

A Fast Gateway Placement Algorithm for Flying Networks

Gonçalo Santos, João Martins, André Coelho, Helder Fontes, Manuel Ricardo, Rui Campos
INESC TEC and Faculdade de Engenharia, Universidade do Porto, Portugal
goncalo.r.santos@fe.up.pt, joao.miguel.martins@fe.up.pt, andre.f.coelho@inesctec.pt,
helder.m.fontes@inesctec.pt, manuel.ricardo@inesctec.pt, rui.l.campos@inesctec.pt

Abstract—The ability to operate anywhere, anytime, as well as their capability to hover and carry cargo on board make Unmanned Aerial Vehicles (UAVs) suitable platforms to act as Flying Gateways (FGWs) to the Internet. The problem is the optimal placement of the FGWs within the flying network, such that the Quality of Service (QoS) offered is maximized. The literature has been focused on optimizing the placement of the Flying Access Points (FAPs), which establish high-capacity small cells to serve the users on the ground, overlooking the backhaul network design, including the FGW placement. The FGW placement problem is exacerbated in highly dynamic flying networks, where the dynamic traffic demand and the movements of the users may induce frequent changes in the placement of the FAPs.

The main contribution of this paper is a fast gateway placement (F-GWP) algorithm for flying networks that determines the optimal position of a FGW. With F-GWP, backhaul communications paths with high enough capacity are established between the FAPs and the FGW, in order to accommodate the traffic demand of the users on the ground. Simulation and experimental results show F-GWP is two orders of magnitude faster than its state of the art counterpart, while ensuring the same flying network performance.

Index Terms—Unmanned Aerial Vehicles, Flying Networks, Aerial Networks, Gateway Placement, Fast Algorithm, Quality of Service.

I. INTRODUCTION

The ability to move in the three-dimensional space, hover above the ground, and carry cargo on board makes Unmanned Aerial Vehicles (UAVs) suitable platforms to act as communications nodes, such as Wi-Fi Access Points (APs) and cellular Base Stations (BSs) [1]. This is paving the way to the deployment of flying networks to provide temporary on-demand broadband wireless connectivity and to reinforce the capacity of existing networks in a myriad of scenarios, including disaster scenarios [2] and outdoor festivities [3], as depicted in Fig. 1.

A challenge inherent to flying networks is the optimal placement of the UAVs such that the traffic demand of the users on the ground and their Quality of Service (QoS) requirements are fulfilled. Several UAV placement solutions have been proposed in the literature [3]–[14], but they are focused on enhancing the radio coverage and improving the QoS offered by the radio access network, which is formed by Flying Access Points (FAPs). Although the users on the ground are directly affected by the access network, the QoS offered by the latter is influenced by the backhaul network. Thus, the

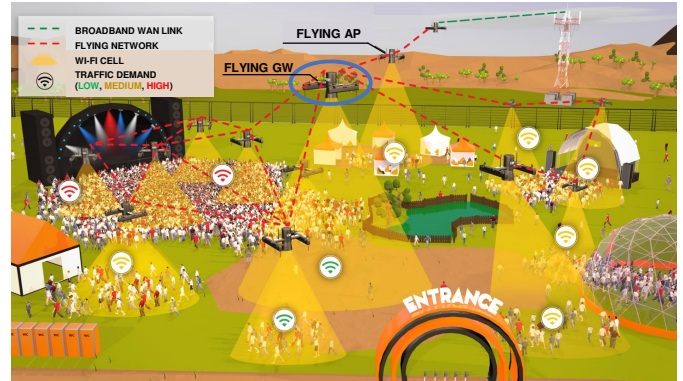


Fig. 1. Flying network providing Internet connectivity to the users in a music festival [4].

backhaul network needs to be carefully designed. A critical aspect in the backhaul network is the gateway placement.

In [15], a centralized traffic-aware gateway placement (GWP) algorithm for flying networks with controlled topology was proposed. GWP takes into account the knowledge about the offered traffic and the future positions of the FAPs to define the optimal position of a Flying Gateway (FGW), aiming at accommodating the traffic demand. However, the GWP algorithm follows an iterative approach, which may lead to long execution times to achieve the optimal solution. This may be a problem when using Edge Computing, especially for large and highly dynamic flying networks where the dynamic traffic demand and the movements of the users may induce frequent changes in the placement of the FAPs.

The main contribution of this paper is a fast gateway placement (F-GWP) algorithm for flying networks with controlled topology. Built upon the GWP algorithm, F-GWP is able to determine the optimal position of a FGW, in order to establish backhaul communications paths between the FAPs and the FGW able to accommodate the traffic demand of the users on the ground, but with execution times two orders of magnitude lower than GWP. This makes F-GWP especially suitable for highly dynamic flying networks using resource-constrained Edge Computing.

The rest of this paper is organized as follows. Section II presents the state of the art on UAV placement algorithms for flying networks. Section III defines the system model. Section IV formulates the problem addressed by this paper.

Section V presents the F-GWP algorithm. Section VI describes the evaluation of the F-GWP algorithm, including the networking scenarios employed, the performance metrics adopted, the performance results achieved, and their discussion. Finally, Section VII presents the main conclusions and directions for future work.

II. STATE OF THE ART

Several placement algorithms have been proposed in the literature to determine the optimal positions of UAVs forming flying networks. They are focused on addressing different challenges, including power consumption of the communications nodes and QoS guarantees [16]. Typically, the algorithms rely on deterministic approaches. A reference example is presented in [9], where the authors propose an algorithm that initially places the UAVs in the boundaries of a specific area composed of a set of users and afterwards follows an inward spiral until all the users are covered. Similar algorithms have been proposed in [5] and [8], where the objective is to maximize the radio coverage and minimize the required transmission power. These algorithms are computationally efficient; however, they typically converge to solutions that are not optimal from the network performance point of view, since the traffic demand of the users is not considered. The location of the users to be covered is also commonly considered to place the UAVs. In [11], an algorithm that places the UAVs vertically projected to the centroids of clusters of users is proposed, aiming at minimizing the required transmission power. An algorithm for placing UAVs that use directional antennas, taking into account a desired coverage area, is presented in [12]. It results in non-overlapping coverage areas formed by UAVs employing the minimum transmission power. Nevertheless, these algorithms are not traffic-aware, thus not providing QoS guarantees.

Stochastic UAV placement approaches have also been proposed. A fast algorithm for placing a UAV providing connectivity to users with different QoS requirements is proposed in [13]. The authors formulated the problem as a Mixed Integer Second Order Cone Problem and studied the relationship between the coverage radius and the height of the UAV. Moreover, they employed the Standard Genetic Algorithm and Multi-Population Genetic Algorithm, in order to maximize the number of covered users, by employing random UAV positions that follow natural selection processes, including selection, crossover, and mutation. In [14], a UAV placement algorithm based on Particle Swarm Optimization (PSO) is proposed. It aims at determining the positions of the UAVs over time such that the connectivity between them is ensured and the aggregate sensor data information in a wireless sensor network is maximized. For that purpose, the information about the positions of the communications nodes is considered to be known in advance. However, these algorithms rely on modeling each UAV as an individual particle and adjusting its movement according to a specific utility function, which may result in long execution times to achieve the optimal solution.

Overall, the state of the art UAV placement algorithms do not ensure QoS guarantees and low execution times simultaneously.

III. SYSTEM MODEL

We assume a flying network organized into a two-tier architecture and composed of two types of UAVs, as depicted in Fig. 1: 1) the FAPs, which are in charge of providing connectivity to the users on the ground, by enabling high-capacity small cells; and 2) the FGW, which forwards the traffic to/from the Internet. The FAPs are dynamically placed according to the traffic demand of the users on the ground by means of a state of the art FAP placement algorithm, such as the one proposed in [3]. For that purpose, the FAPs periodically take a snapshot of the network, including the positions of the users and the amount of traffic they are offering, and send this information to an Edge node, where the FAP placement algorithm is running; the Edge node may be one of the UAVs that compose the flying network or some node on the ground connected to the flying network. In turn, the optimal FGW placement is determined taking into account the future positions of the FAPs, which are defined by the FAP placement algorithm, as well as their traffic demand. The traffic demand of each FAP corresponds to the aggregate traffic exchanged with the users connected to it.

The interaction between the FAP and FGW placement algorithms takes advantage of the holistic and centralized view provided by the Edge node, where both algorithms run in parallel. This allows the use of the information provided by the FAP placement algorithm to calculate in advance the optimal FGW placement. Finally, the Edge node sends the updated positions to the FAPs and the FGW, which position themselves accordingly, enabling a seamless topology reconfiguration and maximizing the QoS offered by the flying network.

IV. PROBLEM FORMULATION

In order to model the wireless links, the Free-space Path Loss model expressed in Eq. (1) in logarithmic scale is employed, since the wireless links between UAVs flying dozen of meters above the ground are characterized by a strong Line of Sight (LoS) component. In Eq. (1), P_R represents the received power at the FGW, P_T is the transmission power, d represents the Euclidean distance between the FAPs and the FGW, f stands for the carrier frequency of the wireless links established between the FAPs and the FGW, and c is the speed of light in vacuum.

$$P_R = P_T - 20 \times \log_{10}(d) - 20 \times \log_{10}(f) - 20 \times \log_{10}\left(\frac{4 \times \pi}{c}\right) \quad (1)$$

We assume that the maximum capacity of the wireless links results from the data rate associated to the Modulation and Coding Scheme (MCS) index selected by the communications nodes. The selection of a given MCS index requires a minimum Signal to Noise Ratio (SNR), $SNR = P_R - P_{Noise}$, considering a constant noise power P_{Noise} , according to the rationale proposed in [15]. The wireless medium is shared

and we assume that all UAVs composing the flying network can listen to any other UAV. In order to avoid collisions of network packets, the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) mechanism is assumed to be employed, which enables transmissions only when the wireless channel is sensed to be idle.

The flying network is represented as a directed graph $G = (U, L)$, in which $U = \{UAV_1, \dots, UAV_N\}$ is the set of UAVs i placed at $P_i = (x_i, y_i, z_i)$, in a venue with dimensions $X \times Y \times Z$, and $L \subseteq U \times U$ represents the set of wireless links between UAV_i and UAV_j , where $i, j \in U$. The flying network active topology is composed of single-hop paths; it is represented by a tree $T(U, L_T)$, rooted at UAV_N , which constitutes a subgraph of G , where $L_T \subseteq L$ is the set of wireless links established between UAV_i and UAV_N , performing the role of FGW.

Let us assume that $UAV_i, i \in \{1, \dots, N-1\}$, acts as FAP and transmits a traffic flow $F_{i,N}$ with bitrate T_i bit/s towards UAV_N . The maximum channel capacity is equal to C^{MAX} bit/s, which is assumed to be equal to the data rate of the maximum MCS index of the wireless technology used. The maximum transmission power allowed for the wireless technology used is defined as P^{MAX} . The capacity of the bidirectional wireless link established between UAV_i and UAV_N at time t_k , where $t_k = k \cdot \Delta t, k \in \mathbb{N}_0, \Delta t \in \mathbb{R}$ is equal to $C_{i,N}(t_k)$. Δt is the update period imposed by the FAP placement algorithm, which considers a trade-off between the stability of the flying network and the time it takes to react to changes in traffic demand. Considering $N-1$ UAVs transmitting a traffic flow $F_{i,N}$ with bitrate $T_i(t_k)$ bit/s towards UAV_N , the problem consists of determining the position of $UAV_N, P_N = (x_N, y_N, z_N)$, such that the transmission power P_T is minimized, while ensuring a minimum SNR_i that guarantees $C_{i,N}(t_k)$ is high enough to accommodate $T_i(t_k)$ bit/s. The objective function is presented in Eq. (2a).

$$\underset{(x_N, y_N, z_N)}{\text{minimize}} \quad P_T \quad (2a)$$

subject to:

$$(N, i), (i, N) \in L_T, i \in \{1, \dots, N-1\} \quad (2b)$$

$$0 \leq P_T \leq P^{MAX} \quad (2c)$$

$$-\frac{X}{2} \leq x_i \leq \frac{X}{2}, i \in \{1, \dots, N\} \quad (2d)$$

$$-\frac{Y}{2} \leq y_i \leq \frac{Y}{2}, i \in \{1, \dots, N\} \quad (2e)$$

$$-\frac{Z}{2} \leq z_i \leq \frac{Z}{2}, i \in \{1, \dots, N\} \quad (2f)$$

$$(x_N, y_N, z_N) \neq (x_i, y_i, z_i), i \in \{1, \dots, N-1\} \quad (2g)$$

$$\begin{aligned} & (x_N - x_i)^2 + (y_N - y_i)^2 + (z_N - z_i)^2 \\ & \leq \left(10^{\frac{K+P_T-SNR_i}{20}}\right)^2, i \in \{1, \dots, N-1\} \end{aligned} \quad (2h)$$

$$K = -20 \log_{10}(f) - 20 \log_{10} \left(\frac{4\pi}{c} \right) - P_{Noise} \quad (2i)$$

$$\sum_{i=1}^{N-1} C_{i,N}(t_k) \leq C^{MAX}, i \in \{1, \dots, N-1\} \quad (2j)$$

$$0 < T_i(t_k) \leq C_{i,N}(t_k), i \in \{1, \dots, N-1\} \quad (2k)$$

The factors that influence the calculation of $P_N = (x_N, y_N, z_N)$ and P_T include a) the capacity of the wireless links between UAV_i and UAV_N at each instant t_k , which is affected by the number of communications nodes that use the same wireless channel and the behavior of the medium access protocol, and b) the interference between the communications nodes. Since these factors are difficult to model in order to solve the problem analytically, in this paper we propose a heuristic algorithm to achieve a solution.

V. FAST GATEWAY PLACEMENT ALGORITHM

In this section, the proposed algorithm is presented, including its rationale and implementation.

A. Rationale

The fast gateway placement (F-GWP) algorithm for flying networks was built upon the GWP algorithm, originally proposed in [15]. Similarly to GWP, the F-GWP algorithm relies on the holistic and centralized view provided by the Edge node, namely the future positions of the FAPs, defined by a state of the art FAP placement algorithm (e.g., the one proposed in [3]), and the traffic demand of the FAPs, in order to determine the optimal position of the FGW. The F-GWP algorithm aims at computing the minimum transmission power for the FAPs and the FGW that allows establishing wireless links between UAV_i and UAV_N with high enough capacity to accommodate the traffic demand T_i . For that purpose, the F-GWP algorithm initially determines the minimum SNR_i that enables the selection of an MCS index, MCS_i , equivalent to a data rate capable of accommodating T_i bit/s [17]. Since the medium is assumed to be shared by multiple FAPs, the fair share of the wireless channel capacity is considered for defining the traffic demand T_i , which results from the data rate associated to MCS_i over the number of FAPs making part of the flying network, as explained in [15].

The minimum SNR_i that enables the selection of MCS_i imposes a minimum received power P_{R_i} . Taking into account the transmission power of the FAPs, P_T , then the maximum admissible distance d_i between UAV_i and UAV_N can be calculated by means of the Free-space Path Loss model – cf. Eq. (1). In the three-dimensional space, d_i represents the radius of the sphere centered at UAV_i , corresponding to its transmission range, within which UAV_N should be placed. Considering $N-1$ UAVs acting as FAPs, UAV_N should be placed in the volume defined by the intersection of the spheres centered at each UAV_i , as depicted in Fig. 2.

The F-GWP algorithm is in charge of determining the minimum transmission power P_T that enables the intersection of all the spheres. While the GWP algorithm determines the value of P_T following an iterative approach, successively increasing P_T by 1 dBm from 0 dBm onwards until the intersection between the spheres occurs, the F-GWP algorithm

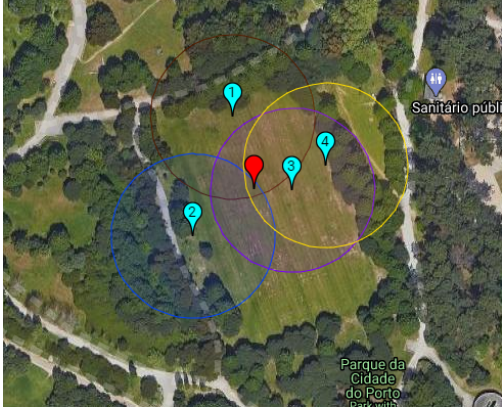


Fig. 2. Aerial view of the intersection of the FAPs' transmission ranges and F-GWP solution (unnumbered red marker) at Porto City Park [18].

solves an optimization problem that aims at minimizing P_T , assumed to be equal for all FAPs and the FGW. For this reason, while the GWP algorithm may lead to long execution times to achieve the optimal solution, the F-GWP algorithm is able to determine the optimal FGW placement and the minimum transmission power of the FAPs and the FGW in shorter time. This makes it especially suitable for highly dynamic flying networks using resource-constrained Edge Computing.

B. Algorithm Implementation

To implement the F-GWP algorithm, we took advantage of the package *SciPy* [19] for *Python* and its companion tool *optimize* [20]. This allows to solve the problem formulated in Section IV employing an iterative method. Using this framework, we first set the objective function and the problem constraints. Then, we defined the initial positions of the UAVs and set the initial transmission power to 1 dBm, which were considered by the solver to achieve the solution. Sequential Least Squares Programming [21] was used, which is an iterative method targeted for constrained non-linear optimization.

Regardless of the direction of the traffic – downlink or uplink – the output of the F-GWP algorithm is the same; all UAVs are configured with the same minimum transmission power and the wireless channel is assumed to be symmetric. In practice, the wireless channel is asymmetric. Yet, the algorithm will still work, if in the computation of the maximum distance d_i the loss for the worst direction (downlink or uplink) is considered; this basically implies the sum of the additional loss to the value obtained using the Free-space Path Loss model (cf. Section V-A). In this way, it is possible to use the F-GWP algorithm in emerging networking scenarios where applications generating symmetric traffic predominate [22], including social networks, video streaming, and online gaming.

VI. PERFORMANCE EVALUATION

The performance evaluation of the F-GWP algorithm is presented in this section, including the networking scenarios

employed, the performance metrics adopted, and the results obtained and their discussion.

A. Networking Scenarios

In order to evaluate the performance of the F-GWP algorithm, different networking scenarios were generated using the IEEE 802.11ac standard [23], taking into account an area 15 m long and wide, and 20 m high. Flying networks with 2, 4, 6, 8, 10, 15, and 20 FAPs were considered. For each number of FAPs, 10 random networks were generated. For each FAP, the minimum SNR_i was defined between 11 dB and 40 dB, which enables data rates from 58.5 Mbit/s to 780 Mbit/s, respectively, the channel bandwidth was set to 160 MHz, and the Guard Interval (GI) was set to 800 ns. The minimum SNR_i was defined taking into account the relation between the SNR and the data rates for the IEEE 802.11ac MCS indexes [17], according to the rationale proposed in [15].

B. Evaluation Metrics

The metrics used to compare the computational performance of the GWP and F-GWP algorithms were 1) the **execution time** (in seconds) required to obtain the solution – i.e., the FGW position and the transmission power P_T – and 2) the **memory used** by each algorithm (in megabytes, MB).

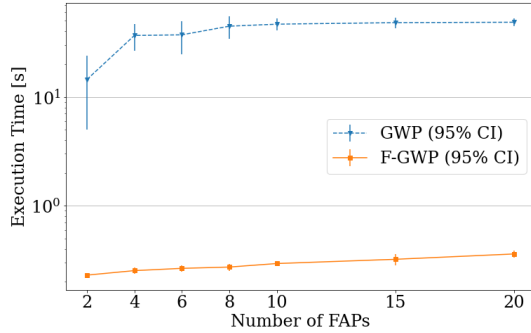
In order to evaluate the network performance achieved with the F-GWP algorithm in comparison with the GWP algorithm, the following networking metrics were considered:

- **Aggregate throughput** – the mean number of bits received per second by UAV_N (FGW);
- **Delay** – the mean time taken by the packets to reach the sink application running at UAV_N , since the instant of time they were generated by the source application running at each FAP.

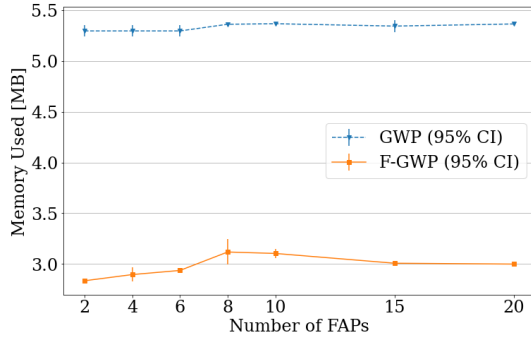
C. Computational Performance Results

The computational performance evaluation was carried out on a *Google Colaboratory* [24] Jupyter Notebook, in a hosted runtime with an Intel® Xeon® processor running at 2.20 GHz (2 threads) and 13 GB of RAM. As we can observe in the plot of Fig. 3a, the F-GWP algorithm is around 100 times faster than the GWP algorithm. For both small and large networks, the execution time is in the order of tens of milliseconds. The increase is linear with the number of FAPs making part of the flying network. The F-GWP algorithm, besides being faster than the GWP algorithm, proves to be much more stable as it has very low variability in almost all scenarios, as shown by the very short 95% confidence intervals.

Regarding memory usage, which is often neglected, we observe in Fig. 3b that F-GWP outperforms GWP using about 3 MB in the tested scenarios, when compared to the 5+ MB used by GWP. Both algorithms are stable, with low variability and slow increase as the network size increases. Memory usage was determined by measuring how much memory the *Python* interpreter was using immediately before and after running the GWP and F-GWP algorithms. The *Python*'s garbage collector, responsible for memory management, was



(a) Execution time for different number of FAPs.



(b) Memory usage for different number of FAPs.

Fig. 3. Computational performance results, in terms of execution time and memory usage, considering different number of FAPs. The F-GWP algorithm (square markers) outperforms the GWP algorithm (round markers) in both metrics.

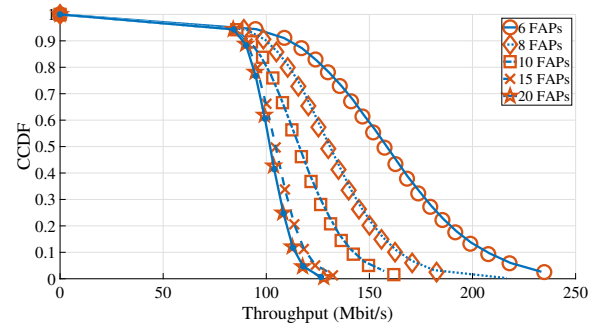
used in automatic mode, which may justify the peak around 8 FAPs that is observable in Fig. 3b.

D. Network Simulation Setup

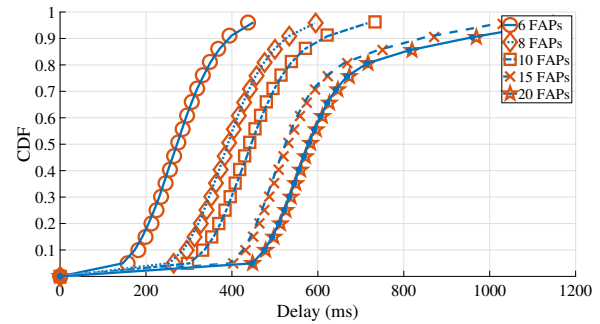
In order to evaluate the network performance when the F-GWP algorithm is employed and compare it against the GWP algorithm, the ns-3 simulator [25] was used. The FAPs and the FGW were carrying a Network Interface Card (NIC) in Ad Hoc mode, which was set to use the IEEE 802.11ac technology in channel 50, 160 MHz channel bandwidth, and 800 ns Guard Interval. The wireless link established between each FAP and the FGW was using a single spatial stream. The FAPs were generating UDP Poisson traffic towards the FGW. The data rate was defined by the *IdealWifiManager* mechanism.

E. Network Performance Results

The network performance results were obtained by means of ns-3 simulations, which consisted of 20 runs under the same exact networking scenarios employed in the computational performance evaluation presented in Section VI-C, considering $RngSeed = 20$ and $RngRun = \{1, \dots, 20\}$. The results are expressed as mean values and they are presented by using 1) the Cumulative Distribution Function (CDF) for the delay that represents the percentage of samples for which the mean packet delay was lower than or equal to x , and 2) the complementary CDF (CCDF) for the aggregate



(a) Throughput CCDF.



(b) Delay CDF.

Fig. 4. Network performance results, in terms of aggregate throughput and mean delay, considering different number of FAPs. The F-GWP algorithm (markers) enables the same network performance as the GWP algorithm (lines).

throughput that represents the percentage of samples for which the aggregate throughput was higher than x .

The obtained results, measured in the FGW, are presented in Fig. 4. They represent the network performance when the FGW is placed in the positions defined by 1) the F-GWP algorithm (markers), and 2) the GWP algorithm (lines), considering the transmission power P_T calculated by each algorithm. The plots clearly show the F-GWP algorithm achieves the same network performance as the GWP algorithm, for which the network performance evaluation presented in [15] allows to conclude significant gains regarding aggregate throughput and delay, while enabling decisions up to 100 times faster (cf. Section VI-C).

F. Discussion

There is a significant difference between the execution times of both algorithms for the same network performance, namely when the number of FAPs in the flying network increases. This happens because, in the GWP algorithm, for each additional FAP an additional equation has to be solved as part of the system of equations used to determine the position of the FGW [15]. While the execution times are higher even for a low number of FAPs, adding more equations as the number of FAPs increases does not have a major impact on the computational performance, as it can be observed in Fig. 3.

On the other hand, the F-GWP algorithm uses the equations as constraints of the optimization problem, which leads to

lower execution times in general. Yet, a higher number of FAPs in the flying network has an increased impact, since it implies more constraints to be taken into consideration, thus increasing the execution time. Since the size of the flying networks that are expected to be used in practice is limited, we do not expect to reach a point where the F-GWP algorithm does not prevail over the GWP algorithm.

The same behavior is observed when comparing the memory usage of both algorithms, for which the difference between them is another aspect in favor of the F-GWP algorithm. This improvement is relevant if we consider the F-GWP algorithm will run in an Edge node, which is resource-constrained.

The fact that we used *Python* and third-party packages (cf. Section V-B) to implement both algorithms leaves room for improvement. Some other low-level languages, such as *C* and *Java*, which can lead to better computational performance for both algorithms, are worthy to be explored.

VII. CONCLUSIONS

This paper proposed a fast gateway placement algorithm for flying networks, named F-GWP, which is able to determine the optimal position of a FGW in shorter time. The experimental results show the F-GWP algorithm has significant gains from the execution time point of view, enabling decisions up to 100 times faster than the state of the art GWP algorithm, while ensuring backhaul communications paths with high enough capacity to accommodate the traffic demand of the users. This makes F-GWP especially suitable for highly dynamic flying networks using resource-constrained Edge Computing. As future work, we aim at exploring the F-GWP algorithm in multi-FGW and multi-hop flying networks composed of UAV relays forwarding traffic between the FAPs and the FGWs.

ACKNOWLEDGMENTS

This work is co-financed by National Funds through the Portuguese funding agency, FCT – Fundação para a Ciência e a Tecnologia, within project UIDB/50014/2020 and by the European Regional Development Fund (FEDER), through the Regional Operational Programme of Lisbon (POR LISBOA 2020) and the Competitiveness and Internationalization Operational Programme (COMPETE 2020) of the Portugal 2020 framework [Project 5G with Nr. 024539 (POCI-01-0247-FEDER-024539)]. The third author also thanks the funding from FCT under the PhD grant SFRH/BD/137255/2018.

REFERENCES

- [1] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: opportunities and challenges," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 36–42, 05 2016.
- [2] N. Zhao, W. Lu, M. Sheng, Y. Chen, J. Tang, F. R. Yu, and K.-K. Wong, "Uav-assisted emergency networks in disasters," *IEEE Wireless Communications*, vol. 26, no. 1, pp. 45–51, 2019.
- [3] E. N. Almeida *et al.*, "Traffic-aware multi-tier flying network: Network planning for throughput improvement," in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, 04 2018, pp. 1–6.
- [4] "Wise," <http://wise.inesctec.pt>, (Accessed on 12/09/2020).
- [5] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Drone small cells in the clouds: Design, deployment and performance analysis," in *2015 IEEE global communications conference (GLOBECOM)*. IEEE, 2015, pp. 1–6.
- [6] E. Kalantari, M. Z. Shakir, H. Yanikomeroglu, and A. Yongacoglu, "Backhaul-aware robust 3d drone placement in 5g+ wireless networks," in *2017 IEEE international conference on communications workshops (ICC workshops)*. IEEE, 2017, pp. 109–114.
- [7] Y. Zeng, R. Zhang, and T. J. Lim, "Throughput maximization for uav-enabled mobile relaying systems," *IEEE Transactions on Communications*, vol. 64, no. 12, pp. 4983–4996, 2016.
- [8] M. Alzenad, A. El-Keyi, F. Lagum, and H. Yanikomeroglu, "3-d placement of an unmanned aerial vehicle base station (uav-bs) for energy-efficient maximal coverage," *IEEE Wireless Communications Letters*, vol. 6, no. 4, pp. 434–437, 2017.
- [9] J. Lyu, Y. Zeng, R. Zhang, and T. J. Lim, "Placement optimization of uav-mounted mobile base stations," *IEEE Communications Letters*, vol. 21, no. 3, pp. 604–607, 2016.
- [10] M. Alzenad, A. El-Keyi, and H. Yanikomeroglu, "3-d placement of an unmanned aerial vehicle base station for maximum coverage of users with different qos requirements," *IEEE Wireless Communications Letters*, vol. 7, no. 1, pp. 38–41, 2017.
- [11] J. Yu, R. Zhang, Y. Gao, and L.-L. Yang, "Modularity-based dynamic clustering for energy efficient uavs-aided communications," *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 728–731, 2018.
- [12] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage," *IEEE Communications Letters*, vol. 20, no. 8, pp. 1647–1650, 2016.
- [13] Y. Chen, N. Li, C. Wang, W. Xie, and J. Xv, "A 3d placement of unmanned aerial vehicle base station based on multi-population genetic algorithm for maximizing users with different qos requirements," in *2018 IEEE 18th International Conference on Communication Technology (ICCT)*. IEEE, 2018, pp. 967–972.
- [14] H. J. Na and S.-J. Yoo, "Pso-based dynamic uav positioning algorithm for sensing information acquisition in wireless sensor networks," *IEEE Access*, vol. 7, pp. 77 499–77 513, 2019.
- [15] A. Coelho *et al.*, "Traffic-aware gateway placement for high-capacity flying networks," *arXiv preprint arXiv:1912.09413*, 2019.
- [16] Z. Kang, C. You, and R. Zhang, "3d placement for multi-uav relaying: An iterative gibbs-sampling and block coordinate descent optimization approach," *arXiv preprint arXiv:2006.09658*, 2020.
- [17] "Mcs index chart - 802.11ac - vht," <https://www.wlanpros.com/mcs-index-charts/>, (Accessed on 08/09/2020).
- [18] "Porto city park," <https://www.cm-porto.pt/jardins-e-parques/parque-da-cidade>, (Accessed on 09/09/2020).
- [19] "Scipy," <https://www.scipy.org/>, (Accessed on 03/09/2020).
- [20] "Scipy's optimize tool," <https://docs.scipy.org/doc/scipy/reference/optimize.html>, (Accessed on 03/09/2020).
- [21] A. Iserles, *Acta Numerica 1995: Volume 4*, ser. Acta Numerica. Cambridge University Press, 1995. [Online]. Available: <https://books.google.pt/books?id=Hs96x3Nah2QC>
- [22] H. Elshaer, F. Boccardi, M. Dohler, and R. Irmer, "Downlink and uplink decoupling: A disruptive architectural design for 5g networks," in *2014 IEEE Global Communications Conference*, 12 2014, pp. 1798–1803.
- [23] "Ieee standard for information technology–telecommunications and information exchange between systems—local and metropolitan area networks—specific requirements—part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications—amendment 4: Enhancements for very high throughput for operation in bands below 6 ghz." *IEEE Std 802.11ac(TM)-2013 (Amendment to IEEE Std 802.11-2012, as amended by IEEE Std 802.11ae-2012, IEEE Std 802.11ad-2012, and IEEE Std 802.11ad-2012)*, pp. 1–425, 2013.
- [24] "Google colabouratory," <https://colab.research.google.com/>, (Accessed on 03/09/2020).
- [25] NS-3, "Network Simulator." [Online]. Available: <https://www.nsnam.org/>