

# EVALUATION OF THE EMPLOYMENT OF UAVs AS FOG NODES

Rodrigo A. C. da Silva, Nelson L. S. da Fonseca, and Raouf Boutaba

## ABSTRACT

In fog computing, processing, network, and storage resources are placed close to end users to ensure low latency in comparison to the latency experienced when accessing the cloud. One limitation of this solution, however, is that fog nodes are usually fixed, whereas demands are variable over time at all locations, which may result in either under- or overprovisioning. This limitation incurs high CAPEX and OPEX costs to cope with user demands. One approach to address this problem is to employ mobile fog nodes dispatched to various locations to cope with the variability in resource demand. This article aims to evaluate the use of unmanned aerial vehicles (UAVs) equipped with processing elements as an alternative to fixed nodes in a fog infrastructure to cope with the variable workload in a metropolitan area. Although previous approaches considered UAVs as part of the network, they did not deploy multiple fog nodes mounted on battery-constrained UAVs. Specifically, we propose in this article a solution to the fog node location problem considering both fixed and mobile nodes to evaluate potential replacements of fixed servers with UAVs. Experimental evaluation of the problem using data generated by real mobile users shows that UAVs can indeed replace parts of the fixed fog infrastructure.

## INTRODUCTION

Cloud computing has facilitated the deployment of numerous online services as well as services to augment the capabilities of user devices [1]. Cloud computing relies on large data centers that host computing, network, and storage resources accessed on demand through the Internet. However, they are typically located in remote areas, making a variety of applications with strict latency requirements infeasible. One solution to alleviate this limitation is the employment of fog computing, an architecture to provide computing, storage, and networking capabilities anywhere along the continuum between the cloud and the end users [2]. Fog nodes are usually deployed as fixed nodes in different locations to support the computational needs of local users [3] and support the strict latency requirements of applications. However, nodes in a fixed infrastructure may need to be over-dimensioned to cope with variable processing demands.

Unmanned aerial vehicles (UAVs) have been employed for military applications, surveillance,

and traffic analysis, to name a few. More recently, UAVs have been considered for integration into cellular networks to serve as base stations [4, 5], allowing providers to expand their coverage area in case of occasional demands or failure of the terrestrial infrastructure. The employment of UAVs as fog nodes is still its infancy. A few studies have considered such use [6, 7], but they have not modeled the limited autonomy of UAVs, which restricts their usability in real infrastructures. A detailed battery model for UAVs needs to be accounted for in the evaluation of their potentiality as processing nodes.

This article aims to assess the advantages of employing UAVs as fog nodes for dealing with variable workload demands generated by mobile users. From this perspective, this article studies the problem of where to locate UAVs as fog nodes (the fog node location problem) in a metropolitan area with the aim of offering cloud services at the edge. In contrast to previous approaches [8, 9], this article considers both fixed nodes and mobile UAV nodes. Fixed nodes are always available due to continuous energy supply, but once deployed, they cannot be migrated to another location easily. On the other hand, UAV nodes can fly between different locations to augment the processing capacity of fixed fog nodes, especially to process workload in excess of the capacity of fixed nodes. However, UAV nodes are mobile and operate on batteries, which limit the length of time they can process user workload. The consideration of both fixed and mobile nodes enables the deployment of an infrastructure capable of handling the variabilities in workload demand; this helps reduce the underutilization of over-dimensioned fixed nodes for processing eventual peak demands.

An algorithm called UAV Fog Node Location (UFL) is introduced in this article to determine the best combination of fixed nodes and UAVs in an infrastructure. The UFL algorithm initially finds an exact solution to the fog node location problem considering only fixed nodes, and then attempts to replace underutilized servers in fixed nodes by UAVs, which can change their location to cope with processing demand at different locations. Simulations considering diverse demand patterns across a metropolitan area as well as real UAV characteristics such as cost, battery capacity, and processing capabilities [10] are employed to address the question: Are UAVs worth adopting to replace fixed nodes in a fog infrastructure? The wireless communication channel is considered

ideal, so results are actually a bound on the value of UAVs used as processing nodes. Results show that the current cost of UAVs is a limiting factor to their usage as fog nodes. In the future, however, and assuming that costs continue to decrease, UAVs could provide interesting solutions for optimal infrastructure dimensioning. Results show that when UAV costs are the same as those of fixed nodes, UAV deployment is indeed advantageous. In this case, the coverage of several locations using UAVs reduces CAPEX and provides more flexibility in coping with unexpected and timely increases in processing demands.

The remainder of this article is structured as follows. First, previous work is reviewed. The problem in this article is presented, and the experimental setting and numerical results are then discussed. Finally, conclusions are drawn and future work suggested.

## UAVS AS NETWORK EDGE ELEMENTS

UAVs can be used to extend the communication network infrastructure as processing nodes at the edge of the network. One common question in this usage is where to position UAVs to achieve some target objectives. The facility location problem is a design problem in which a set of candidate locations that can host facilities is given as input, with the solution to the problem being the subset of locations for the deployment of these facilities.

Optimizing the location of edge facilities has been explored in previous work considering objectives such as the cost of facilities [8, 11] and energy consumption [9]. In a cost-based formulation [8], a mixed-integer linear programming (MILP) model was used to obtain the best fog deployment to serve the largest workload at the lowest cost. Another approach [11] considered cloudlets in the deployment and proposed an MILP model to achieve a trade-off between deployment cost and latency experienced by end users. The results obtained by these studies concluded that infrastructure providers can reduce the deployment cost if some degradation in service can be tolerated [8, 11]. The solution in [9] optimizes the number of users served; the reduction in energy spent by mobile devices accessing the infrastructure is considered to be a secondary objective. Fog nodes are deployed in locations where they bring possible opportunities for maximum offloading with minimal energy. A limitation in these approaches is that they consider only fixed fog nodes, but actually the demands on fog nodes are variable. Consequently, a large number of servers are underutilized most of the time in these nodes.

UAVs are expected to be part of future 6G networks, serving as base stations (BSs) [4, 5], processing nodes [6, 7], and even as users [12]. The authors of [12] considered UAVs to be users of terrestrial BSs and optimized the trajectory followed by UAVs to minimize the task completion time while offloading workload to the ground BSs. The SkyCore architecture for LTE networks [4] proposes the use of unthetered UAVs as BSs, deployed jointly with ground stations. In the three-layer architecture envisioned in [5], UAVs provide coverage to ground users with resources located in satellites, known as CubeSats, furnishing connectivity for ground users by connecting them to satellites.

The UAVFog architecture considered the use of drones as fog nodes [6]. The authors discussed the integration of UAVs with the cloud in monitoring applications, but they did not discuss the battery limitation of drones. The authors in [7] considered the employment of UAVs equipped with servers as processing nodes at the edge hovering at different areas to provide a processing capacity for mobile users. The study showed that the employment of UAVs significantly increased the number of users served in relation to those served in an infrastructure with only fixed nodes. One major drawback of this solution is the consideration of unlimited flight time. Although it is possible to extend UAV flight time by using solar power or tethered drones, such solutions cannot be generalized, since not all aircraft can carry the solar panels. Moreover, tethered drones impose extremely limited mobility.

This article aims to investigate open questions in previous work. Previous solutions in [8, 9, 11] either suggest using underutilized servers or allow degradation in quality of service (QoS). In an attempt to overcome these limitations, we have investigated the employment of mobile nodes to extend the capacity of fixed servers, thus reducing resource underutilization. Previous work on the employment of UAVs in cellular networks [4, 5, 12] did not explore the possibility of UAVs as processing nodes.

Table 1 summarizes the work reported in this section, classifying each article according to the problem studied. Our present article considers the employment of UAVs as processing nodes and their battery capacity. Energy consumption is accounted for both processing and flying, constituting a more realistic scenario than those evaluated in previous work.

## FOG NODE LOCATION PROBLEM

### SYSTEM MODEL

We consider a system in which mobile users in a metropolitan area request services at different locations, and the processing demand at these locations is a function of user mobility. Applications such as augmented reality and traffic navigation have strict latency requirements and cannot be processed in the cloud, and users need to connect to a nearby fog node to offload the processing of these applications. If there is no fog node available, requests are rejected (blocked) since they cannot be migrated to more distant nodes due to latency requirements.

A fog node is a small facility that has processing, storage, and networking capabilities. It can process workload offloaded by end users without the need to send it to the cloud through the Internet, thus considerably reducing the response time of applications. Fog nodes serve users in their coverage area, and nodes can have more than one compute server. However, not all servers of a node are continuously needed since the processing demand varies over time.

UAVs can travel from one node to the other to increase the processing capacity of a destination node. UAVs can land to process the workload instead of just hovering. When on the ground, UAVs have greater autonomy, since the energy consumption of a UAV is much lower than when

UAVs can be used to extend the communication network infrastructure as processing nodes at the edge of the network. One common question in these usages is where to position UAVs to achieve some target objectives. The facility location problem is a design problem in which a set of candidate locations that can host facilities is given as input, with the solution to the problem being the subset of locations for the deployment of these facilities.

The fog node location problem consists of deciding on the locations where fog nodes should be deployed. The main input to this problem is the set of potential locations for hosting fog nodes, the workload demand, and the available budget for acquiring fixed and UAV fog nodes. The output is the set of locations selected for the deployment of nodes as well as the number of servers at each node.

Reference	Problem	Criteria	Solution	UAV	UAV battery
[8]	Fog node location	Workload acceptance and deployment cost	MILP	No	–
[9]	Fog node location	Workload acceptance and energy consumption	MILP and heuristic	No	–
[11]	Cloudlet location	Deployment cost and delay	MILP	No	–
[12]	Trajectory optimization	Completion time	Heuristic	Yes	No
[4]	UAV base stations	User coverage	Architecture and testbed	Yes	Yes
[5]	UAV base stations	User coverage	Architecture	Yes	No
[6]	UAV fog nodes	User coverage	Architecture	Yes	No
[7]	Mobile server location	Workload acceptance	Heuristic	Yes	No
Our article	Fog node location	Workload acceptance and deployment cost	Heuristic	Yes	Yes

TABLE 1. Comparison of related work.

hovering [10], with energy consumed only by communication and processing.

UAVs can be in one of four different states: turned off, standby, processing, and flying. UAVs are initially turned off, with their battery fully charged. When a UAV is turned off, it does not consume energy. When its service is needed, a UAV starts its operation and remains on until service is no longer required. In the standby state, a UAV is on the ground but not processing, and its energy consumption is fixed; moreover, it can be quickly switched to the processing or flying state. In the processing state, the UAV is also on the ground, but it consumes energy for processing and data transmission. Finally, in the flying state, the UAV is moving between different locations; flights are allowed only to complement the capacity of a fog node at the destination. The flying state is the one that consumes the largest amount of energy, with consumption depending on the distance traveled and the speed of traveling, both horizontally and vertically. The sum of the energy consumed by all operations must be lower than the UAV battery energy capacity, which demands a precise location plan to extend the drone operational time. Other sources of consumption such as environmental factors are not considered in this article.

### PROBLEM STATEMENT

The fog node location problem consists of deciding on the locations where fog nodes should be deployed. The main input to this problem is the set of potential locations for hosting fog nodes, the workload demand, and the available budget for acquiring fixed and UAV fog nodes. The output is the set of locations selected for the deployment of nodes as well as the number of servers at each node. Additionally, the number of UAVs and their flight plan should be determined. The primary goal is to process the maximum possible amount of workload, and the secondary goal is to reduce the infrastructure cost.

The location problem can be formulated as a mathematical programming model. In this model, goals (objectives) and constraints are given by linear equations, and decision variables are binary, integer, and real numbers. For instance, whether a UAV is active at a time interval is a binary deci-

sion, the number of fixed servers in a location is an integer variable, and the workload served in a location at any time interval is a real variable. Therefore, the problem can be formulated as a bi-criteria mixed-integer linear programming model. In the formulation, time is discretized, and the workload at each location is accounted for each time slot. The capacity and cost of fixed and UAV nodes are given. The energy spent in all states as well as the UAV battery capacity are provided as input to the optimization problem. The output is the workload processed at each location for every time slot, the number of fixed servers at each location, the number of UAVs employed, the state, and the location of all UAVs at every time slot.

Such deployment planning is a network design problem, and as such is typically solved offline. However, solving this problem optimally using existing solvers does not scale to large problem instances. The modeling of potential flight routes, the location of UAVs at every time interval, and the UAVs' activity/inactivity periods lead to an exponential growth in the number of constraints. To circumvent these limitations, a heuristic algorithm is proposed next.

### UAV FOG NODE LOCATION ALGORITHM

We propose a heuristic algorithm called the UAV Fog Node Location (UFL) algorithm. Figure 1a shows its flowchart. The algorithm starts by solving the formulation with only fixed nodes, and the pre-defined budget limiting the number of servers and nodes that can be deployed. Based on the solution obtained, the algorithm identifies servers that can potentially be replaced by UAVs; these servers are typically underutilized and deployed only to deal with peak demands. The UFL algorithm then attempts to use UAVs to cover several locations at different times to reduce the deployment cost. The algorithm considers the ratio between the cost of UAVs and the cost of fixed servers.

The first step of the algorithm considers only fixed nodes with the result obtained using an optimization solver (Step 1). The next step is the identification of servers to be replaced (Step 2). For each fog node, the algorithm identifies if a server can be replaced by a UAV, a situation that arises if a server is not processing requests for all time intervals, the

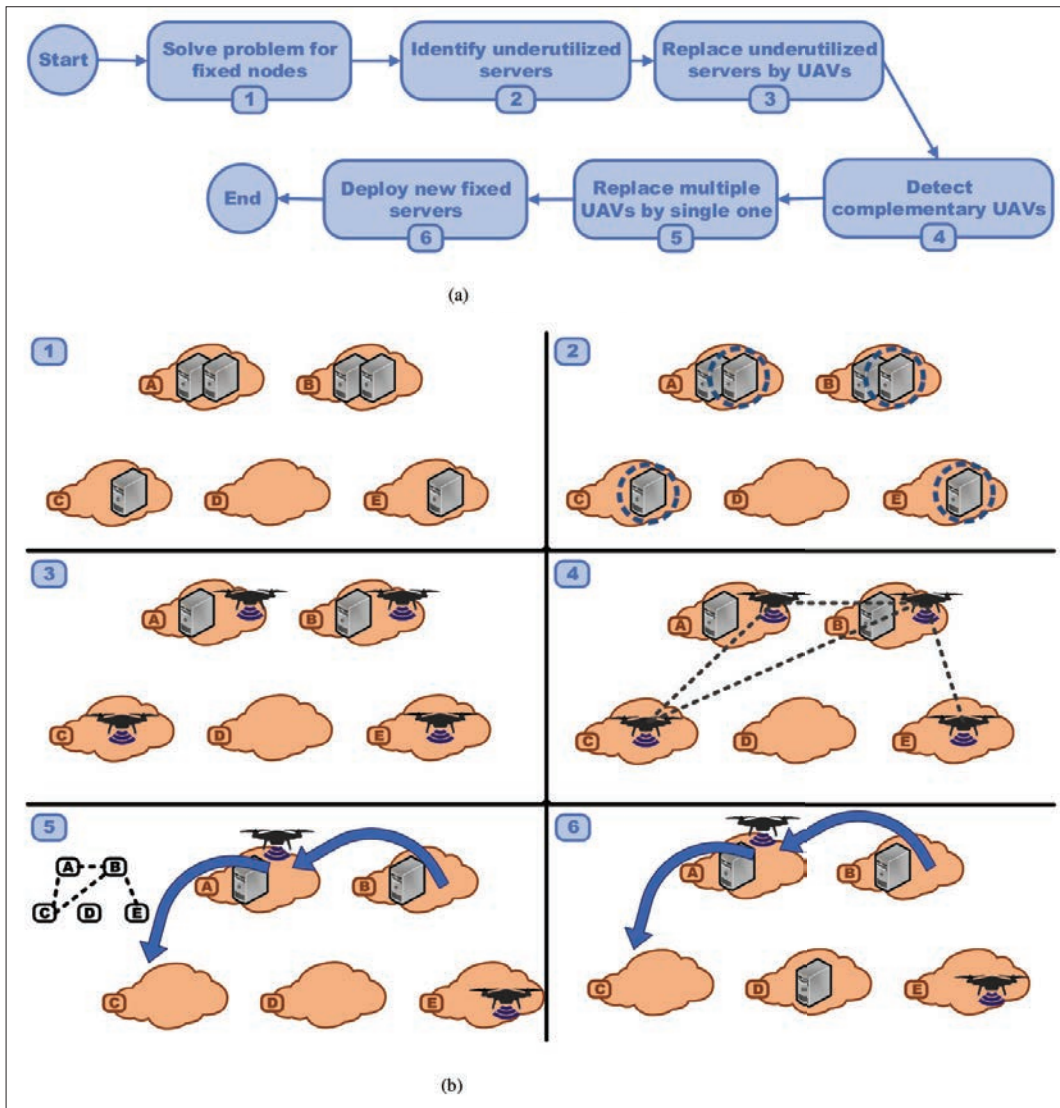


FIGURE 1. UFL algorithm: a) algorithm flowchart; b) example of execution.

UAV processing capacity is greater than the offered workload, and the energy that will be consumed by the UAV is less than its available battery capacity. The energy needed is the sum of the energy spent in processing and that in standby in periods associated with the potential replacement. The identified servers are then replaced by UAVs (Step 3).

To reduce the infrastructure cost (secondary objective), all pairs of UAVs are considered to be replaced by a single UAV (Step 4). Two UAVs can be replaced by a single one if three conditions are fulfilled. First, the two UAVs must be in processing state in different time periods. Second, the time for traveling between the two locations is less than the time elapsed between the end of the processing at the node from which the UAV departs and the beginning of processing at the destination node. Third, the UAV battery should be sufficient to support full operation, including the flight between the fixed fog nodes. If all conditions are met and after serving the workload at a location, a UAV can fly to another location to serve the workload at the new location.

The algorithm evaluates a potential reduction in the number of UAVs (Step 5). Such an evaluation is carried out by considering a graph in which

each UAV is a vertex, and each potential pair of locations for replacement is an edge of the graph. Then the algorithm finds maximal cliques, which determines the minimum number of UAVs to be deployed. If the solution still leaves a backlog of unprocessed workload and the number of UAVs has been reduced in the last step, the unused budget can be employed to further reduce the unserved workload (Step 6).

Figure 1b exemplifies the steps involved in planning fog nodes in five locations. The budget comprises six servers, and the solution obtained in Step 1 indicates fog nodes in locations A and B with two servers each due to their larger processing demand; other locations have a low processing demand, with only one server in locations C and E. Four fixed servers are identified as being underused in Step 2, and are then replaced by UAVs (Step 3). In Step 4, the algorithm detects the pairs of servers that have complementary processing demands in time, and that the battery of a single UAV is adequate to support the operation in both locations; the dashed lines indicate these pairs. Step 5 shows the graph of UAVs, which contains two maximal cliques. Locations A, B, and C can be served by a single UAV; that is, during a discretized time

Based on the solution obtained, the algorithm identifies servers that can potentially be replaced by UAVs; these servers are typically underutilized and deployed only to deal with peak demands.

The UFL algorithm then attempts to use UAVs to cover several locations at different times to reduce the deployment cost. The algorithm considers the ratio between the cost of UAVs and the cost of fixed servers.

$$\frac{\text{number of UAVs required by [7]} - \text{number of UAVs required by UFL}}{\text{number of UAVs required by UFL}} \times 100$$

Equation for mean percentage difference between two schemes.

Operation	Energy consumption
Fly horizontally	245.2815 W
Fly vertically up	$(-16.9396 H^2 + 216.6944 H - 157.9473) J$
Fly vertically down	$(4.6817 H^2 - 11.9708 H + 135.3118) J$
Stand-by state	8.2637 W
Processing state	15.7637 W
Operation	Speed
Horizontal	10 m/s
Vertical up/down	1 m/s
Variable parameters	Values
Battery capacity	4500 mAh/14.8 V 34200 mAh/22.8 V
Processing capacity	50 % and 100 % of a fixed server
UAV price	1, 2, 3, and 4 times the price of a fixed server

TABLE 2. Parameters adopted in the UAV simulations. H refers to the vertical distance traveled, either upward or downward.

interval, a UAV can either process the workload in one location or fly to another location to provide service at the destination. Since one UAV has replaced three fixed servers, an extra server can be deployed in location D, thus increasing the overall workload served (Step 6). The final solution has three fixed servers and two UAVs, thus employing fewer devices than the initial solution.

The UFL algorithm, as presented here, considers that any fixed server can be replaced by a UAV server. If the price of a UAV is greater than that of a fixed server, the solution obtained may not always be worth adopting. As a consequence, alternative solutions should be obtained which consider different ratios between the cost of a UAV node and that of a fixed node. Such an analysis facilitates long-term planning for the evaluation of the infrastructure, thus helping avoid unnecessary expenses in the deployment of the original infrastructure.

## EVALUATION OF THE UFL ALGORITHM

To answer the question of whether UAVs are worth adopting for replacing fixed fog nodes, extensive simulations of the UFL algorithm involving realistic scenarios were carried out.

## SCENARIO AND DATA DESCRIPTION

The parameter values defining the scenarios in the simulations are summarized in Table 2. The UAVs considered are multi-rotor drones that have the capacity to land in limited spaces. The characteristics of the simulated UAVs are based on real drones described in previous work [4, 10]. In [10], the authors measured the energy consumption of an Intel AeroReady to Fly Drone and derived

models to estimate the energy consumption for diverse operations. These models are used to simulate the energy consumption in vertical and horizontal flights, as well as the energy consumed in the standby state. The energy for the processing state is calculated as the energy in standby plus the energy spent by a Jetson TX2, a typical onboard computer for drones. The speed of the drone was based on the Intel AeroReady to Fly Drone. The energy spent during landing and take-off (vertical operations) is taken into account in the computation of the energy consumed during a flight.

Since the UAV used in [10] does not have a powerful battery, the present evaluation also considers different battery models [4]. Moreover, two other parameters were varied as a function of the fixed servers: the UAV processing capacity and UAV price.

The locations and the workload demands were based on real data made publicly available by Telecom Italia [13]. This data set was collected using call detail records (CDR) of mobile users between November and December 2013 in the metropolitan region of Milan, Italy. This article uses the CDR with information about Internet access to model the workload, since these records include various applications. The geographical area of the data set was represented as a  $100 \times 100$  grid, with each cell having its own CDR information. The dataset contains CDRs aggregated in intervals of 10-minute duration.

The cells identified in the dataset [13] cannot be used directly as the set of locations. Actually, users in the data set can request services from a BS that can be located in a different cell. Since the location of BSs was not included in the dataset [13], data from the OpenCellId project [14] were used to obtain the locations. The OpenCellId is a public database with location information of BSs worldwide collected by mobile users. The location of the BSs was obtained by filtering the existing BSs for the same period available in the Milan dataset [13]. The workload of each cell was mapped to the closest BS, as in [15]. In the case of multiple BSs in a cell, the workload is equally balanced among the BSs.

After processing the two databases, the set of locations is formed by the BSs from the OpenCellId project, with the workload for each location taken from the Milan dataset [13]. The geolocation of BSs is used to calculate the distance between locations, both horizontal and vertical. These data were used to calculate the time a UAV would need to travel and consequently, the energy required for these trips. The capacity of the server was 500 processing units per time interval, and the duration of each interval was 10 minutes, which results in 144 intervals in 24 hours. Time is discretized in intervals of 10-minute duration, which allows capturing users' mobility in a metropolitan area since users typically take more than 10 minutes to commute from one location to another. Moreover, 1150 locations were considered.

## NUMERICAL EVALUATION

The UFL algorithm was coded in Python, and its first step (bi-criteria formulation for the deployment of fixed nodes only) was solved using the Gurobi Optimizer solver. The results produced by the UFL algorithm were compared to those obtained by the solver. Two metrics were evaluat-

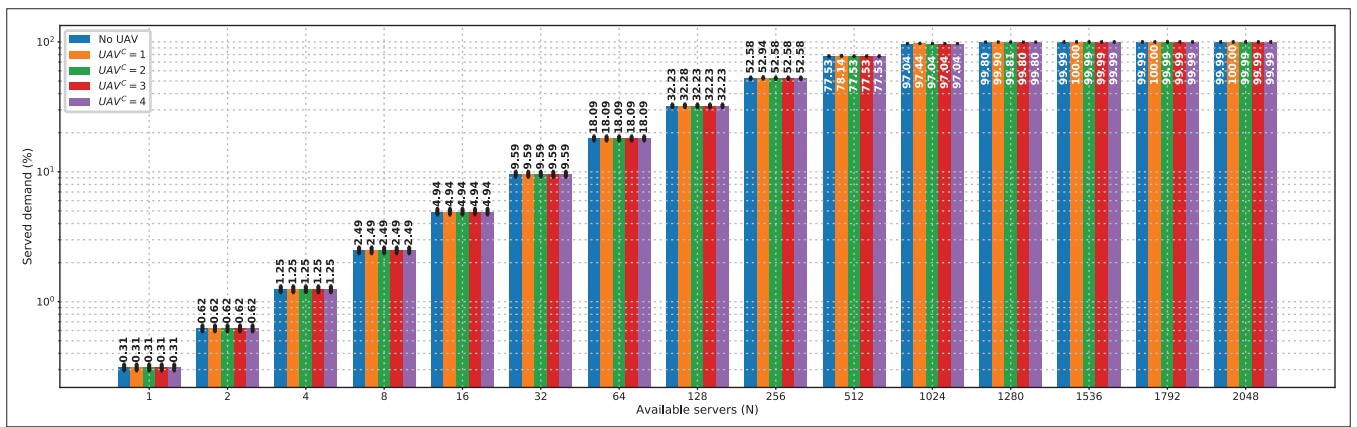


FIGURE 2. Acceptance ratio for the 34,200 mAh/22.8 V battery and  $UAV^P = 1$ .

ed related to the two objectives of the problem: the acceptance ratio of workload and the number of devices deployed (servers and UAVs). The first metric is the ratio between the workload served and the total workload requested by the end users. The second metric is the number of servers used for the solution using only fixed nodes, and the number of servers and UAVs employed computed by the UFL algorithm. Sixty executions were carried out to derive each value with a 95 percent confidence interval. The number of available servers for deployment ( $N$ ) was varied from 1 to 2048.  $UAV^C$  denotes the ratio between the cost of a UAV and the cost of a fixed server. Similarly,  $UAV^P$  is the ratio between the processing capacity of a UAV server and that of a fixed server.

The acceptance ratio using UAVs with the most powerful battery and the same capacity as a fixed server is shown in Fig. 2. The acceptance ratio increases until  $N = 1280$ , when servers are sufficient to deal with all the demand. The demand served depends predominantly on the fixed infrastructure capacity due to the limited autonomy of UAVs to stay powered for long periods. Nevertheless, improvements were noticed when UAVs and fixed servers have the same cost and  $N \geq 128$  since UAVs could be widely deployed. In these cases, UAVs improved the acceptance of workload and, in some cases ( $N \geq 1536$ ), provided 100 percent acceptance of workload. Higher costs of UAVs limited their number considerably, and as a consequence, the workload acceptance did not change in relation to deployment with only fixed nodes.

The results described below show the impact of the cost of UAVs, UAV processing capacity, and their autonomy on the deployment. Variations in these parameters do not make significant changes in the workload acceptance when compared to those already shown (Fig. 2). This is due to the fact that the workload acceptance is predominantly optimized by the fixed servers, with only slight improvements by the effect of using UAVs. In line with these results, only numbers of UAVs and servers are considered in the following experiments.

The number of employed devices is shown in Fig. 3 for an optimal deployment with only fixed nodes as well as for hybrid deployments (fixed servers and UAVs) obtained by the use of the UFL algorithm. Figure 3a shows the results for the greatest battery capacity, with UAV cost and processing capacity equal to those of fixed servers. For  $N < 128$ , almost all fixed servers in fog nodes were

heavily used for long periods of time, so replacing servers with UAVs was not possible. When a larger number of devices is available for deployment, a large number of fixed servers is replaced by UAVs, which shows that despite the large number of locations (1150), only about 200 fixed servers could not be replaced by aerial servers, that is, an infrastructure with only 20 percent of the locations being fixed nodes and UAVs serving the remaining 80 percent of the locations.

Nowadays, a UAV's cost is three to four times the cost of a traditional fixed server, and under these circumstances, the employment of several UAVs is not advantageous. Figure 3b shows the results for UAVs four times more expensive than a fixed server. For  $N \leq 1280$ , the average number of employed UAVs is very close to zero. This low number of flying servers is due to the fact that using the same UAV to serve two different locations is not always possible because of the required flight time between the locations, which led to quickly drain of the UAV battery. As shown, the UAV price is decisive in being considered in large deployments.

The results considering UAV servers having 50 percent of the capacity of a fixed server are shown in Fig. 3c. The greater the capacity, the larger the number of servers replaced by UAVs, since UAVs with limited processing capacity cannot always deal with the peak demands supported by a fixed server. The increase in the processing capacity and the increase in the number of UAVs is not linear: increasing the UAV processing capacity from 50 to 100 percent leads to an increase of less than 10 percent in the number of UAVs for  $N > 1280$ . This is explained by the pattern of the frequency of peak demands. Servers replaced by UAVs are seldom used; therefore, they deal with rather sporadic demands. Given these low demands, UAVs with powerful computers are not required.

The final analysis concerns the battery capacity. Figure 3d shows the results for the UAVs with the smallest battery capacity. This type of battery has very limited autonomy, and thus is of little use in such an infrastructure. Only when all the demand was met ( $N \geq 1536$ ) can approximately 25 UAVs replace fixed servers. The problem is the autonomy of the batteries, which prevents the replacement of a single underloaded server with a UAV. To further increase the number of UAVs, the battery life would have to be sufficient to maintain the UAVs being turned on for several hours, which is not a realistic assumption for battery-constrained

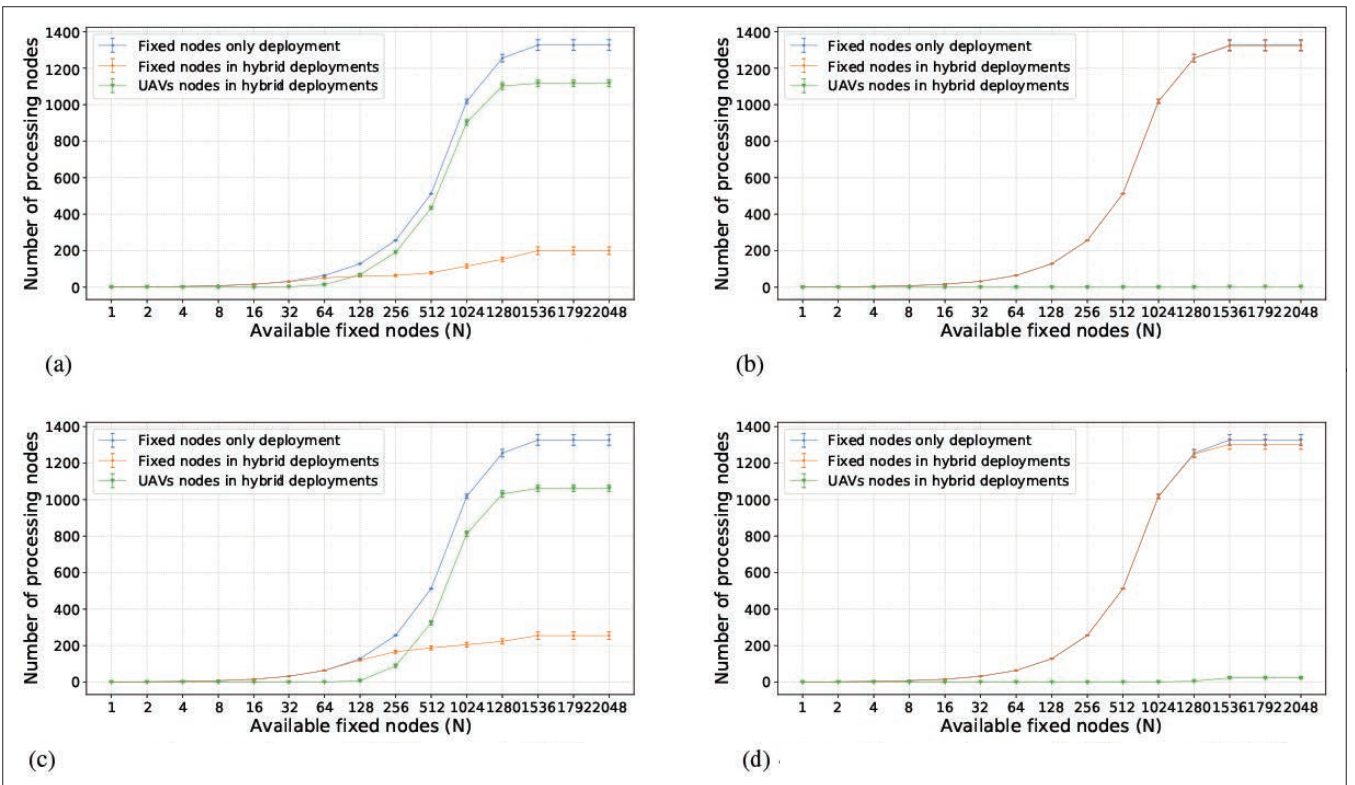


FIGURE 3. Number of servers and UAVs used for various scenarios: a) 34,200 mAh/22.8 V battery,  $UAV^C = 1$  and  $UAV^P = 1$ ; b) 34,200 mAh/22.8 V battery,  $UAV^C = 4$  and  $UAV^P = 1$ ; c) 34,200 mAh/22.8 V battery,  $UAV^C = 1$  and  $UAV^P = 0.50$ ; d) 4500 mAh/14.8 V battery,  $UAV^C = 1$  and  $UAV^P = 1$ .

UAVs. Technologies for charging batteries without interrupting the operation can help to extend UAV operation in fog infrastructures.

A comparison between the UFL algorithm and the dispatching scheme in [7] was carried out. Figure 4 depicts the mean percentage difference in the number of UAVs demanded by the two schemes; see the equation above. The scheme in [7] differs from UFL in three ways: first, it assumes unlimited energy; second, it can dispatch UAVs to process the workload at every time interval without evaluating future demands; and third, UAVs do not fly between different locations. We imposed battery limitation in the scheme in [7] for the sake of fair comparison. We denoted the original solution with battery limitation “single location.” We also implemented a version that allows a UAV to serve multiple locations, denoted “multiple locations.” UAVs are used in multiple locations if the battery can support the flight and the processing of the workload at the destination.

The results indicate that most fixed servers process heavy loads in small infrastructures ( $N \leq 32$ ), which does not create opportunities for replacing fixed nodes with UAVs. For large infrastructures ( $N \geq 1536$ ), the scheme in [7] requires a greater number of UAVs when compared to UFL. When UAVs have batteries with large capacity, the dispatching scheme employs up to 2 percent more UAVs to process the same workload. Moreover, for batteries with small capacity, the scheme in [7] requires, on average, 7 percent more UAVs when they can serve only a single location and almost 11 percent when they can serve multiple locations. The battery capacity has an impact on the number of required UAVs, since flights consume large amounts of

energy. Such consumption reduces the UAV operational time, especially with small battery capacity, preventing the processing of future workload, and consequently calls for more UAVs. The UFL algorithm produces more efficient deployments due to its consideration of energy consumption and planning of UAVs’ trajectories.

The present findings have revealed advantages and disadvantages in relation to the adoption of such hybrid infrastructures with both fixed and UAV nodes. One advantage is that hybrid infrastructures can simplify the deployment of fixed nodes where processing demands are low, thus reducing costs for deploying and maintaining nodes with underused servers continuously turned on. However, as long as UAV prices are higher than those of traditional servers, their use will remain limited. With price reduction, UAVs in fog infrastructures may become much more widespread.

## CONCLUSIONS

This article has investigated the employment of unmanned aerial vehicles as fog nodes by solving a fog node location problem. By considering UAVs in this early stage, it is possible to plan the best deployment and avoid placing fixed servers in locations with low demand. This article has described the UFL algorithm, which first solves the problem optimally by considering only fixed servers, and then tries to replace underutilized servers with UAVs, which can potentially serve more than one location. Results were obtained by varying different UAVs types and using a publicly available dataset. The UFL algorithm can be used for long-term planning of large fog infrastructures. Results show that a significant portion of the infrastructure could be replaced by UAVs

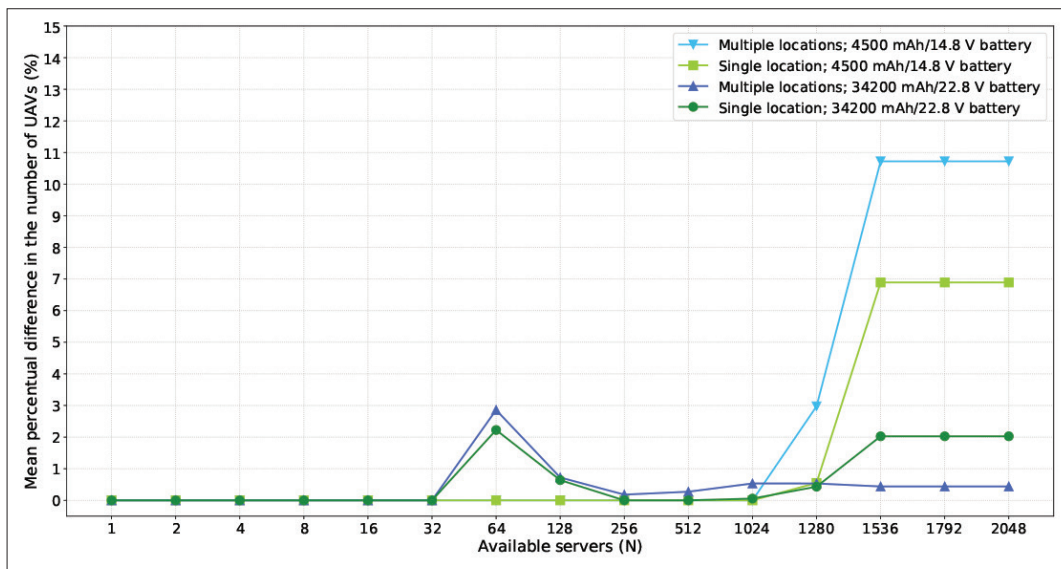


FIGURE 4. Difference in the number of UAVs required by the dispatching scheme in [7]. Results obtained for  $UAV^C = 1$  and  $UAV^P = 1$ .

depending on their price evolution. An additional benefit of using UAVs is the energy saved compared to an infrastructure with only fixed servers constantly powered all the time. Our investigation has revealed that such a deployment depends on the prices of UAVs being close to that of traditional servers. Currently, UAVs cost three to four times more than traditional servers, but prices are expected to decrease in the future as a function of mass production and wide use of unmanned aircraft.

The findings in this article suggest opportunities for future investigation. Similar solutions can be evaluated for different scales of infrastructure, from small neighborhoods to wider areas. By using fixed wing drones, for example, long hours of flight can be achieved, but this would require logistics to allow landings and takeoffs. Second, the solution presented in this article could be adapted to a scenario where the infrastructure already exists, and the UAVs are simply added to serve unforeseen and timely processing demands.

## ACKNOWLEDGMENTS

This research was funded by Google Latin American Research Awards 2019, the Government of Canada through the Emerging Leaders in the Americas Program (ELAP), National Council for Scientific and Technological Development (CNPq) grant number 140464/2018-2, and São Paulo Research Foundation (FAPESP) grant 15/24494-8.

## REFERENCES

- [1] N. L. S. da Fonseca and R. Boutaba, *Cloud Services, Networking, and Management*, Wiley, 2015.
- [2] "OpenFog Reference Architecture," 2017; <https://www.openfogconsortium.org/ra/>, accessed 24 May, 2017.
- [3] X. Masip-Bruin et al., "Foggy Clouds and Cloudy Fogs: A Real Need for Coordinated Management of Fog-to-Cloud Computing Systems," *IEEE Trans. Wireless Commun.*, vol. 23, no. 5, 2016, pp. 120–28.
- [4] M. Moradi et al., "Skycore: Moving Core to the Edge for Untethered and Reliable UAV-Based LTE Networks," *Proc. ACM MobiCom 2018*, 2018, pp. 35–49.
- [5] R. Bassoli et al., "Cubesat-Based 5G Cloud Radio Access Networks: A Novel Paradigm for On-Demand Anytime/Anywhere Connectivity," *IEEE Vehic. Tech. Mag.*, vol. 15, no. 2, 2020, pp. 39–47.
- [6] N. Mohamed et al., "Uavfog: A UAV-Based Fog Computing for Internet of Things," *Proc. IEEE SmartWorld/SCALCOM/UIC/ATC/CBDCOM/IOP/SCI 2017*, 2017, pp. 1–8.

- [7] J. Wang, K. Liu, and J. Pan, "Online UAV-Mounted Edge Server Dispatching for Mobile-to-Mobile Edge Computing," *IEEE Internet of Things J.*, vol. 7, no. 2, Feb 2020, pp. 1375–86.
- [8] R. A. C. da Silva and N. L. S. da Fonseca, "On the Location of Fog Nodes in Fog-Cloud Infrastructures," *Sensors*, vol. 19, no. 11, 2019.
- [9] R. A. C. da Silva and N. L. S. da Fonseca, "Location of Fog Nodes for Reduction of Energy Consumption of End-User Devices," *IEEE Trans. Green Commun. Net.*, vol. 4, no. 2, 2020, pp. 593–605.
- [10] H. V. Abeywickrama et al., "Empirical Power Consumption Model for UAVs," *IEEE VTC-Fall 2018*, 2018, pp. 1–5.
- [11] Q. Fan and N. Ansari, "Cost Aware Cloudlet Placement for Big Data Processing at the Edge," *IEEE ICC 2017*, May 2017, pp. 1–6.
- [12] X. Cao et al., "Mobile Edge Computing for Cellular-Connected UAV: Computation Offloading and Trajectory Optimization," *IEEE SPAWC 2018*, 2018, pp. 1–5.
- [13] G. Barlacchi et al., "A Multi-Source Dataset of Urban Life in the City of Milan and the Province of Trentino," *Scientific Data*, vol. 2, no. 1, Oct. 2015, p. 150055.
- [14] Opencellid; <http://www.opencellid.org>, accessed 1 Dec., 2021.
- [15] L. Chen et al., "Complementary Base Station Clustering for Cost-Effective and Energy-Efficient Cloud-Ran," *Proc. IEEE Smart-World/SCALCOM/UIC/ATC/CBDCOM/IOP/SCI 2017*, Aug. 2017, pp. 1–7.

## BIOGRAPHIES

RODRIGO A. C. DA SILVA is currently a Ph.D. candidate in computer science with the Institute of Computing, State University of Campinas (Unicamp), Brazil. He received his M.Sc. and B.Sc. degrees in computer science from Unicamp in 2015 and 2012, respectively. His current research interests include UAV communications, energy efficiency, fog computing, and cloud computing.

NELSON L. S. DA FONSECA received his Ph.D. degree in computer engineering from the University of Southern California in 1994. He is a full professor with the Institute of Computing at Unicamp. He served as an IEEE Communications Society Distinguished Lecturer. He served as the IEEE ComSoc Vice President Technical and Education Activities, Vice President Publications, and Vice President for Member Relations. He is past Editor-in-Chief for *IEEE Communications Surveys & Tutorials* (2008–2012).

RAOUF BOUTABA [F] received his M.Sc. and Ph.D. degrees in computer science from Sorbonne University in 1990 and 1994, respectively. He is currently a University Chair Professor and the director of the Cheriton School of Computer Science at the University of Waterloo, Canada. He also holds an INRIA International Chair in France. He is the founding Editor-in-Chief of *IEEE Transactions on Network and Service Management* (2007–2010) and the current Editor-in-Chief of the *IEEE Journal of Selected Areas in Communications*. He is a Fellow of the EIC, the Canadian Academy of Engineering, and the Royal Society of Canada.

One advantage is that hybrid infrastructures can simplify the deployment of fixed nodes where processing demands are low, thus reducing costs for deploying and maintaining nodes with underused servers continuously turned on. However, as long as UAV prices are higher than those of traditional servers, their use will remain limited. In case of price reduction, UAVs in fog infrastructures may become much more wide-spread.