Towards a context-aware Wi-Fi-based Fog Node discovery scheme using cellular footprints

Zeineb Rejiba*, Xavier Masip-Bruin*, Eva Marín-Tordera*
*Advanced Network Architectures Lab (CRAAX), Universitat Politècnica de Catalunya (UPC), Barcelona, Spain
{zeinebr, xmasip, eva}@ac.upc.edu

***Abstract*—Recently, new computing paradigms such as fog and edge computing started to emerge in an attempt to cope with the low latency requirements of new classes of IoT applications. In order for these paradigms to fully realize their potential, an important challenge to address is the discovery of fog nodes (FNs) with spare computational resources that can be used to host time-sensitive and computationally intensive application components. We particularly focus on this problem in this paper, considering the case of a WiFi-based FN discovery process. More specifically, we evaluate how practical it is to trigger the discovery of fog nodes based on a mobile phone's historical cellular footprints in order to obtain a high discovery rate and a low energy overhead. To this end, we conducted a small-scale cellular data collection to be used to test different learning approaches, including the K-Nearest Neighbors and the decision tree algorithms as well as a Hidden Markov Model (HMM). According to our evaluation results, HMM was found to achieve the maximum discovery and energy saving ratios. The impact of initial FN misdetections on the user-FN contact ratio has also been studied.**

***Index Terms*—device discovery, fog computing, context-awareness, cellular footprints**

I. Introduction

New classes of pervasive mobile and IoT applications are constantly being developed and continuously raising tight QoS requirements. Such applications usually rely on a cloud-based backend for smart data processing and long-term storage. However, such an approach fails to cope with the aforementioned QoS constraints, especially in terms of perceived end-to-end latency. Besides, given the frequency of application-to-cloud data transmissions, this also leads to an unnecessary volume of core network traffic overwhelming the underlying infrastructure, while such data is mostly needed at the edge of the network where it was generated.

These observations have led to the advent of new computing paradigms, such as the Fog computing [1] and the Fog-to-Cloud computing [2], where the use of cloud resources is complemented by idle resources provided by distributed nodes at the edge of the network, generally referred to as fog nodes (FNs). Leveraging the edge presence of such FNs and their vicinity to end users, increased QoS levels can be guaranteed.

However, in order to fully take advantage of such FNs, the considered applications need to be equipped with a suitable discovery service allowing the detection of nearby FNs, even when no knowledge about their potential presence has been previously acquired. Within this context, we proposed in a previous work [3] a 802.11 beaconing-based discovery solution, comprised of two processes: broadcasting and scanning. In the broadcasting process, the FN advertises discovery information within specific fields of the Wi-Fi beacon; whereas a device willing to make use of that FN's resources performs a scanning process to detect such beacons and consequently connect with that FN.

When evaluating the use of such a WiFi-based discovery using a real smartphone-based implementation [4], we found that the periodic scan process it is based on consumes a significant amount of power, given its wireless nature. In fact, a too high scan frequency incurs a high power consumption while a too low frequency may lead to potential FNs being missed. Thus, instead of a periodic scan process, we study in this paper the possibility of realizing a context-aware scanning mechanism in order to achieve a smarter discovery process with increased energy efficiency and a high discovery rate so that the promised benefits of a fog-based execution can be guaranteed.

In order to achieve the proposed context-awareness scan, the main idea consists in using previously-observed information about the surrounding cellular context of a mobile device (i.e. the cellular tower ID and the signal strength) to infer locations with a high likelihood of presence of a FN. Since a FN is predicted to be present within such locations, then the scan should be enabled. However, if the current cellular information has been correlated with a location where no FN has been seen previously, the scan should be disabled to save energy.

To evaluate the feasibility of such an approach in the FN discovery process, we conducted a small-scale cellular data collection experiment in a small section of the city and used the obtained data to train and test different learning approaches, including the K-Nearest Neighbors (KNN) and the decision tree algorithms as well as a Hidden Markov Model (HMM). The discovery ratio, the saving ratio and the user-FN contact ratio have been provided as performance metrics for the three considered approaches.

The remainder of this paper is organized as follows: Section II provides an overview of related works about cellular-based context-awareness mechanisms and discovery solutions proposed in fog computing scenarios. Section III describes the proposed cellular context-based fog node discovery. Section IV highlights the evaluation results and some concluding remarks are provided in Section V.

II. Related work

In this section, we provide an overview of related works about cellular-based context-awareness mechanisms in addition to reporting recent contributions addressing the discovery problem in fog computing scenarios.

A. The use of cellular footprints for context inference

Different mechanisms have been studied in the literature to enable context-awareness. A trivial method would be the use of the GPS coordinates. However, it incurs a high energy cost while not being suitable for indoor environments. Another option would be to use the Wi-Fi footprint of a given place (consisting in the list of detected APs, their characteristics and their signal strength), which was shown to provide promising accuracy values. However, since it also uses the Wi-Fi interface for context inference, it does not solve the energy consumption issue that we intend to address. Finally, the use of the cellular footprint of a certain location (the set of observed cellular tower IDs and their corresponding signal strengths) as a context indicator has been widely-studied in the fields of localization [5], [6], efficient discovery of access points [7] and human places of interest [8]. This approach can achieve promising results in terms of accuracy, while not causing an additional energy cost, since cellular information is already continuously received by a mobile phone.

This is proved for instance in the study conducted in [7], where authors propose to exploit the correlation between surrounding information (such as the cell tower ID, the Bluetooth ID and the user speed) and previous Wi-Fi contacts in history data to estimate the remaining time until the next Wi-Fi AP encounter. The authors found that the information gains obtained from the cell ID approach are larger than the ones from Bluetooth IDs in addition to being the most energy-efficient. Authors in [5] leverage cellular information for localization purposes and their proposed system was able to estimate the position of a mobile user with a very small error (20m) using information from three cell towers (one connected and two neighboring). A dedicated application server is used to serve localization requests, being sent by the device every five seconds. In order to improve the location's precision, a continuous update of a cellular fingerprinting database is made in addition to considering the previous estimated location to predict the actual one. However, it is worth noting that more recently, most modern phones no longer provide programmatic access to neighboring cell information and only the serving cell information is available for use. That is why, the work conducted in [6] proposes the use of a single cell tower information (the associated one) to provide localization. The proposed solution is based on a Hidden Markov Model and it allows achieving accurate GSM localization with a 50m error approximately in urban areas, which is very promising, given the limited information used for making predictions in nearby locations, where cellular footprints are likely to overlap.

Similar to [6], we also consider the use of the serving cell information only in our proposed approach, as described in Section III.

B. Discovery in fog computing scenarios

Aligned with our efforts in the area of discovery in an edge/fog computing scenario and in addition to recent related efforts reported in our previous work [3], additional contributions have been made that we describe next. *Kinaara* is proposed in [9] as a framework for distributed discovery and allocation of mobile edge resources. Those resources are organized within proximal clusters managed by trusted mediator entities. Device discovery is facilitated by the use of a distributed indexing scheme allowing the organization of resources into a logical ring structure based on resource similarity. Although the use of such a similarity-based indexing yields fast discovery times, the paper does not mention how the mediator entities are actually discovered. *eDisco* is proposed in [10], where DNS SRV records are used to enable discovery of nearby edge servers on a client's network path. Those records need only to be added to the DNS zone of the entity in charge of the edge server's installation. A rather different but related scenario is addressed in [11], where the use of a fog-based MQTT server is considered to assist IoT devices in the Bluetooth Low Energy (BLE)-based discovery process. In fact, the fog server's role is to keep track of the BLE advertiser device trajectory and it controls the enabling/disabling of the BLE scanner's interface according to the advertiser's geolocation, transmitted through Wi-Fi to the broker. This allows obtaining a high discoverability rate and significant power savings. However, this paper puts the focus on increasing the energy-efficiency of the BLE scanning process, without considering the potential energy overhead at the BLE advertiser's side because of frequent location transmissions to the broker.

III. Cellular context-based fog node discovery

Fig. 1. Illustration of FN deployment in a city

As fog nodes start to be deployed in geo-distributed locations in a given city (an example deployment is shown in Fig 1), it is highly important for a mobile device to know where these nodes are located in order to take advantage of their resources to achieve enhanced application executions. Therefore, leveraging the wide deployment of cellular towers in cities nowadays, our objective is to be able to predict the presence of a specific FN, given the cellular footprint that is currently observed by a mobile device. A cellular footprint

here corresponds to the **(serving_cell_id, signal_strength)** pair.

In order to achieve this objective, two phases are needed. First, the previously observed cellular footprints are stored along with the information about the presence or absence of a FN while that footprint is seen. Such information is acquired from initial non-context-aware discovery executions. Then, the collected data is given as an input to a learning algorithm to predict the appropriate locations, in which the scan should be triggered. As a result, the scan for FNs should be enabled in the cases where the algorithm predicts the existence of a FN. On the other hand, in case the algorithm infers that the user is present at a non-FN location, the scan should be kept disabled in order to reduce the number of unnecessary scans, possibly affecting the device's energy consumption.

Since we are dealing with labeled data, where the prediction values are previously known, the use of a supervised classification algorithm is considered. More specifically, we select the K-nearest neighbors (KNN) and Decision Tree (DT) algorithms for our evaluation, since they do not assume a specific distribution for the data. We additionally evaluate the use of a probabilistic approach based on a Hidden Markov Model (HMM) to learn from historical transition probabilities from one state to the other (e.g. "none" $\rightarrow$ FN).

In the following, we provide an overview of how the three considered approaches have been used:

- *KNN*: The basic idea behind the original KNN algorithm is quite simple. In fact, for each new data entry, the distance to each of the historical data entries is calculated. Then, the data is sorted by decreasing order of the calculated distances and the top K entries are retrieved. Finally, the prediction corresponds to the majority class found within the top K elements. We further customize the KNN algorithm in the following two ways. First, in order to avoid extensive distance calculations to each historical data item each time a scan decision has to be made, we reduce the considered dataset to a smaller subset only containing entries having the same cell ID as the one in the new cellular footprint. Then, we calculate the distance in terms of signal strength, as follows:

$$distance = \frac{|ss1 - ss2|}{max(|ss1|, |ss2|)}$$

  where ss_i is the signal strength at time t_i.
  Second, in the cases when the current cell ID has not been previously seen in the historical data, no prediction occurs and the scan is triggered in order to avoid missing a FN, at the expense of potentially causing an energy overhead.
- *Decision Tree (DT)*: In our case, we use a classification tree where predictions can be made based on the cell ID and signal strength input variables. We initially split the dataset based on the cell ID variable, thus obtaining a subtree for each unique cell ID. Then, for each subtree, we further split the data entries according to the signal strength variable to obtain splits, which are as pure as possible. The Gini index [12] is used to evaluate the purity of the performed splits. To avoid potential overfitting, the splitting process stops when a given tree depth or when a minimum number of data entries is reached. The prediction then corresponds to the majority class found in the data entries associated to the considered tree node. Similar to what we did in the KNN approach, if the current cell ID is unknown, the scan is enabled.
- *HMM*: A Hidden Markov Model is a statistical model allowing predicting the most likely sequence of states $(s_1, s_2, ..., s_N)$, given an input sequence of observations $(o_1, o_2, ..., o_N)$, where N is the considered sequence length. In our case, the observations are the set of cellular footprints whereas the hidden states to be inferred from those observations are the presence/absence of a given FN. More specifically, the predictions can be made based on the following HMM characteristics: (i) *Emission probabilities* referring to the probability of a certain observation (cellular footprint) given a state (FN_ID / none); (ii) *transition probabilities* corresponding to the probability of transition from one state at time t to another state at (t+1) and (iii) *prior probabilities* corresponding to the general probability of occurrence of a certain state. The prediction in this case corresponds to the last item of the state sequence, i.e. s_N.

A qualitative comparison of the three considered models is provided in Table I.

IV. Performance Evaluation

In this section, we detail our experimentation approach and we then analyze the obtained results.

A. Experimentation approach

For the purposes of this work, we assume a small-scale fog node deployment in specific locations of the city, including our research lab, another university building (engineering school) as well as the city hall (all shown in Fig 2). This placement is justified by the fact that, in a real deployment, these locations could potentially be candidates for hosting fog-based applications such as smart building management solutions or enhanced entertainment services.

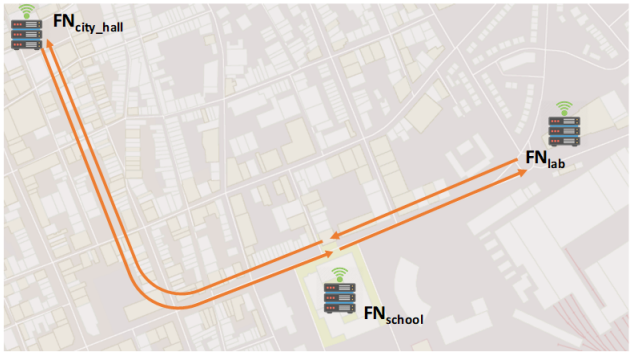

Fig. 2. Considered data collection scenario

In order to obtain the cellular context information corresponding to this city area, we used an Android-based mobile

TABLE I
QUALITATIVE COMPARISON OF THE 3 CONSIDERED MODELS

Algorithm	Pros	Cons
KNN	• No training needed • No assumption about underlying data distribution	• Distance is calculated to each item in the dataset
DT	• Simple to interpret	• Parameters needed such the max depth • Training phase needed to build the tree
HMM	• Capturing state transition sequences beneficial for time-series data	• Complexity associated with keeping track of the different probabilities and finding the most likely state sequence. • Long sequence means increased accuracy at the cost of delaying the initial prediction [6]

network monitoring application[1] , which logs cellular traces including the associated cell ID and its corresponding signal strength every minute (or whenever the serving cell changes). Then, we repeatedly move (at walking speed) following the path marked with the orange arrows in Fig 2 while keeping the monitoring application ON. A stop of a duration of 10 minutes is made at each FN location to account for the fact that users generally stay for a non-trivial amount of time in specific places of interest, as defined in [8]. This also ensures that we obtain enough representative data entries of a particular FN location.

The previous data collection experiment is repeated five times for two different network operators. The obtained data was further labeled manually in order to tag the trace entries corresponding to the fog node locations, since there is no real FN deployment in the city at this stage. The resulting data entries have the following structure: **(cell_id, signal_strength, FN_id)**, where FN_id corresponds to the FN identifier, if any; otherwise, it is set to "none".

The three approaches described in Section III were implemented in Python and their performance was evaluated using 5-fold cross-validation. Different values for the algorithm parameters were tested and the ones yielding the best results (i.e. less overfitting) were chosen, as shown in Table II. The discovery ratio as well as the saving ratio are provided next as performance evaluation metrics.

TABLE II
ALGORITHM PARAMETERS

Algorithm	Parameters
KNN	• k = 5
DT	• Max_depth = 3 • Min_entries = 3
HMM	• Emission probabilities, Transition probabilities, Prior probabilities were determined based on their actual corresponding fractions in the training data • Sequence length = 5

B. Obtained results

1) Discovery ratio: An important metric to consider in our scenario is the discovery ratio (more generally referred to as true positive rate) and it indicates the algorithm's ability to successfully discover a FN out of the total opportunities in which the user was indeed present at a FN location. It is defined as follows:

$$Discovery\ ratio = \frac{Nb.\ of\ successful\ discoveries}{Nb.\ of\ total\ discovery\ opportunities}$$

As it can be seen in Fig 3, there is only a slight difference in the performance of the KNN and DT algorithms in terms of discovery ratios. They were both able to successfully discover FNs in ∼85% (resp. ∼76%) of the time for the first (resp. second) operator. It is worth noting that both algorithms were not able to achieve a better performance because of the relatively small size of the considered datasets in addition to the fact that measurements were carried out in nearby locations, where it is difficult to make predictions based on limited information (only the serving cell ID and the signal

[1]https://play.google.com/store/apps/details?id=com.signalmonitoring.gsmsignalmonitoring&hl=en

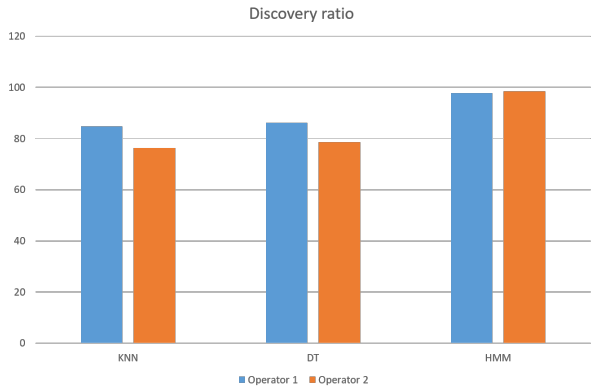

Fig. 3. Discovery ratio

strength). Moreover, signal fluctuations also contribute to the uncertainty causing erroneous predictions. On the other hand, HMM achieves a discovery ratio of up to 98 % even in presence of such limited and uncertain information. This is mainly due to the fact that it leverages the whole sequence of state transitions to make predictions.

2) Saving ratio: Since one of our goals is to reduce the number of unnecessary scans in order to achieve energy savings, we consider the saving ratio (more generally referred to as true negative rate) as a second performance indicator. In fact, we consider a saving was achieved when the algorithm predicts "none" in a non-covered location, whereas saving opportunities correspond to the number of actual occurrences of such non-covered locations. The ratio of both values represents the saving ratio, as defined in the following:

$$Saving\ ratio = \frac{Nb.\ of\ savings\ achieved}{Nb.\ of\ actual\ saving\ opportunities}$$

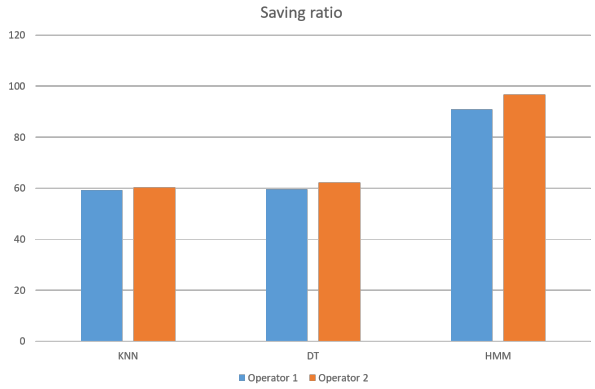

Fig. 4. Saving ratio

As depicted in Fig 4 and similarly to the discovery ratio case, KNN and DT have similar saving patterns of up to 62%. Such a behavior can be attributed to the aforementioned factors affecting the discovery ratio. On the other hand, HMM can achieve considerable savings reaching 96%. Although in a real-world scenario, the mobile device may be resorting to a cloud-based execution in the case of a non-covered fog area, thus consuming energy to do so, it is still preferable to keep the number of unnecessary scans as low as possible. This not only helps conserving the device's battery, but it also contributes to an efficient use of the wireless channel.

C. Discussion

1) User - FN contact ratio: Sections IV-B1 and IV-B2 have focused on the models' performance mostly from a statistical perspective, considering that a prediction is made each time a cellular measurement is available (i.e. every minute or when the serving cell changes). However, in certain time steps, predictions are not required. In fact, if the user has already detected a FN, then discovery is not needed as long as the user is still connected to it. Therefore, the predictions that matter the most are the ones leading to an initial successful discovery of a FN. If such a successful discovery is delayed due to an incorrect prediction (see first two predictions in the example in Fig 5), then this may lead to shorter user – FN contact duration, which in turn increases the risks of potential SLA violations due to the inability to use the FNs' resources earlier[2].

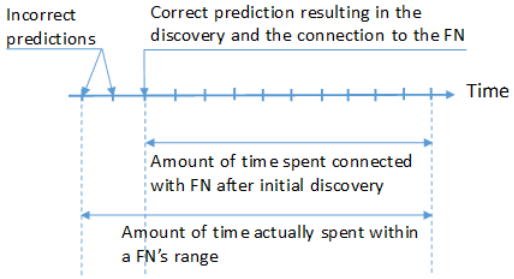

Fig. 5. Illustration of the User-FN contact ratio

Consequently, in order to evaluate the impact of using cellular context-based predictions for scan triggering, we consider another metric, the "user – FN contact (UFC) ratio", illustrated in Fig 5 and defined as the fraction of time the user has spent connected to a FN after the initial successful discovery divided by the total time spent within a FN's coverage. The closer the UFC ratio to 1, the better it is, in order to fully benefit from the FN's resources.

Fig 6 and Fig 7 plot the UFC ratio (shown in percentage) for the two considered operators and for the three considered FNs of Fig 2. As it can be seen, all three algorithms guarantee a minimum UFC ratio of 80% (resp. 92%) per FN (on average, resp.). It is also worth noting that even using simple algorithms such as KNN, a maximum of a 100% UFC rate can be achieved for certain fog nodes (see FN_{lab} in Fig 7), mainly the ones characterized with relatively stable cellular patterns (as opposed to FN_{school} where we observed many fluctuations for the second operator).

2) Impact of mobile FNs on false triggering: In this section, we consider the case of mobile FNs (cars, buses, etc…) that may show up during the user's mobility. In fact, such mobile FNs might be detected by the discovery scheme in locations

[2]A cloud-level execution penalty may occur in this case.

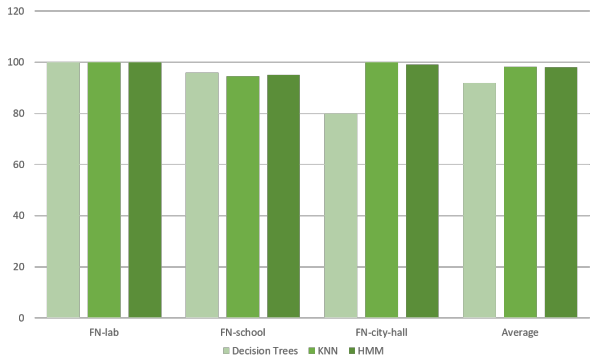

Fig. 6. User-Fog contact ratio (Operator 1)

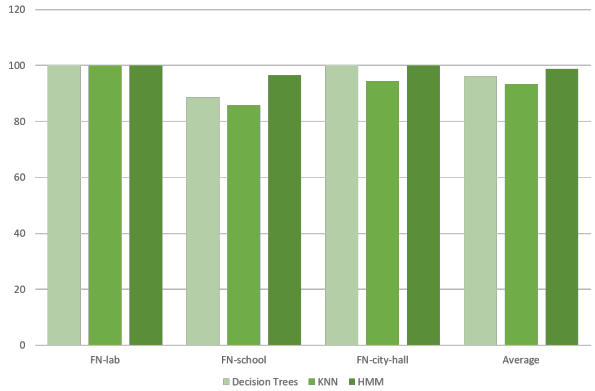

Fig. 7. User-Fog contact ratio (Operator 2)

where the cellular footprint is not generally representative of a FN (i.e. locations where saving opportunities can occur) and therefore this information will be added to the data to be used for future predictions. However, we note that this is not likely to cause false triggering of scans in such locations, since the proposed methods are based on the use of maximum likelihoods to make predictions, whereas mobile FNs will be seen infrequently in the same location by the same user. While this can ensure robustness of the proposed approach to FN mobility in terms of false triggering, it does not solve the problem of finding the right context to discover a mobile FN because of the uncertainty caused by the FN's mobility. We aim to address this specific issue in a future work in order to increase the mobile user - mobile FN contact ratio.

V. Conclusion

In this paper, we explored the use of surrounding cellular information as a context indicator for Wi-Fi based fog node discovery. The goal is to predict the presence or absence of a fog node based on the current cellular footprint in order to trigger or disable the discovery accordingly. We specifically compared three different prediction methods based on the use of KNN, decision trees and HMM. We found that the HMM model obtains the highest performance in terms of discovery and saving ratios, at the cost of more complexity inherent to the model's functioning. On the other hand, the use of decision trees and the simpler KNN algorithm achieve lower performance. We believe this can be overcome by dynamically learning from new cellular context values on the go, in addition to increasing the sampling frequency (compared to the considered 1 sample/minute). We also note that the proposed context-awareness mechanism is targeted towards mobile devices having access to an active SIM card. Alternative context-awareness mechanisms (GPS-based or Wi-Fi-based) may be envisioned for other devices, especially if energy consumption is not a concern. Finally, our future work will consider the fact that mobility might also affect FNs (e.g. an in-vehicle FN) and as such, we aim to study suitable mechanisms to maximize the user - FN contact ratio in the case of such FN mobility.

Acknowledgment

This work was partially supported by the H2020 EU mF2C Project ref. 730929 and by the Spanish Ministry of Economy and Competitiveness and by the European Regional Development Fund, under contract TEC2015-66220-R (MINECO/FEDER).

References

[1] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the first edition of the MCC workshop on Mobile cloud computing - MCC '12*, p. 13, ACM, 2012.

[2] X. Masip-Bruin, E. Marín-Tordera, G. Tashakor, A. Jukan, and G. J. Ren, "Foggy clouds and cloudy fogs: A real need for coordinated management of fog-to-cloud computing systems," *IEEE Wireless Communications*, vol. 23, no. 5, pp. 120–128, 2016.

[3] Z. Rejiba, X. Masip-Bruin, A. Jurnet, E. Marín-Tordera, and G.-J. Ren, "F2C-Aware: Enabling Discovery in Wi-Fi-Powered Fog-to-Cloud (F2C) Systems," in *2018 6th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud)*, pp. 113–116, IEEE, 2018.

[4] Z. Rejiba, X. Masip-Bruin, and E. Marín-Tordera, "Analyzing the deployment challenges of beacon stuffing as a discovery enabler in fog-to-cloud systems," in *2018 European Conference on Networks and Communications (EuCNC)*, pp. 1–276, June 2018.

[5] G. Aloi, G. Caliciuri, V. Loscrí, and P. Pace, "Accurate and energy-efficient localization system for smartphones: A feasible implementation," in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC*, pp. 3478–3481, IEEE, 2013.

[6] M. Ibrahim and M. Youssef, "A hidden markov model for localization using low-end GSM cell phones," in *Communications (ICC), 2011 IEEE International Conference on*, pp. 1–5, IEEE, 2011.

[7] J. Jeong, J. Lee, Y. Kim, J. W. Lee, and S. Chong, "Impact of surrounding information on Wi-Fi sensing efficiency," in *International Conference on ICT Convergence*, pp. 410–415, IEEE, 2013.

[8] K. Yadav, V. Naik, A. Kumar, and P. Jassal, "PlaceMap: Discovering human places of interest using low-energy location interfaces on mobile phones," in *ACM DEV 2014 - Proceedings of the 2014 Annual Symposium on Computing for Development*, vol. 5, pp. 93–102, ACM, 2014.

[9] A. Salem, T. Salonidis, N. Desai, and T. Nadeem, "Kinaara: Distributed discovery and allocation of mobile edge resources," in *Mobile Ad Hoc and Sensor Systems (MASS), 2017 IEEE 14th International Conference on*, pp. 153–161, IEEE, 2017.

[10] A. Zavodovski, N. Mohan, and J. Kangasharju, "eDisco: Discovering Edge Nodes Along the Path," *arXiv preprint arXiv:1805.01725*, 2018.

[11] R. Venanzi, B. Kantarci, L. Foschini, and P. Bellavista, "MQTT-Driven Node Discovery for Integrated IoT-Fog Settings Revisited: The Impact of Advertiser Dynamicity," in *Service-Oriented System Engineering (SOSE), 2018 IEEE Symposium on*, pp. 31–39, IEEE, 2018.

[12] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, "Classification and regression trees," 1984.