*, vol. 20, no. 2, pp. 58–65, 2016.

[23] A. Zanella, L. Vangelista, N. Bui, A. Castellani, and M. Zorzi, "Internet of Things for Smart Cities," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, 2014.

[24] B.-G. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti, "CloneCloud: Elastic execution between mobile device and cloud," in *Proceedings of the 6th Conference on Computer Systems*, ser. EuroSys '11. ACM, 2011, pp. 301–314.

[25] W. Felter, A. Ferreira, R. Rajamony, and J. Rubio, "An updated performance comparison of virtual machines and linux containers," in *Proceedings of the 2015 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*, 2015, pp. 171–172.

[26] S. Soltesz, H. Pötzl, M. E. Fiuczynski, A. C. Bavier, and L. L. Peterson, "Container-based operating system virtualization: a scalable, high-performance alternative to hypervisors," in *Proceedings of the 2007 EuroSys Conference*, 2007, pp. 275–287.

[27] P. Liu, D. Willis, and S. Banerjee, "Paradrop: Enabling lightweight multi-tenancy at the network's extreme edge," in *Proceedings of the IEEE/ACM Symposium on Edge Computing (SEC)*, 2016, pp. 1–13.

[28] A. Madhavapeddy and D. J. Scott, "Unikernels: the rise of the virtual library operating system," *Communications of the ACM*, vol. 57, no. 1, pp. 61–69, 2014.

[29] A. Madhavapeddy, R. Mortier, C. Rotsos, D. J. Scott, B. Singh, T. Gazagnaire, S. Smith, S. Hand, and J. Crowcroft, "Unikernels: library operating systems for the cloud," in *Proceedings of the International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, 2013, pp. 461–472.

[30] F. Manco, C. Lupu, F. Schmidt, J. Mendes, S. Kuenzer, S. Sati, K. Yasukata, C. Raiciu, and F. Huici, "My VM is lighter (and safer) than your container," in *Proceedings of the 26th Symposium on Operating Systems Principles (SOSP)*, 2017, pp. 218–233.

[31] D. Fesehaye, Y. Gao, K. Nahrstedt, and G. Wang, "Impact of cloudlets on interactive mobile cloud applications," in *Proceedings of the 16th IEEE International Enterprise Distributed Object Computing Conference (EDOC)*, 2012, pp. 123–132.

[32] Y. Jararweh, L. Tawalbeh, F. Ababneh, and F. Dosari, "Resource efficient mobile computing using cloudlet infrastructure," in *Proceedings of the 9th IEEE International Conference on Mobile Ad-hoc and Sensor Networks (MSN)*, 2013, pp. 373–377.

[33] M. Chen, Y. Hao, Y. Li, C. F. Lai, and D. Wu, "On the computation offloading at ad hoc cloudlet: architecture and service modes," *IEEE Communications Magazine*, vol. 53, no. 6, pp. 18–24, 2015.

[34] F. Chi, X. Wang, W. Cai, and V. C. M. Leung, "Ad hoc cloudlet based cooperative cloud gaming," in *Proceedings of the 6th International Conference on Cloud Computing Technology and Science*, 2014, pp. 190–197.

[35] E. Zeydan, E. Bastug, M. Bennis, M. A. Kader, I. A. Karatepe, A. S. Er, and M. Debbah, "Big data caching for networking: moving from cloud to edge," *IEEE Communications Magazine*, vol. 54, no. 9, pp. 36–42, 2016.

[36] M. Satyanarayanan, Z. Chen, K. Ha, W. Hu, W. Richter, and P. Pillai, "Cloudlets: at the leading edge of mobile-cloud convergence," in *Proceedings of the 6th International Conference on Mobile Computing, Applications and Services (MobiCASE)*, 2014, pp. 1–9.

[37] T. Verbelen, P. Simoens, F. D. Turck, and B. Dhoedt, "Cloudlets: Bringing the cloud to the mobile user," in *Proceedings of the Third ACM Workshop on Mobile Cloud Computing and Services*, ser. MCS '12, 2012, pp. 29–36.

[38] Z. Pang, L. Sun, Z. Wang, E. Tian, and S. Yang, "A survey of cloudlet based mobile computing," in *Proceedings of the International Conference on Cloud Computing and Big Data (CCBD)*, 2016, pp. 268–275.

[39] A. Y. Ding, B. Han, Y. Xiao, P. Hui, A. Srinivasan, M. Kojo, and S. Tarkoma, "Enabling energy-aware collaborative mobile data offloading for smartphones," in *Proceedings of the 10th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, (SECON)*, 2013, pp. 487–495.

[40] J. Gedeon, C. Meurisch, D. Bhat, M. Stein, L. Wang, and M. Mühlhäuser, "Router-based brokering for surrogate discovery in edge computing," in *Proceedings of the 37th IEEE International Conference on Distributed Computing Systems Workshops (ICDCSW)*, 2017, pp. 145–150.

[41] C. Meurisch, A. Seeliger, B. Schmidt, I. Schweizer, F. Kaup, and M. Mühlhäuser, "Upgrading wireless home routers for enabling large-scale deployment of cloudlets," in *Proceedings of the 7th International Conference on Mobile Computing, Applications, and Services (MobiCASE)*, 2015, pp. 12–29.

[42] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: A survey," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 450–465, 2018.

[43] V. Chandrasekhar, J. G. Andrews, and A. Gatherer, "Femtocell networks: a survey," *IEEE Communications Magazine*, vol. 46, no. 9, pp. 59–67, 2008.

[44] A. Sathiaseelan, A. Lertsinsrubtavee, A. Jagan, P. Baskaran, and J. Crowcroft, "Cloudrone: Micro clouds in the sky," in *Proceedings of the 2nd Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use (DroNet)*, 2016, pp. 41–44.

[45] B. Wang, "Coverage problems in sensor networks: A survey," *ACM Computing Surveys*, vol. 43, no. 4, pp. 32:1–32:53, 2011.

[46] S. M. Mohamed, H. S. Hamza, and I. A. Saroit, "Coverage in mobile wireless sensor networks (M-WSN): A survey," *Computer Communications*, vol. 110, pp. 133–150, 2017.

[47] R. Mulligan and H. M. Ammari, "Coverage in wireless sensor networks: A survey," *Network Protocols & Algorithms*, vol. 2, no. 2, pp. 27–53, 2010.

[48] A. A. Ghosh and S. K. Das, "Coverage and connectivity issues in wireless sensor networks: A survey," *Pervasive and Mobile Computing*, vol. 4, no. 3, pp. 303–334, 2008.

[49] M. Li, W. Cheng, K. Liu, Y. He, X. Li, and X. Liao, "Sweep coverage with mobile sensors," *IEEE Transactions on Mobile Computing*, vol. 10, no. 11, pp. 1534–1545, 2011.

[50] B. Liu, O. Dousse, J. Wang, and A. Saipulla, "Strong barrier coverage of wireless sensor networks," in *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ser. MobiHoc, 2008, pp. 411–420.

[51] J. Gedeon and I. Schweizer, "Understanding spatial and temporal coverage in participatory sensor networks," in *Proceedings of the 40th IEEE Local Computer Networks Conference Workshops (LCN Workshops)*, 2015, pp. 699–707.

[52] G. Fan and S. Jin, "Coverage Problem in Wireless Sensor Network: A Survey," *Journal of Networks*, vol. 5, no. 9, pp. 1033–1040, 2010.

[53] C. Meurisch, J. Gedeon, A. Gogel, T. A. B. Nguyen, F. Kaup, F. Kohnhäuser, L. Baumgärtner, M. Schmittner, and M. Mühlhäuser, "Temporal coverage analysis of router-based cloudlets using human mobility patterns," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, 2017, pp. 1–6.

[54] J. Gedeon, J. Krisztinkovics, C. Meurisch, M. Stein, L. Wang, and M. Mühlhäuser, "A multi-cloudlet infrastructure for future smart cities: An empirical study," in *Proceedings of the 1st International Workshop on Edge Systems, Analytics and Networking (EdgeSys)*, 2018, pp. 19–24.

[55] X. Meng, V. Pappas, and L. Zhang, "Improving the scalability of data center networks with traffic-aware virtual machine placement," in *Proceedings of the 29th IEEE International Conference on Computer Communications (INFOCOM)*, 2010, pp. 1154–1162.

[56] J. T. Piao and J. Yan, "A network-aware virtual machine placement and migration approach in cloud computing," in *Proceedings of the 9th International Conference on Grid and Cloud Computing (GCC)*, 2010, pp. 87–92.

[57] T. Bahreini and D. Grosu, "Efficient placement of multi-component applications in edge computing systems," in *Proceedings of the 2nd ACM/IEEE Symposium on Edge Computing*, 2017, pp. 5:1–5:11.

[58] L. Yang, J. Cao, G. Liang, and X. Han, "Cost aware service placement and load dispatching in mobile cloud systems," *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1440–1452, 2016.

[59] J. Gedeon, M. Stein, L. Wang, and M. Mühlhäuser, "On scalable in-network operator placement for edge computing," in *Proceedings of the 27th International Conference on Computer Communications and Networks (ICCCN)*, 2018, pp. 1–9.

[60] Z. Xu, W. Liang, W. Xu, M. Jia, and S. Guo, "Capacitated cloudlet placements in wireless metropolitan area networks," in *Proceedings of the 40th IEEE Conference on Local Computer Networks (LCN)*, 2015, pp. 570–578.

[61] M. Jia, J. Cao, and W. Liang, "Optimal cloudlet placement and user to cloudlet allocation in wireless metropolitan area networks," *IEEE Transactions on Cloud Computing*, vol. 5, no. 4, pp. 725–737, 2017.

[62] A. Ceselli, M. Premoli, and S. Secci, "Cloudlet network design optimization," in *Proceedings of the 14th IFIP Networking Conference*, 2015, pp. 1–9.

[63] M. Syamkumar, P. Barford, and R. Durairajan, "Deployment characteristics of "the edge" in mobile edge computing," in *Proceedings of the 2018 Workshop on Mobile Edge Communications (MECOMM)*, 2018, pp. 43–49.

[64] H. Yao, C. Bai, M. Xiong, D. Zeng, and Z. Fu, "Heterogeneous cloudlet deployment and user-cloudlet association toward cost effective fog computing," *Concurrency and Computation: Practice and Experience*, vol. 29, no. 16, pp. 1–9, 2017.

[65] E. Bulut and B. K. Szymanski, "Wifi access point deployment for efficient mobile data offloading," *Mobile Computing and Communications Review*, vol. 17, no. 1, pp. 71–78, 2013.

[66] E. Bulut and B. Szymanski, "Rethinking offloading wifi access point deployment from user perspective," in *Proceedings of the 12th IEEE International Conference on Wireless and Mobile Computing (WiMob)*, 2016, pp. 1–6.

[67] A. Elmokashfi, D. Zhou, and D. Baltrünas, "Adding the next nine: An investigation of mobile broadband networks availability," in *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking (MobiCom)*, 2017, pp. 88–100.

[68] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile edge computing-A key technology towards 5G," *ETSI white paper*, no. 11, pp. 1–16, 2015.

[69] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On Multi-Access Edge Computing: A Survey of the Emerging 5G Network Edge Architecture & Orchestration," *IEEE Communications Surveys and Tutorials*, vol. 19, no. 3, pp. 1657–1681, 2017.

[70] C.-F. Huang and Y.-C. Tseng, "The coverage problem in a wireless sensor network," *Mobile Networks and Applications*, vol. 10, no. 4, pp. 519–528, Aug 2005.

[71] I. Schweizer and B. Schmidt, "Kraken.me: Multi-device user tracking suite," in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, ser. UbiComp '14 Adjunct, 2014, pp. 853–862.

[72] C. Fiandrino, A. Capponi, G. Cacciatore, D. Kliazovich, U. Sorger, P. Bouvry, B. Kantarci, F. Granelli, and S. Giordano, "Crowdsensim: a simulation platform for mobile crowdsensing in realistic urban environments," *IEEE Access*, vol. 5, pp. 3490–3503, 2017.

[73] V. F. S. Mota, D. F. Macedo, Y. Ghamri-Doudane, and J. M. S. Nogueira, "On the feasibility of wifi offloading in urban areas: The paris case study," in *Proceedings of the IFIP Wireless Days (WD)*, 2013, pp. 1–6.

[74] V. Burger, M. Seufert, F. Kaup, M. Wichtlhuber, D. Hausheer, and P. Tran-Gia, "Impact of WiFi offloading on video streaming QoE in urban environments," in *Proceedings of the International Conference on Communication Workshops ICCW*, 2015, pp. 1717–1722.

[75] K. Jain and V. V. Vazirani, "Approximation algorithms for metric facility location and *k*-median problems using the primal-dual schema and lagrangian relaxation," *Journal of the ACM*, vol. 48, no. 2, pp. 274–296, 2001.