

RESEARCH ARTICLE

NSGA-II Based Joint Topology and Routing Optimization of Flying Backhaul Networks

SÉRGIO E. SABINO¹ AND ANTÓNIO M. GRILO^{2,1}, (Senior Member, IEEE)

¹Instituto Superior Técnico—Universidade de Lisboa, 1049-001 Lisbon, Portugal

²INESC-ID, 1000-029 Lisbon, Portugal

Corresponding author: António M. Grilo (antonio.grilo@inesc-id.pt)

This work was supported in part by the Fundação Calouste Gulbenkian, and in part by the National Funds through the Fundação para a Ciência e a Tecnologia (FCT) under Project UIDB/50021/2020.

ABSTRACT This paper addresses the problem of optimizing the deployment of Flying Backhaul Networks (FBNs). The latter comprise Unmanned Aerial Vehicles (UAVs), which are used as access points to provide coverage to a set of ground nodes deployed in a target area. The optimization problem is addressed by means of a Multi-Objective Optimization Algorithm (MOEA), which calculates Pareto curves of UAV placement, providing different trade-offs between the considered objectives: (1) to minimize the number of UAVs, and (2) to maximize the Packet Delivery Ratio (PDR). The selected MOEA is NSGA-II. An embedded single objective Genetic Algorithm (inner-GA) is used to optimize routing, finding the paths that maximize the PDR. In order to obtain consistent solutions for the PDR taking into account MAC layer contention, the scheme makes use of an existing fixed-point algorithm (FPA). Simulation results were obtained for different scenarios combining average versus maximin PDR objective functions, two different routing optimization algorithms, as well as single sink versus multiple sink traffic patterns.


INDEX TERMS Flying backhaul networks, topology optimization, routing optimization, NSGA-II, unmanned aerial vehicles.

I. INTRODUCTION

Traditionally, Unmanned Aerial Vehicles (UAVs) have been used by the military for surveillance and reconnaissance operations. However, with the advent of robust wireless networking technologies, UAVs equipped with wireless transceivers can be enabled to communicate with ground nodes (GNs) as well as other UAVs [1]. As such, a swarm of UAVs can be configured to operate cooperatively as a Flying Backhaul Network (FBN). Among the possible applications of UAV-based networks, also known as Flying Ad Hoc Networks (FANETs), civilian application are envisioned to play an important role, e.g., UAVs can be deployed in critical scenarios, such as natural disaster (wildfire, floods, earthquake, etc.) to provide communication services to the GNs. In such scenarios, communication among different rescue teams is vital for mission coordination. In the absence of a functional conventional terrestrial communication network (e.g., cellu-

lar network), such coordination may fail. For example, due to the large areas to be covered or accessed, teams may not be able to report their findings or subsequent strategic plan in a timely manner, and satellite communication may not be an option due the high cost of operation and/or lack of terminal equipment. This is where FBNs may be decisive.

FBNs present the following advantages: firstly, UAV deployment is faster than recovering a crippled cellular network. Secondly, the acquisition and operational costs are lower. However, the design of such network is challenging and should be done carefully. Finding a suitable UAV topology that meets the overall network Quality of Service (QoS) requirements is a challenging task in an FBN comprising multiple UAVs. This issue is still attracting the academic community [2], [3], [4], [5], [6], [7]. Essentially, optimized UAV placement can be achieved either using a centralized or distributed scheme. In a typical centralized UAV scheme, there will be a single entity responsible for processing GN data and updating the UAVs with the new positions. This is different from a distributed placement scheme, where UAVs

The associate editor coordinating the review of this manuscript and approving it for publication was Xujie Li .

work cooperatively to adjust their positions based on local interactions [8].

This paper proposes a centralized algorithm architecture to optimize deployment and routing of an FBN whose mission is to cover GNs located in a target area. The goal is to maximize the Packet Delivery Ratio (PDR), while committing the minimum number of UAVs. In order to achieve this multi-objective goal, the proposed scheme relies on a MOEA, more specifically the Non-dominated Sorting Genetic Algorithm II (NSGA-II) [9], together with a nested single objective Genetic Algorithm (GA).

The main contributions of this work are the following:

- (i) A comprehensive review and comparative analysis of the related work on FBN topology optimization;
- (ii) A novel multi-objective optimization scheme, which uses NSGA-II to calculate the Pareto curves of non-dominated FBN topologies, providing different trade-offs between number of UAVs (minimized) and PDR (maximized), considering MAC layer contention;
- (iii) An algorithm hierarchical structure, where the outer NSGA-II finds non-dominated topologies, while the inner single objective GA finds the best routes for each tested topology, and a fixed-point algorithm [10] is used as objective function to estimate end-to-end PDR;
- (iv) Network performance results and novel insights considering different combinations of average versus maximum PDR metrics, shortest path versus GA based routing, as well as single sink versus multiple sink traffic patterns, some of them under the perspective of MAC layer contention.

This paper is a development of previous work presented in [11], extending it with an updated and improved related work analysis, a deep explanation of the algorithms, and a more complete set of test scenarios and objective functions, where the advantages and limitations of the proposed approach become more evident.

The remainder of this paper is organized as follows. The related work is presented in Section II. Section III presents the proposed system model. Section IV presents the mathematical formulation of the objective functions and their constraints. In Section V, NSGA-II based UAV placement and routing optimization are presented. Section VI presents the simulation scenarios and results. Section VII presents the conclusions and future work.

II. RELATED WORK

This section presents the related work on UAV placement algorithms for FBNs. Generally, UAV placement schemes fall into two main categories: distributed and centralized [12].

In distributed schemes, UAVs exchange and use local information in order to optimize their positions or trajectories, aiming to satisfy the required level of coverage of user nodes. This approach typically makes the network more responsive and resilient in case of unexpected changes. Most of the existing distributed algorithms adapt concepts coming from physics or animal behavior. For instance, Basu *et al.* [13]

proposed an UAV placement scheme inspired on bird flocking. The aim is to maintain connectivity among UAVs, while adapting to the mobility of the nodes on the ground. In order to achieve this goal, each UAV should follow a set of rules/behavior, namely, “move to the point above ground Centroid”, “repel from UAV”, “attract toward UAV” and “random walk”. The rules are represented by a state digram, and the transition from one state to another depends on the distance between UAVs. The authors in [14] propose a distributed mobility algorithm based on a Virtual Spring Force (VSF) model, through which the UAVs self-organize into a mesh structure by guaranteeing QoS over the aerial link, and providing coverage to isolated GNs. This algorithm was further developed in [8], where Connection Recovery and Maintenance (CRM), as well as Mobility Prediction (MP) mechanisms were also integrated. In [4], the authors propose a Graph Convolutional Multi-Agent Reinforcement Learning (MARL) method to maximize coverage of ground nodes by UAVs. The convolutional input layers allow local collaboration among neighbour UAVs, which is translated into better coverage performance compared with Deep Q-Learning (DQL) running independently in each UAV. The number of UAVs is fixed, and the reward function only considers the coverage score and energy spent by the UAV, not taking into account MAC layer performance.

Centralized schemes rely on a single entity having full knowledge of node positions, and control over the UAVs. In this paper, we propose a centralized scheme to jointly optimize UAV placement and routing in FBNs. Therefore, the remainder of this section will focus approaches of this kind proposed so far. In the past few years, this topic was the target of a significant number of research works. The works on centralized schemes may be divided in two groups.

The first group does not consider inter-UAV connectivity (e.g., the UAV relay nodes are directly connected to terrestrial base stations), or simply abstracts inter-UAV communication, taking it for granted, so that the placement algorithm does not have to bother with it. Galkin *et al.* [15], proposed small cells mounted on UAVs to offload ground users from the macrocell infrastructure. The K-means clustering algorithm is used to optimally place the UAVs. In [16], Kalantari *et al.* proposed a 3D UAV placement scheme using the Particle Swarm Optimization (PSO) algorithm. Similarly to our work, UAVs are used as flying access points, and the main goal is to find the minimum number of UAVs and their 3D coordinates to service all users with some target QoS requirement. Mozaffari *et al.* [17], use circle packing theory to deploy multiple UAVs, in order to maximize the coverage area. The paper [12] was a previous work by our team. Here, an NSGA-II based scheme is used to optimize two different objectives: (1) to maximize the fulfillment of the data rates required by the GNs, and (2) to minimize the number of UAVs. A scheme to reduce algorithm search space based on the computation of the convex hull ([18]) formed by the GNs was proposed, which is also adopted in the present work. The link budget is calculated based on a log-distance

path loss model. IEEE 802.11g data rates are considered, and the data rate of a link is simply the highest among those whose receiver sensitivity is lower than the received power. In [2], the authors consider the 3D placement of drone relays in mobile cellular networks, namely 5G, where each serial base station relay must be within reach of a ground base station. It starts by defining the optimal coverage problem as a Mixed-Integer Linear Problem (MILP), which is NP-complete. The complexity is reduced by the OnDrone algorithm, which is based on an Extremal Optimization Algorithm (EOA). The paper also proposes a Hungarian method to solve the problem of minimizing the paths of the aerial base stations towards their optimal destinations, when the coverage optimization algorithm is recomputed. Then, it proposes to improve on-route coverage by using Bézier curves instead of straight lines in the routes towards the destination points, in order to bias the routes towards areas with higher density of GNs. Interference is taken into account in the objective function, and it also considers 3D beamforming to reduce interference. In [5], Network-based Heterogeneous PSO (NHPSO) is proposed. In this scheme, a heterogeneous scale-free network is employed as the topology structure. Besides, it introduces a heterogeneous strategy of particles along with the heterogeneous topology structure, where topological central particles (i.e., high-degree particles) are encouraged to utilize more information from neighbors for self-improving, while low-degree particles tend to learn among themselves to maintain the diversity. The proposed NHPSO is compared with other PSO and non-PSO algorithms in standard optimization problems, as well as in a cellular network coverage problem with a fixed number of UAVs. In [6], the authors use a GA in order to optimize the placement of patch UAV base stations, after a disaster that reduces the capacities of base stations. The GA uses a weighed objective function that takes into account the number of UAVs, energy expenditure, and a penalty related with QoS level. UAVs are considered fully functional base stations. In [7], the authors present two algorithms to deploy UAV base stations to cover GNs in a 4G network, so that the number of drone base stations is minimized and the sum data rate is maximized. The first algorithm, called Data-Driven 3D Placement (DDP), is based on balanced K-means clustering. The second algorithm, called Enhanced DDP (eDDP), is an enhancement that tries to minimize the overlapping between coverage areas of the drone and ground base stations. It partitions the area in different parts and then employs the first algorithm to minimize the number of drone base stations and to deploy them separately for each partition.

The second group of centralized schemes considers that the UAVs form a multihop mesh backhaul network in order to deliver the traffic to destination nodes or gateways to outside networks. This mesh backhaul must form a connected graph, guaranteeing the delivery of traffic from any point of the network to any other point of the network, and it must be considered in the optimization process. The work presented in this paper belongs to this group.

Reina *et al.* [19] proposed an optimized deployment scheme that uses Multi-Layout Multi-Population Genetic Algorithm (MLMPGA) as the optimization technique. The network has three main requirements that should be satisfied and balanced: it should provide coverage and redundancy, and it should be fault tolerant. In order to achieve this goal, the authors define a weighted multi-objective fitness function, allowing the use of the single-objective MLMPGA. In [20], the authors mathematically formulate the placement optimization of UAVs as a multi-objective problem and solve it as bi-objective linear optimization model. In paper [21], The authors propose a system named Traffic-Aware Multi-Tier Flying Network (TMFN). A TMFN consists of a mobile and physically reconfigurable network of Flying Mesh Access Points (FMAs) and Gateway UAVs organized in a two-tier architecture, which is able to quickly readjust its topology according to the traffic demands of the users. In order to control the TMFN's topology, the authors propose a traffic aware Network Planning (NetPlan) algorithm, based on the concept of Potential Fields (PFs). Although the paper assumes a multihop network architecture and tests the proposed algorithm in such an environment with ns-3, the NetPlan algorithm only takes into account the access links between GNs and FMAs. In [22], the authors present a scheme to explore a region of interest where a terrestrial network is deployed, detecting holes in the network topology, after which an algorithm optimizes the selection of spots for placement of patch UAVs that will increase the communication performance. In [23], the authors propose a topology construction and adjustment scheme, where the optimal topology is built using a PSO algorithm, while the adjustment is based on gradient descent using the same performance metric. The two algorithms are integrated so that PSO only runs when the edit distance between the current graph and the previous one calculated by PSO is high enough. Thus, there is a compromise between the optimality of PSO and the computational performance of gradient descent. There is no attempt to minimize the number of UAVs, which is fixed, so the PSO is single objective. Departing from the previous work, in [3], the same team proposes a joint mission assignment and topology management scheme. The scheme comprises three algorithms. The first one performs a global optimization of mission assignment (greedy), then PSO based router placement and routing. The second algorithm locally adjusts relay UAV positions before the difference (edit distance) relative to the initial network is too large. The third algorithm performs mission reassignment of UAVs. Both the second and third algorithms try to avoid running the first one (i.e., global optimization), in order to reduce computation complexity. In both works, the network performance metric is based on link distance and thus does not take into account MAC layer aspects. In [24], the authors focus on cellular networks, proposing an heuristic iterative algorithm to obtain a connected network comprising ground terminal nodes and a ground control station, which is achieved by means of a mesh network of UAV relay nodes. The worst case complexity of the proposed algorithm

is demonstrated to be $O(n^2)$. However, the algorithm is not proved to achieve the minimum number of UAV relay nodes or the respective optimal positions.

Table 1 lists the related works described above, highlighting their main characteristics for comparison purposes. The present paper focuses the global joint topology and routing optimization of the UAV backhaul network using a centralized algorithm, considering MAC layer effects on link performance. Although this is a development from our previous work in [12], the latter did not consider the multihop FBN among the UAVs in the optimization scheme.

There are several related works in Table 1 that consider multiple optimization objectives (e.g., [19]). These works use to reduce the multiple objectives to single objective optimization, by means of prioritization, weighted sum, product, etc. Such techniques are more prone to become trapped in local minima, and to miss Pareto optimal solutions in non-convex spaces. In contrast with these works, our scheme employs a true multi-objective optimization algorithm to calculate Pareto curves (number of UAV base stations versus PDR) instead of single solutions. Based on the latter, the decision-making entities can choose the one that best fits the application conditions and mission management strategy at hand.

The work [23] is the one that bears more resemblance to ours in terms of system model and performance objective, though it fixes the number of UAVs, solving a single objective problem with PSO. Unlike this work and [3] by the same team, we integrate MAC layer contention in the global optimization of the UAV positions. This is done by means of the fixed-point algorithm (FPA) presented in [10]. Although we do not include a low complexity adjustment mechanism to minimize the frequency with which the global optimization algorithm must be run to adapt to GN position changes, it could be easily integrated with a scheme similar to the one proposed in [23].

Meta-heuristic algorithms besides GA have already been used for routing optimization: Ant Colony Optimization (ACO), Artificial Bee Colony (ABC), PSO, Simulated annealing (SA), Firefly algorithm, Differential Evolution (DE), hybrid algorithms, etc. Similarly, multi-objective algorithms besides NSGA-II exist in the literature, such as Multi-Objective Differential Evolution (MODE), Multi-Objective PSO (MOPSO), Strengthen Pareto Evolutionary Algorithm (SPEA), hybrid algorithms, etc. Our choice of GA based meta-heuristics is backed by existing literature (e.g., [25], [26], and [27]). Detailed comparison between alternative algorithms deserves a dedicated study, being out-of-scope of this paper.

III. SYSTEM MODEL

We consider a connected wireless network comprising a set of UAVs (U) and a set of GNs (V) deployed in an area \mathcal{A} . UAVs are assumed to be deployed in the 3D plane with their coordinates represented as $q_{(x,y,h)}^u \in \mathcal{A}$, where h represents the flight altitude of a certain UAV u_i . On the other hand, GNs are assumed to be on the ground with coordinates $q_{(x,y,0)}^v \in \mathcal{A}$.

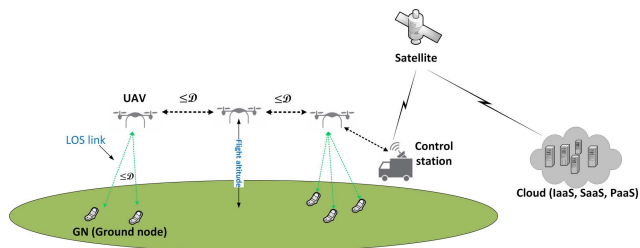


FIGURE 1. Mobile network supported by an FBN of UAV base stations.

\mathcal{A} , representing a 2D plane positioning in \mathcal{A} . Nodes have a maximum communication range which is denoted by \mathcal{D} , and communication among GNs is only realized through UAVs in a multihop fashion. Fig. 1 depicts the proposed communication system. The control station can be connected to external networks, such as cellular network, satellite or Cloud.

Definition 1: A wireless network is said to be connected when there is a path between every pair of nodes. Hence, in a connected network all nodes are reachable.

Definition 2: Two or more nodes are said to be neighbors when the Euclidean distance between each pair is shorter than or equal to \mathcal{D} .

IV. MULTI-OBJECTIVE OPTIMIZATION PROBLEM

The scheme proposed in this paper aims to find the best trade-offs between the FBN cost in terms of the number of UAVs, and the achieved PDR. In this section, mathematical formulation of each objective function is presented.

A. OBJECTIVE 1: MINIMIZING THE NUMBER OF UAVS

Similarly to [12], the present work assumes that there is a cost associated with each used UAV. Thus, minimizing the number of UAVs is desirable. This is done by restricting the UAV coverage to the sub-area $a' \subset \mathcal{A}$ that corresponds to the convex hull (convex envelope) [18] formed by the GNs in \mathcal{A} . This restriction of the deployment area also reduces the complexity of the algorithm. In order to further reduce that complexity, a' is discretized in a grid layout according to the following relation: $\Delta = \mu\mathcal{D}$; $\mu \in [0, 1]$, where Δ is the distance between two neighboring UAVs, which is adjusted by changing μ . Let $Q \subset a'$ be the discrete set of allowed UAV deployment points.

Let $q_j \in Q$ be the j^{th} potential UAV placement point. Let $\{\delta_{q_j}^u\}, \forall u \in U, \forall q_j \in Q$ be defined as a set of binary variables indicating which points are currently being used by an UAV, as follows:

$$\delta_{q_j}^u = \begin{cases} 1 & \text{if UAV } u \text{ is located at } q_j \\ 0 & \text{Otherwise.} \end{cases}$$

Let $\{\zeta_v^u\}, \forall u \in U, \forall v \in V$ be defined as a set of binary variables indicating which GNs are being serviced by each deployed UAV. It is assumed that a GN will be connected to

TABLE 1. Related work on FBN optimization.

Algorithm Architecture	Reference	UAV Altitude	Inter-UAV Multihop Routing	Optimization Objectives	Link Performance Model of the Algorithm	Types of Algorithms
Distributed	[13]	Fixed	No	GN coverage, inter-UAV distance	Fixed transmission range	Bird flocking heuristic
	[14]	Fixed	Yes	Link budget, GN coverage	Link budget	VSF heuristic
	[8]	Fixed	Yes	Link budget, GN coverage	Link budget	VSF heuristic, CRM, MP
	[4]	Fixed	Yes	max coverage, min energy consumption	Fixed transmission range	MARL (reward is product of objective functions)
Centralized	[15]	Fixed	No	min transmission distance	NA	K-means
	[16]	Variable	No	max GN coverage, min UAV cost, max data rate	Link budget, spectral efficiency, UAV capacity	PSO (prioritized objectives)
	[12]	Variable	No	min UAV cost, max data rate	Link budget, IEEE 802.11g data rates	NSGA-II
	[2]	Variable	No	max GN coverage, min reposition delay	Signal-to-interference-plus-noise-ratio, data rate	EOA, MILP, Bézier flight routes
	[5]	Variable	No	max total data rate	Shannon capacity (fixed background noise power)	NHPSO
	[6]	Fixed	No	min UAV cost, min throughput penalty	Shannon capacity (w/ interference)	GA (weighted sum of objectives)
	[7]	Variable	No	min UAV cost	Shannon capacity (w/ interference)	K-means, heuristics
	[19]	Fixed	Yes	max GN coverage, max fault tolerance, max redundancy	Fixed transmission range	MLMPGA (weighted sum of objectives)
	[20]	Variable	Yes	min UAV cost, min altitude	Fixed transmission range	Bi-objective linear model, ϵ -constraint method
	[21]	Fixed	Yes	max coverage	Link budget	PFs
	[22]	Fixed	Yes	min hole packet reception ratio	TOSSIM signal-to-interference-plus-noise-ratio and error models	Connectivity-based k-hop clustering, strongly connected components
	[23]	Fixed	Yes	min sum of path lengths	distance based link quality	PSO, gradient descent
	[3]	Fixed	Yes	max mission satisfaction, max link quality	distance based link quality	PSO, gradient ascent, heuristics
	[24]	Fixed	Yes	min UAV cost	Fixed transmission range	Heuristic
	Present work	Fixed	Yes	min UAV cost, max PDR	Fixed transmission range, CSMA/CA MAC	NSGA-II w/ nested inner-GA and FPA

the closest deployed UAV:

$$\zeta_v^u = \begin{cases} 1 & \text{if GN } v \text{ is connected to UAV } u \\ 0 & \text{Otherwise.} \end{cases}$$

The first objective is to assign values to δ_{qj}^u , so that valid solutions are found to the following problem:

$$\min \sum_{q_j \in Q} \sum_{u \in U} \delta_{qj}^u \tag{1}$$

s.t.:

$$\sum_{q_j \in Q} \delta_{qj}^u \leq 1, \forall u \in U, \tag{2a}$$

$$\sum_{u \in U} \delta_{qj}^u \leq 1, \forall q_j \in Q, \tag{2b}$$

$$\sum_{u \in U} \zeta_v^u \geq 1, \forall v \in V, \tag{2c}$$

Constraint (2a) indicates that each UAV u cannot be placed in more than one point at the same time. Constraint (2b)

indicates that each point q_j is occupied by a single UAV. Constraint (2c) ensures that a GN is within communication range of at least one UAV. The cardinality of the set Q defines the maximum number of UAVs that can be used to cover a given convex hull. Details on how to maintain the network fully connected can be found in [12].

B. OBJECTIVE 2: MAXIMIZING THE PACKET DELIVERY RATIO

The PDR model is based on the one proposed in [10] by Baras *et al.* The model provides quantitative statistic relationships between (1) the PDR loss parameters used to characterize multi-user interference and physical path conditions, and (2) the traffic rates between the origin-destination pairs. The model takes into account the effects of the hidden nodes, scheduling algorithms, IEEE 802.11 MAC and PHY layer transmission failures, finite packet retries at the MAC layer, etc., in arbitrary network topologies where multiple paths (i.e., traffic flows) share nodes.

In order to model the MAC layer, the following assumptions are made:

- (i) The network consists of $|U| + |V|$ nodes and a path set P that is used to forward traffic between the source-destination (S-D) pairs in the network;
- (ii) The 802.11 MAC layer with RTS/CTS mechanism;
- (iii) The unit of time is a time slot, which is equal to the back-off slot of the 802.11 protocol;
- (iv) The nodes access the channel with a fixed probability as proposed in [28];
- (v) The scheduler above the MAC layer keeps scheduling the same packet until it is successfully transmitted by the MAC layer, thus recovering from MAC layer failures when the transmission retries are exceeded.

For the scheduler behavior, the following assumptions are considered: the set of paths that goes through a node i is denoted by P_i ; the scheduler behavior is specified by the scheduler coefficient $k_{i,p}$, which is the average serving rate of path p packets at node i ; the computation of $k_{i,p}$ takes into account the arrival rate, denoted by $\lambda_{i,p}$, the average service time $T_{i,p}$ of path p packets at node i , the probability $\beta_{i,p}$ of PHY and MAC transmission attempt failure (this takes place during the initial stage of the MAC protocol, when transmitter and receiver nodes perform the RTS/CTS handshake). Based on this, $k_{i,p}$ can be calculated as follows:

$$k_{i,p} = \begin{cases} \frac{\lambda_{i,p}}{(1 - \beta_{i,p}^m)} & \text{if } \kappa_T \leq 1 \\ \frac{\lambda_{i,p}}{(1 - \beta_{i,p}^m)} & \text{Otherwise,} \\ \kappa_T & \end{cases} \quad (3)$$

where m is an input parameter representing the maximum number of packet transmission retries at the MAC layer, and $\kappa_T = \sum_{p' \in P_j} \frac{\lambda_{i,p}}{(1 - \beta_{i,p'}^m)} T_{i,p'}$. Since the model assumes that the scheduler keeps scheduling the same packet until

it is successfully transmitted by the MAC layer, in order to compensate for the transmission failures at the MAC layer, the scheduling rate should be higher than the node arrival rate by a factor $1/(1 - \beta_{i,p}^m)$, i.e., the probability of lost packet due to exceeding MAC layer retries. On the other hand, if utilization is equal to one, some arriving packets cannot be served, but the service rate for each path is still proportional to its compensated arrival rate as given in the second line of Equation (3).

From Equation (3), one can derive the fraction of incoming traffic rate that is sent over each path. Let $h_{i,p}^-$ represent the node that precedes node i in path p . The arrival rate from path p at node i is calculated as follows:

$$\lambda_{i,p} = k_{h_{i,p}^-} (1 - \beta_{h_{i,p}^-}^m) \quad \text{for all } i, p. \quad (4)$$

This is obviously not valid when node i is the first node (i.e., the originator node) of the path, in which case $\lambda_{i,p}$ is set as an input parameter of the algorithm. A comprehensive explanation on how to compute $\beta_{i,p}$ is provided in [10]. The computation of $T_{i,p}$ is equal to the sum of four components as follows:

$$T_{i,p} = (1 - \beta_{i,p}^m) d_{i,p} + u_{i,p} + b_{i,p} + c_{i,p}, \quad (5)$$

where $d_{i,p}$ is the time spent by successful transmission of path p packets at node i (thus, it only applies when the MAC succeeds, with probability $1 - \beta_{i,p}^m$), $u_{i,p}$ is the average time consumed by the successful transmission from the neighbors of node i , $b_{i,p}$ is the average back-off time spent by node i during the transmission of path p packets, and $c_{i,p}$ is the average time spent by failed transmissions.

In order to find a consistent solution for the parameters $k_{i,p}$, $\lambda_{i,p}$, $\beta_{i,p}$ and $T_{i,p}$, the scheme uses the FPA equations as provided in [10]. The FPA structure and stopping conditions are adopted from [29].

In order to compute the network PDR at the destination node(s), represented by T , the scheme considers a set of active connections in the network, denoted by C , and the set of paths used in connection $c \in C$, denoted by P_c . Two metrics are considered, corresponding to alternative PDR objective functions:

- (i) **Average PDR.** This metric is computed as follows:

$$T = \frac{\left(\sum_{c \in C} (\sum_{p \in P_c} \lambda_{last,p}) \right)}{\left(\sum_{c \in C} (\sum_{p \in P_c} \lambda_{first,p}) \right)}, \quad (6)$$

where $\lambda_{first,p}$ and $\lambda_{last,p}$ denote the arrival rate of packets belonging to path p at the source and destination nodes, respectively. In this case, the intention is to maximize the PDR as follows:

$$\max T; \quad \text{s.t., Equation (2a), (2b) and (2c).} \quad (7)$$

- (ii) **Minimum PDR.** The PDR provided in path p_i , denoted t_{p_i} , in connection $c \in C$ is calculated as follows:

$$t_{p_i \in P_c} = \frac{\lambda_{last,p_i \in P_c}}{\lambda_{first,p_i \in P_c}}, \quad (8)$$

where $\lambda_{last, p_i \in P_c}$ is the arrival rate of the packets at the destination node, and $\lambda_{first, p_i \in P_c}$ is the arrival rate at the source node in path p_i . In this case, the optimization problem is defined as follows:

$$\max \min_{\{p_1, \dots, p_{|P_c|}\}} t_{p_i \in P_c}; \text{ s.t., Equation (2a), (2b) and (2c).,} \tag{9}$$

which corresponds to maximizing the lowest PDR, i.e., the PDR of the flow that gets the lowest grade of service.

Differently from [10], where Automatic Differentiation (AD) was used to reach the optimal PDR, the proposed scheme has two optimization objectives, which prompted the use of NSGA-II, as explained in the following section.

V. NSGA-II BASED UAV PLACEMENT AND ROUTING OPTIMIZATION

NSGA-II [9] is an elitist MOEA, which comprises two main procedures: Pareto ranking and diversity preservation. Pareto ranking aims to sort the population into different non-domination levels (i_{rank}) in ascending order. Here, the lowest ranking level contains the best set of solutions. On the other hand, diversity preservation is used to maintain a good spread of solutions in the obtained solution set. Members of each non-dominated front are assigned a value called *crowding distance* ($i_{distance}$), which is used to determine the density of solutions surrounding a particular solution in the population. In order to distinguish the best solutions, NSGA-II uses the *crowded-comparison* operator, denoted by \prec_n . The operator assumes that every solution i in the population has two attributes: i_{rank} and $i_{distance}$. The partial order \prec_n is defined as follows:

$$i \prec_n j \text{ if } (i_{rank} < j_{rank}) \text{ or } [(i_{rank} = j_{rank}) \text{ and } (i_{distance} > j_{distance})]. \tag{10}$$

Assuming initial population of size N , Algorithm 1 shows the main loop of NSGA-II proposed by the authors in [9], where the calls to the routines *fast-non-dominated-sort* (R_t) and *crowding-distance-assignment* (\mathcal{F}_i) correspond respectively to the Pareto ranking and diversity preservation procedures described above. R_t has size equal to $2N$, being formed by combining parent S_t and offspring Z_t populations. \mathcal{F}_i refers to the i^{th} front or level.

The chromosome structure X represents the 3D coordinates of a set of UAVs deployed inside the convex hull area a' . Therefore, the chromosome is represented as follows:

$$X = [(q_{(x_1, y_1, h_1)}^{u_1}), (q_{(x_2, y_2, h_2)}^{u_2}), \dots, (q_{(x_n, y_n, h_n)}^{u_n})].$$

In order to find optimal solutions, NSGA-II must be able to compute the values of two objectives from the above representation of the chromosome. The first objective is directly given by n , representing the number of UAVs forming the deployed network. It also represents the size of the chromosome, which is variable. As for the second objective, i.e., the PDR, the scheme runs an embedded GA (inner-GA) in order

Algorithm 1 NSGA-II Main Loop

- 1: $R_t = S_t \cup Z_t$
- 2: $\mathcal{F} = \text{fast-non-dominated-sort}(R_t)$
- 3: $S_{t+1} = \emptyset$ and $i = 1$
- 4: **Until** $|S_{t+1}| + \mathcal{F}_i \leq N$
 - 4.1. *crowding-distance-assignment*(\mathcal{F}_i)
 - 4.2. $S_{t+1} = S_{t+1} + \mathcal{F}_i$
 - 4.3. $i = i + 1$
- 5: **Sort**(\mathcal{F}_i, \prec_n)
- 6: $S_{t+1} = S_{t+1} \cup \mathcal{F}_i[1 : (N - |S_{t+1}|)]$
- 7: $Z_{t+1} = \text{make-new-population}(S_{t+1})$
- 8: $t = t + 1$

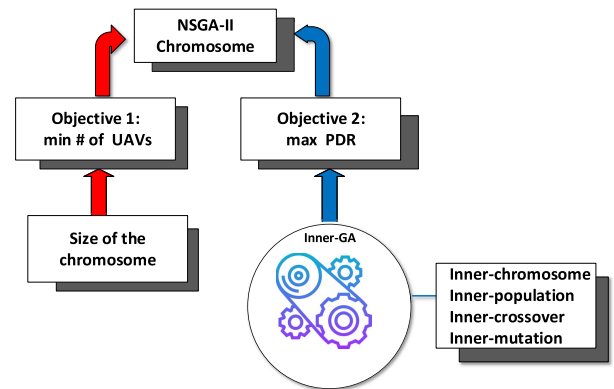


FIGURE 2. Representation of the objective function computation.

to find the best routes from source GNs to the destination GNs, which lead to a higher value of PDR for the current placement of the UAVs. The value of the second objective will be the highest PDR found by the inner-GA, which corresponds to the best set of routing paths for a given NSGA-II individual. Fig. 2 depicts the idea behind the computation of the objective functions.

A. INNER-GA CHROMOSOME ROUTING REPRESENTATION

The inner-GA will have its own set of parameters as presented next: Given the UAVs' positions from the NSGA-II chromosome, the inner-chromosome will be a set of hash tables of variable size, each of which representing a flow from a source GN to a destination GN. For a given set of UAVs occupying specific positions in the network, a source GN may be able to reach the same destination GN through different paths. Therefore, the scheme uses those paths to distinguish different inner-chromosomes as they would represent different flows. As already stated, the communication between GNs is realized through UAVs in a multihop fashion. Fig. 3 and Fig. 4 show a hypothetical communication network with the flows of data and the corresponding inner-chromosome representation, respectively.

Each key in the inner-chromosome represents a unique ID of a node in the network, and the stored value corresponds to

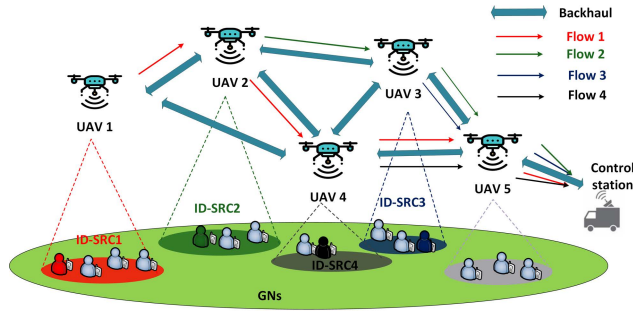


FIGURE 3. Hypothetical communication network and data flow.

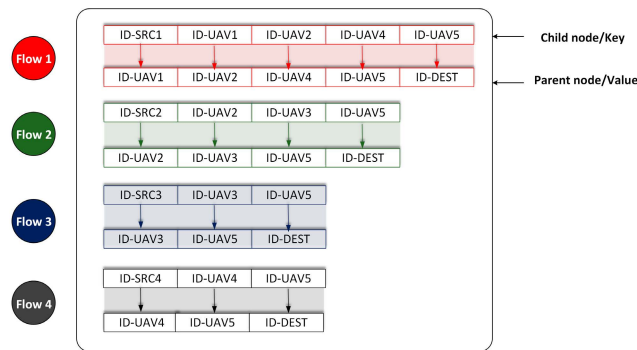


FIGURE 4. Inner-chromosome represented as hash table with key-value association.

the ID of the subsequent node in the downstream direction, i.e., toward the destination node.

B. INNER-GA PARAMETERS AND GENETIC OPERATORS

The inner-initial-population is a random generation of N_{inner} inner-chromosomes. The fitness of each inner-chromosome is evaluated by Equation (6), when average PDR is being used. Best individuals (i.e., inner-chromosomes with high score) go to the mating pool.

The inner-crossover occurs with probability $p_{c_{inner}}$ and is performed by exchanging a subset of flows with same source-destination pair between two inner-chromosomes. This is depicted in Fig. 5, taking into account the network represented in Fig. 3. The exchanging point – which corresponds to the number of exchanged flows – is chosen randomly within the number of existing flows. The resulting offspring after exchanging 4 flows is shown in Fig. 6.

As regards to inner-mutation, for each inner-chromosome, a flow is randomly chosen, for which a new route from the source to the respective destination node is generated. Breadth First Search (BFS) algorithm is used to generate a set of routes. Then, the new path is randomly chosen from the generated set, and the old path is replaced with the new one with a probability $p_{m_{inner}}$. This procedure is shown in Fig. 7.

C. NSGA-II PARAMETERS AND GENETIC OPERATORS

Regarding the initial population of NSGA-II, a set of N randomly generated chromosomes is used to form the initial population. The length of the chromosomes in the population

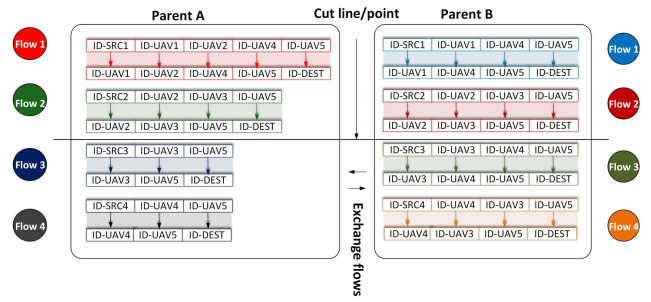


FIGURE 5. Inner-crossover.

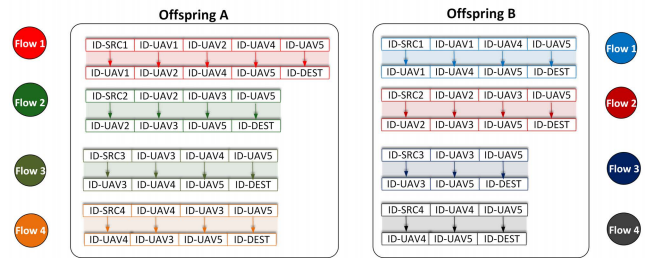


FIGURE 6. Inner-offspring.

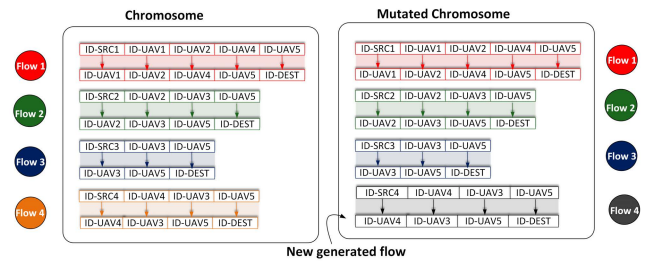


FIGURE 7. Inner-mutation.

may be different from each other depending on the number of deployed UAVs to cover all GNs. In this study, the size of each chromosome and the best solution found by the inner-GA respectively correspond to the number of UAVs and PDR in a given topology. The NSGA-II selection operation uses binary tournament selection based on the *crowded-comparison* operator \prec_n , in order to choose the best chromosome following Equation (10). Here, given two solutions with differing non-domination ranks, NSGA-II prefers the solution with the lowest (i.e., best) rank. Otherwise, if both solutions belong to the same front, then NSGA-II prefers the solution that is located in a less crowded region.

The genetic operators (crossover and mutation) are implemented in a way similar to our previous work [12], where the crossover (with the probability p_c) between two chromosomes is performed by finding a midpoint in a' and drawing (diagonally in 45/-45 degrees or horizontally or vertically) a cutting line to divide the area in two parts in each chromosome. Next, the operator removes all UAVs that are within $\frac{1}{2}D$ distance radius along the cutting line within a' . If the separation line is either diagonally or vertically drawn, the leftmost part of one parent is joined with the rightmost part

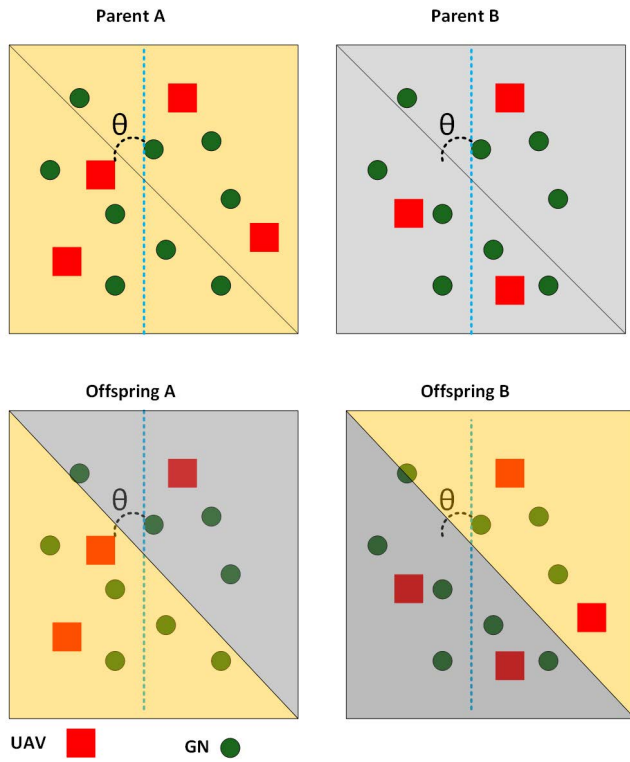


FIGURE 8. NSGA-II crossover.

of the other to form an offspring. On the other hand, if it is horizontally drawn, the uppermost and bottommost will be joined instead. Since the operator has removed some UAVs, there may be some uncovered GNs in the vicinity of the separation line, which makes the resulting offspring an invalid individual. In this case, the operator repairs the offspring by repeatedly choosing a random uncovered GN and placing an UAV in the closest available point $q_{(x,y,h)}^u$ until all GNs are covered. UAVs which are not serving or bridging any GNs are removed.

As regards to the mutation operator, for each chromosome, an UAV is randomly selected with a probability p_m , then it is either temporarily removed or reallocated to a new randomly chosen available point $q_j \in Q$ with a probability of 50%. If the above procedures fail to produce a valid individual, then the UAV is put back in its initial position. Examples of crossover and mutation operations are depicted in Fig. 8 and Fig. 9, respectively.

D. COMPLEXITY ANALYSIS

In [9], the authors propose NSGA-II and estimate its performance as $O(MP^2)$, where M is the number of objectives and P is the size of the population. This estimate assumes that complexity is determined by the non-dominated sorting part. While being useful for comparison with other MOEA under generic objective functions, such analysis is not suitable when the objective functions are themselves the most significant source of complexity. On the other hand, the complexity analysis of objective functions becomes particularly difficult

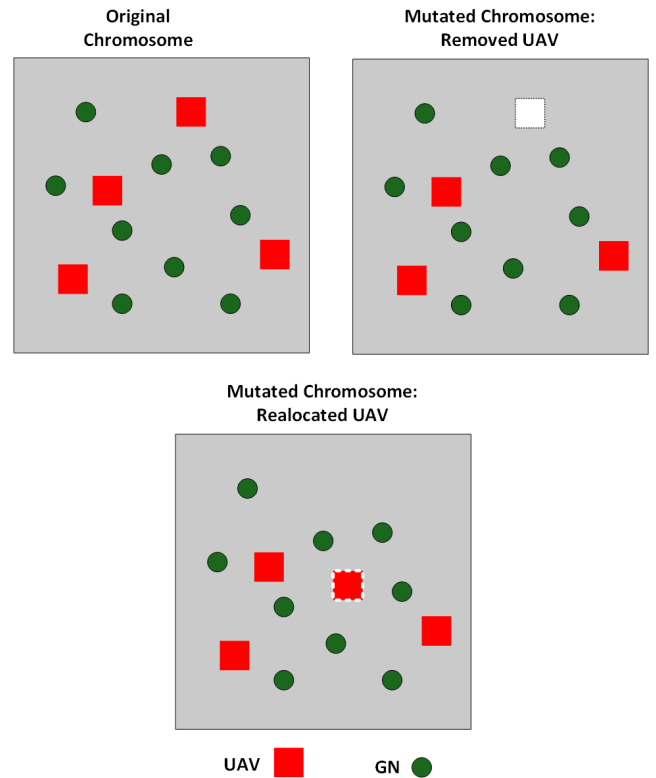


FIGURE 9. NSGA-II mutation.

when the chromosome has variable length – as is the case of the proposed scheme –, though a worst case characterization is possible.

The proposed topology optimization scheme has a nested algorithm structure, in which NSGA-II forms the outer layer. One of the NSGA-II objective functions (PDR maximization) is particularly complex, making use of the inner-GA to find the best routing, i.e., the one that maximizes the PDR for a given UAV topology. In turn, the PDR objective function of the inner-GA is calculated by means of the FPA proposed in [10]. It is in this nested objective function that resides most of the time complexity of the algorithm. The number of UAVs ($|U|$) corresponds to the size of the NSGA-II chromosome, and significantly affects the performance of the FPA. As already seen, since $|U|$ constitutes one of the NSGA-II objective functions, it is a variable, making it more difficult to estimate the complexity of NSGA-II. As such, in the following analysis it is considered that the number of UAVs is fixed and equal to $|U|^{max}$, corresponding to the maximum allowed number of UAVs (an input parameter of the algorithm).

The FPA updates the values of the variables based on the values calculated in the previous iteration. Considering only scalar operations, the most complex updates involve four nested cycles, where two iterate over the number of nodes ($N = |U|^{max} + |V|$) and two over the number of flows or paths ($|F|$). Such cases correspond to variables θ , β , r , w and z in [10]. Each FPA execution will finish when

the defined convergence criteria are met, which may take a variable number of iterations I_{FPA} . Nevertheless, in order to allow a complexity estimate, I_{FPA} can be fixed as the expected worst case. Based on this assumption, FPA complexity is estimated as $O(I_{FPA} \cdot N^2 \cdot |F|^2)$.

Based on the above result, knowing that, in each inner-GA generation, the number of new offspring is proportional to the size of the population ($P_{innerGA}$), and that the FPA must be run for each new individual, the complexity of the inner-GA can be estimated as $O(G_{innerGA} \cdot P_{innerGA} \cdot I_{FPA} \cdot N^2 \cdot |F|^2)$, where $G_{innerGA}$ is the configured number of generations of the inner-GA. Since NSGA-II is also a GA, a similar reasoning can be applied to estimate its complexity. Knowing that in each NSGA-II generation the number of new offspring is proportional to the size of the population (P_{NSGAII}), and that the inner-GA must be run for each new individual, the complexity of the NSGA-II algorithm can be estimated as $O(G_{NSGAII} \cdot P_{NSGAII} \cdot G_{innerGA} \cdot P_{innerGA} \cdot I_{FPA} \cdot N^2 \cdot |F|^2)$, where G_{NSGAII} is the configured number of generations of NSGA-II.

From the above analysis, it becomes obvious that the simulation duration is highly influenced by the time or number of iterations/generations needed by the algorithms to converge, namely the stopping conditions, as well as the initial guesses. Depending on the value of the tolerated error, the FPA will stop before convergence if the value is set too high, or there will be useless additional iterations if the value is set too low. Another important factor is the initial guess of the FPA: when the algorithm departs closer to the fixed-point, the convergence is faster, thus the challenge is to make a good guess during initialization. As regards to the GAs, the number of generations may be fixed based on empirical studies, or variable depending on more flexible stopping criteria, for example the rate of change of the population over a number of generations. A parameter that conditions the required number of NSGA-II generations is the degree of granularity of the search space, which in this scheme corresponds to the number of candidate placement points within the convex hull. The latter depends on μ : smaller values of μ create larger search space, and hence a higher value of G_{NSGAII} is required to converge.

VI. SIMULATION PARAMETERS AND RESULTS

This section presents simulation results of the proposed NSGA-II implementation. Simulations were performed on a virtual machine running GNU/Linux (ubuntu) OS, x86_64 architecture, 64 bits CPU, 1.992 GHz CPU clock speed, 4 cores per socket, 1 socket and 1.6 Gbits of RAM. In order to implement the algorithm, C++ was used as the programming language.

The network comprises 24 GNs uniformly distributed in a $500 \times 500 \text{ m}^2$ area. The maximum transmission range \mathcal{D} is set to 100 m and $\mu = 0.40$. It should be recalled that the PDR computation uses an FPA. Similarly to [10], in order to make the convergence faster, the FPA initializes the values of the parameters assuming communication is perfect for every

TABLE 2. Default parameters.

Number of GNs	24
\mathcal{A}	$500 \times 500 \text{ m}$
μ	0.40
\mathcal{D}	100
Flight altitude	80 m
\mathbf{L}	{30, 60, 90, 120, 150} kbps

TABLE 3. NSGA-II and inner-GA parameters.

NSGA-II maximum generations	40
NSGA-II population size	60
NSGA-II crossover probability (p_c)	0.7
NSGA-II mutation probability (p_m)	0.4
Inner-GA maximum generations	30
Inner-GA population size	60
Inner-GA crossover probability ($p_{c_{inner}}$)	0.7
Inner-GA mutation probability ($p_{m_{inner}}$)	0.2

connection, thus time $T_{i,p}$ consists only of the time taken by successful transmissions, plus the back-off time needed for the first trial. Moreover, since perfect channel conditions are assumed, the probabilities of failure are initialized to 0 in all the links. As the FPA algorithm runs, the values of these variables change and converge to the approximate or real ones. It is considered that all nodes generate traffic simultaneously. Simulations are performed with different load values given by the set $\mathbf{L} = \{30, 60, 90, 120, 150\}$ kbps.

Simulations take into account two scenarios with two objectives each:

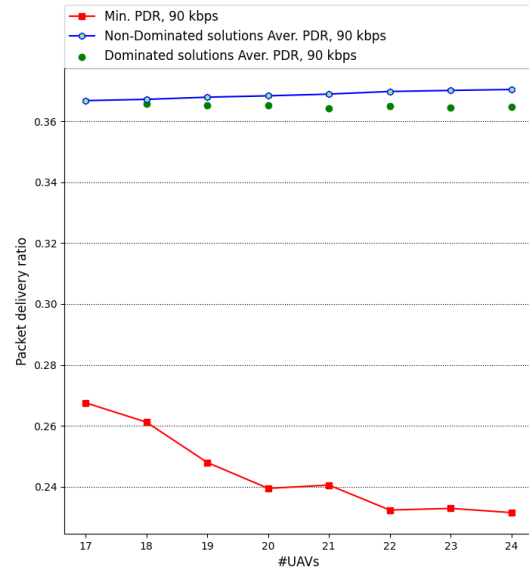
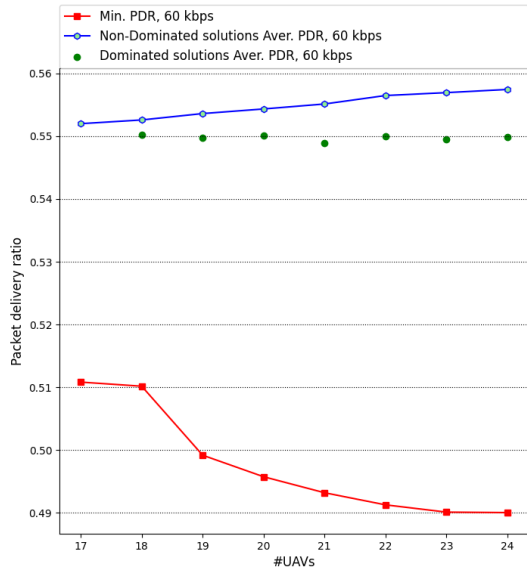
- *Scenario (A)*: aims (1) to minimize the number of UAVs, and (2) to maximize the average PDR at the destination node(s).
- *Scenario (B)*: aims (1) to minimize the number of UAVs, and (2) to maximize the minimum PDR at the destination node(s).

In order to assess the impact of having more than one sink/destination node, these scenarios were simulated with two different traffic patterns:

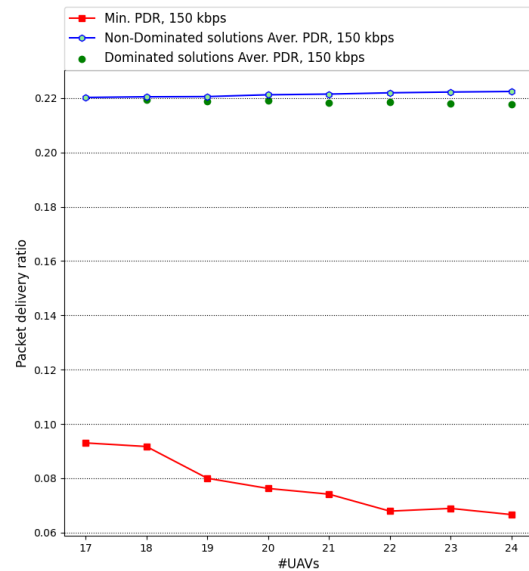
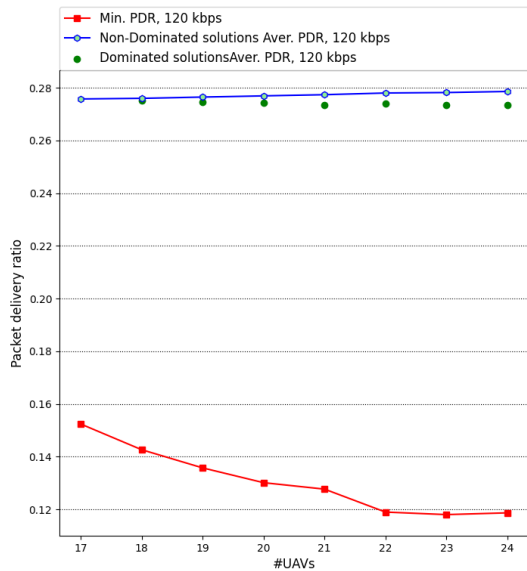
- All traffic from the GNs is sent to a single destination node with $ID = 0$. This node v_0 is placed at 2D coordinates $q_{(0,0)}^{v_0}$.
- Twelve (half) randomly chosen GNs generate traffic towards destination node with $ID = 0$ placed at $q_{(0,0)}^{v_0}$. The other twelve GNs generate traffic towards destination node with $ID = 25$ placed at $q_{(500,500)}^{v_{25}}$.

In the UAV placement algorithm presented in [12], the use of different altitudes (in the altitude range of [40,120] meters) did not affect the performance, as the communication range was quite long when compared to the maximum allowed flight altitude. Based on these results, and since this work does not address altitude optimization, a fixed flight altitude of 80 m is assigned to the UAVs. Table 2 shows the summary of the global/default parameters.

As already stated, an important and challenging step when designing a GA is defining the stopping criteria, i.e., the point where the algorithm should stop executing. There have been several studies on the subject [30], [31]. Differently from the



(a) Pareto front for # UAVs vs average PDR, and minimum PDR for 60 kbps. (b) Pareto front for # UAVs vs average PDR, and minimum PDR for 90 kbps.



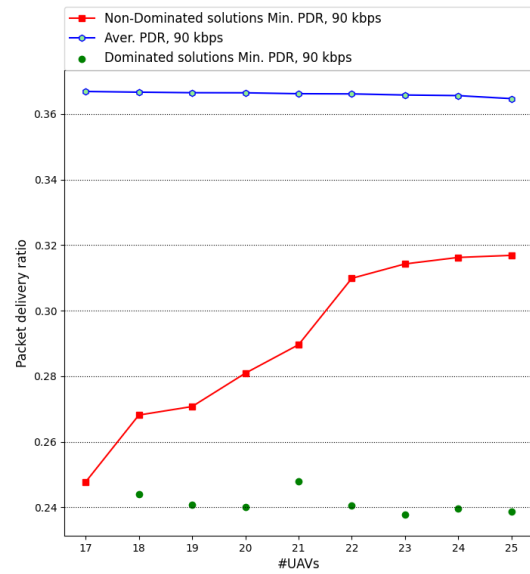
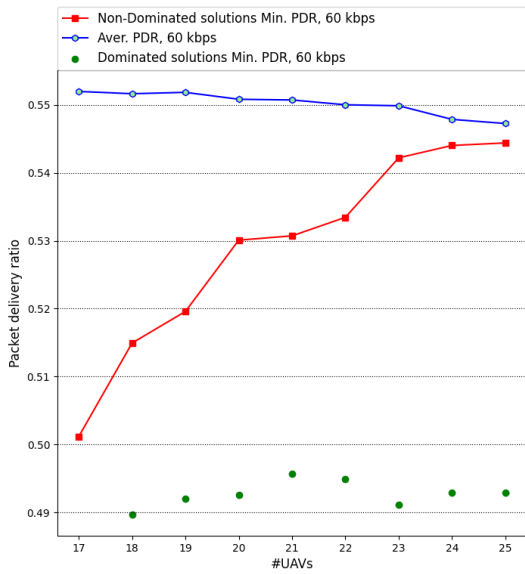
(c) Pareto front for # UAVs vs average PDR, and minimum PDR for 120 kbps. (d) Pareto front for # UAVs vs average PDR, and minimum PDR for 150 kbps.

FIGURE 10. Scenario (A) for single destination node.

present proposal, most of these studies consider fixed size and string based chromosomes, hence, their stopping criteria are not directly applicable to the present GA scheme, which uses variable length and non-string based chromosomes. In [12], the authors adopted the rule of stopping when there is no significant improvement during the last ten iterations. However, as the current GA proposal entails nested algorithms (i.e., inner-GA and FPA), that would add more complexity and an additional burden on computational resources. As such, the stopping criteria is set as a fixed number of

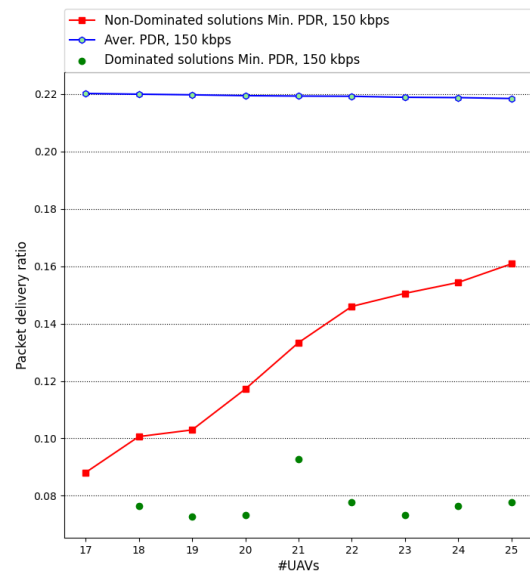
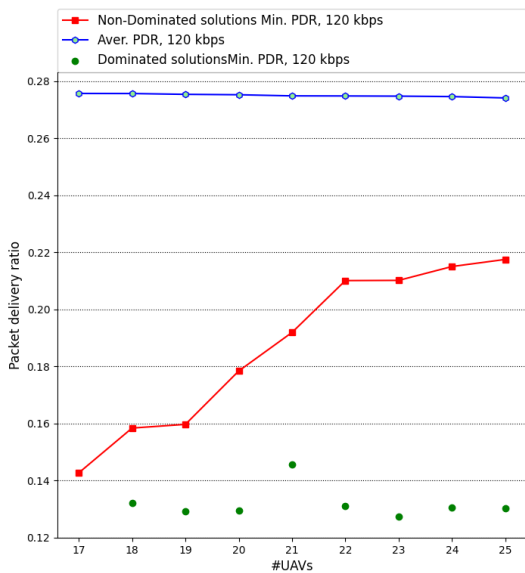
generations. In order to determine the number of generations, multiple simulations were performed to adjust the NSGA-II parameters, such as p_c , p_{cinner} , p_m , and p_{minner} to avoid excessive computation, as well as premature convergence. Table 3 shows the summary of the NSGA-II and inner-GA parameters. Parameters such as population sizes and genetic operator probabilities were not optimized and thus can be a subject of further studies.

In each scenario, and for each load value taken from the set \mathbf{L} , 25 runs of NSGA-II with different seeds,



(a) Pareto front for # UAVs vs minimum PDR, and average PDR for 60 kbps.

(b) Pareto front for # UAVs vs minimum PDR, and average PDR for 90 kbps.



(c) Pareto front for # UAVs vs minimum PDR, and average PDR for 120 kbps.

(d) Pareto front for # UAVs vs minimum PDR, and average PDR for 150 kbps.

FIGURE 11. Scenario (B) for single destination node.

[1, 2, ..., 25], were performed. In each simulation, the values of the following parameters were collected: optimized average and minimum PDR for each number of deployed UAVs, and the time spent in each run. For each 25 runs, the average and minimum PDR with 95% confidence interval were computed.

In some circumstances, the maximization of the average PDR might imply that some flows in the network present a degraded service quality. On the other hand, the maximization of the minimum PDR may cause the average PDR to drop

drastically. However, the choice of either to maximize the average or minimum PDR is scenario specific, i.e., the latter takes into account the improvement of individual PDRs of the flows for fairness consideration, and the former takes into account the overall PDR for efficiency consideration. In order to visualize such behaviors, the simulation results are presented as follows: for each load value in L , a pair of results is captured, namely, the parameters to be optimized for the specific scenario, i.e., *Scenario (A)* or *Scenario (B)*, and the behavior of the minimum or average PDR when it is not

being optimized, respectively. *Scenario (A)* and *Scenario (B)* are considered together with single or multiple destination nodes, as presented in the following sections.

A. SIMULATION RESULTS CONSIDERING SINGLE DESTINATION

This section presents the performance of the proposed NSGA-II for *Scenario (A)* in Fig. 10, and for *Scenario (B)* in Fig. 11. Results for the initial load, $L_0 = 30$ kbps, are not presented for space reasons. From the simulation results, it was observed that the valid solutions that were found have a minimum of 17 UAVs in both scenarios.

In Fig. 10 (*Scenario (A)*), the trade-off between the solutions is clear. One may have fewer UAVs with lower PDR, or more UAVs with higher PDR. In fact, having fewer UAVs means that some nodes may have to carry more traffic than the others (i.e., the aggregation of flows through the same or neighboring routes is higher), leading to poor performance in terms of PDR. It is also observable that, despite the increase of the average PDR, as the number of UAVs increases, the minimum PDR decreases (recall that this is not an objective function in *Scenario (A)*), i.e., some flows experience starvation at the cost of an overall higher PDR. This scenario becomes advantageous when one cares about the average PDR of the GNs as a whole, rather than fairness among flows.

For *Scenario (B)* (see Fig. 11), the trade-off between the PDR metrics is also observed: the average PDR tends to decrease as the minimum PDR increases. The maximum PDR used in this scenario is advantageous when one cares about the fairness among flows, rather than the overall PDR in the network, i.e., it matters to guarantee a minimum acceptable performance for each flow.

The performance of NSGA-II with inner-GA was also compared with NSGA-II using Dijkstra based routing. Dijkstra finds the shortest path between source and destination nodes, which is the route selection criterion employed by default in routing protocols. It can be observed that inner-GA was able to find alternative paths from the sources to the destination, and slightly outperformed shortest path in both scenarios, i.e., it maximizes the PDR metric while minimizing the number of UAVs in a range between [9.5%-37.52%] for *Scenario (A)* (see Fig. 12), and [8.7%-31.93%] for *Scenario (B)* (see Fig. 13). This only applies to load values above 30 kbps, since the latter leads to a PDR of 100% in all cases.

The PDR tends to decrease as the load increases, which is also observed in Fig. 12 and Fig. 13. This complies with the expected behavior, since higher load causes additional contention at the MAC layer.

B. SIMULATION RESULTS CONSIDERING MULTIPLE DESTINATIONS

Following the same structure of Section VI-A, this section will firstly present the results for *Scenario (A)*, and then the results for *Scenario (B)*. Two destination nodes are now considered, located at coordinates $q_{j_0} = (0, 0)$ and

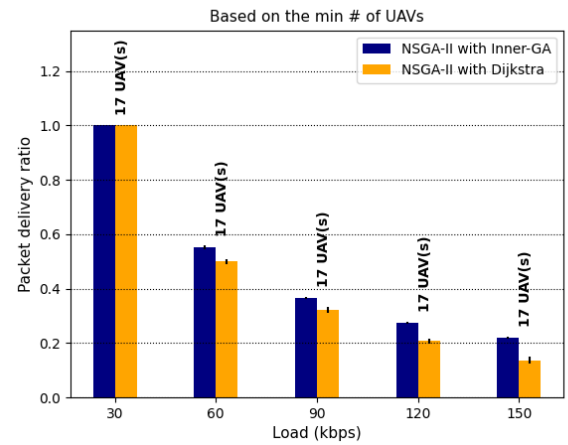


FIGURE 12. Single destination: comparison between inner-GA and Dijkstra performance for *Scenario (A)*.

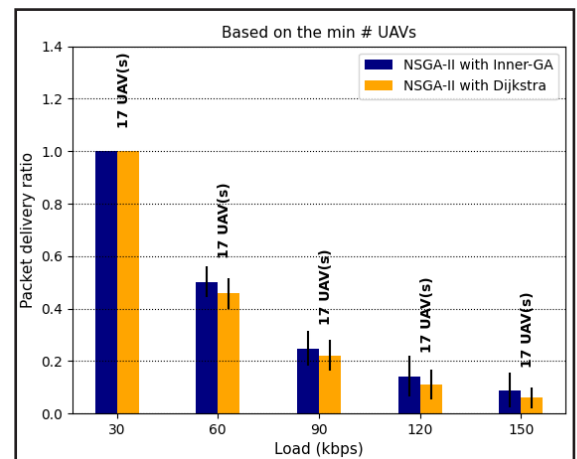
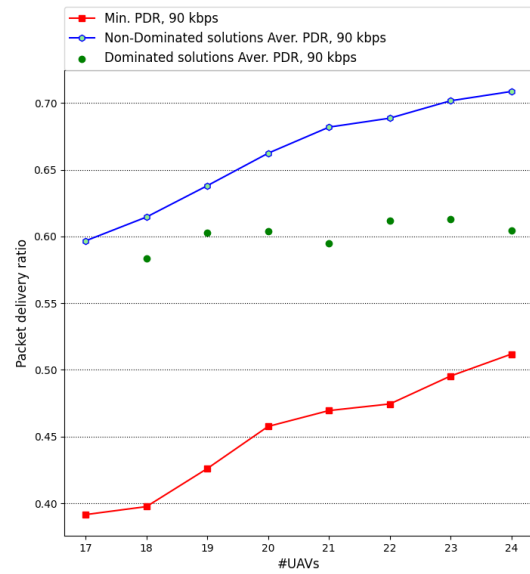
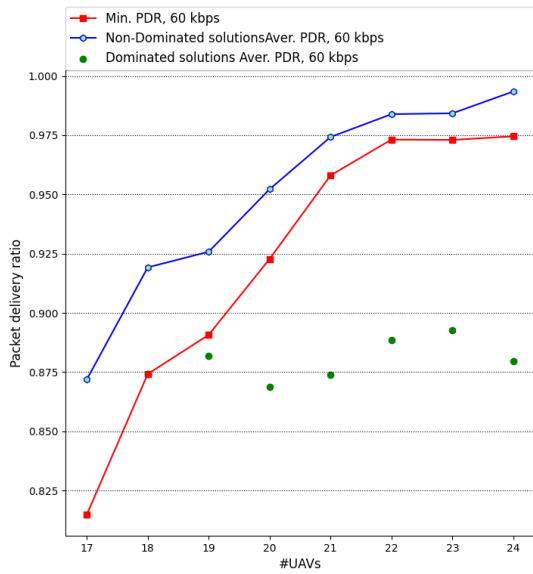


FIGURE 13. Single destination: comparison between inner-GA and Dijkstra performance for *Scenario (B)*.

$q_{j_1} = (500, 500)$, corresponding to opposite corners in the deployment area.

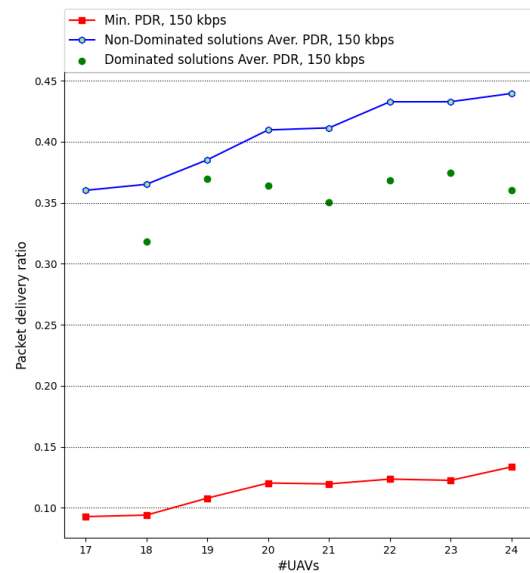
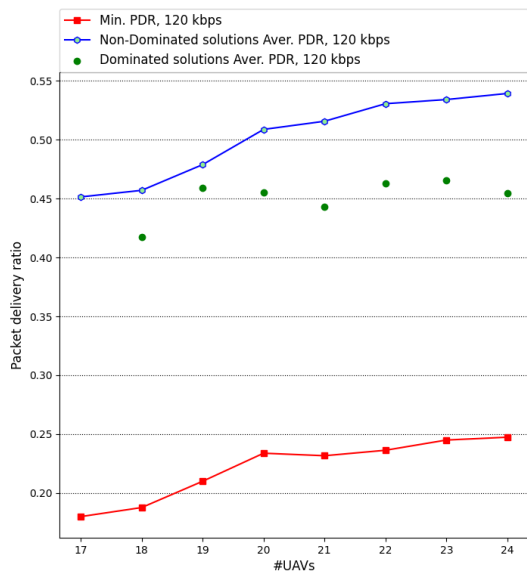
In Fig. 14 (*Scenario (A)*), and differently from the single destination scenario (see Fig. 10), the results show an increase of the minimum PDR (not an objective function in *Scenario (A)*) as the average PDR increases, i.e., the previously observed conflicting behavior is not present. This is justified by the fact that data traffic is now split between two destination nodes located at opposite coordinate points in the target area. This configuration has the following advantages:

- (i) Reduces the chance of having some nodes relaying data belonging to many different flows, thus avoiding potential bottlenecks;
- (ii) Potentially reduces the degree of source-destination starvation, as the number of immediate interfering neighboring flows is reduced. This is a well known problem in multi-hop ad hoc wireless networks, also known as Flow In the Middle (FIM) problem [32]. In fact, the inner-GA tries to find paths that suffer less from FIM, in order to maximize the PDR.



(a) Pareto front for # UAVs vs average PDR, and minimum PDR for 60 kbps.

(b) Pareto front for # UAVs vs average PDR, and minimum PDR for 90 kbps.



(c) Pareto front for # UAVs vs average PDR, and minimum PDR for 120 kbps.

(d) Pareto front for # UAVs vs average PDR, and minimum PDR for 150 kbps.

FIGURE 14. Scenario (A) for two destination nodes.

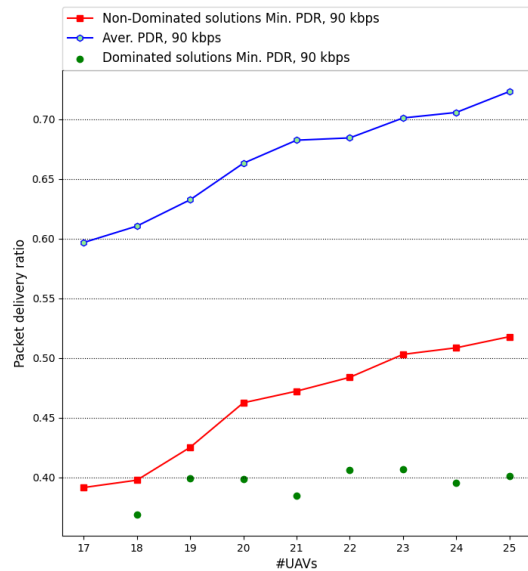
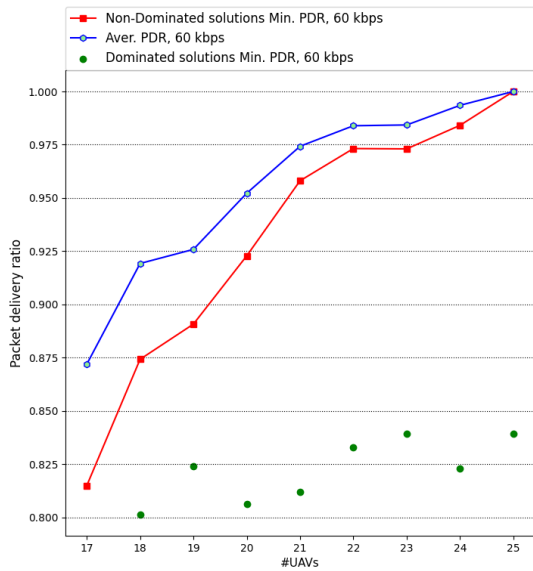
In general, this leads to increased load balancing and thus overall capacity, which is translated into higher PDR in both metrics. The performance improvement gained by having one additional destination node for each load value is shown in Table 4.

Similarly, in *Scenario (B)* (see Fig. 15), the results show an increase of both the minimum and average PDR (not an objective function in *Scenario (B)*) with the increase of the number of UAVs. Table 5 presents the performance

improvement ranges from single to multiple destinations. At 30 kbps there is no improvement, since the latter leads to a PDR of 100% in all cases.

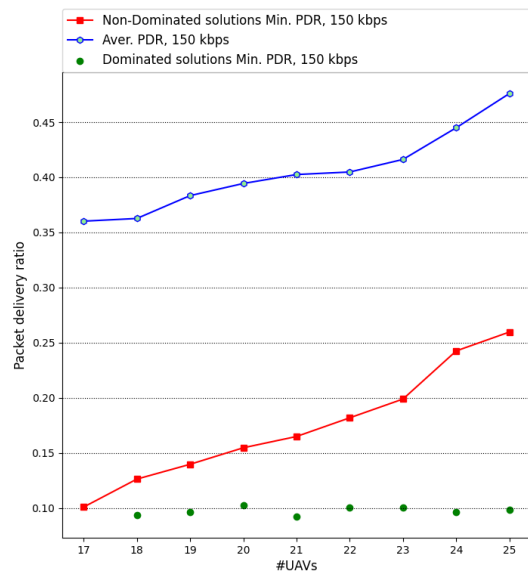
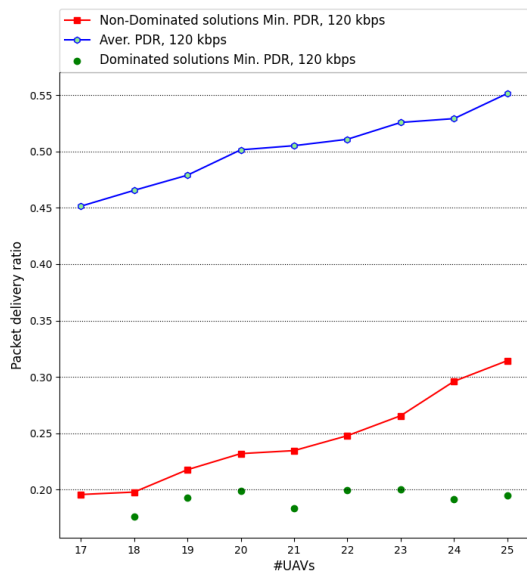
The previous results allow the conclusion that having multiple destinations placed apart from each other is beneficial regarding the PDR.

Fig. 16 and Fig. 17 present the performance of inner-GA over Dijkstra for the average PDR [9.50%-34.18%] and maximum PDR [9.30%-36.72%], respectively. The addition of the



(a) Pareto front for # UAVs vs average PDR, and minimum PDR for 60 kbps.

(b) Pareto front for # UAVs vs average PDR, and minimum PDR for 90 kbps.



(c) Pareto front for # UAVs vs average PDR, and minimum PDR for 120 kbps.

(d) Pareto front for # UAVs vs average PDR, and minimum PDR for 150 kbps.

FIGURE 15. Scenario (B) for two destination nodes.

second destination node also improved the PDR of Dijkstra. However this was not enough to outperform the inner-GA. Again, for a load of 30 kbps, the PDR is 100% in both cases.

C. SIMULATION PERFORMANCE

As already noted, accurately estimating the complexity and simulation time of a MOEA is not trivial, particularly when dealing with chromosomes of variable length. Empirical results can provide additional insights on the factors that determine these metrics. The average time spent running the

TABLE 4. Performance improvement from single to multiple destination nodes for min # UAVs and max average PDR.

Load	Interval [min.- max.]
30 kbps	—
60 kbps	[36.68%-43.89%]
90 kbps	[38.51%-47.71%]
120 kbps	[38.91%-48.33%]
150 kbps	[38.86%-49.39%]

simulations in different scenarios (see Fig. 18 and Table 6 for clarity) was measured. In general, simulations with multiple

TABLE 5. Performance improvement from single to multiple destination nodes for min # UAVs and maximin PDR.

Load	Interval [min.- max.]
30 kbps	—
60 kbps	[38.49%-45.55%]
90 kbps	[32.56%-39.25%]
120 kbps	[15.23%-30.82%]
150 kbps	[12.73%-38.06%]

TABLE 6. Average simulation time.

# destination	max average PDR	maximin PDR
Single	14907.21 s	15051.00 s
Multiple	13307.25 s	14302.99 s

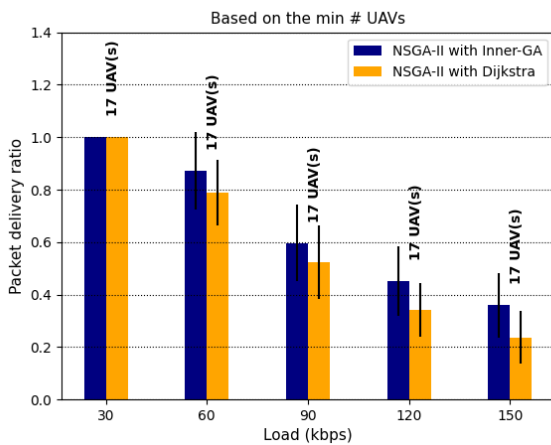


FIGURE 16. Multiple destinations: comparison between inner-GA and Dijkstra performance for Scenario (A).

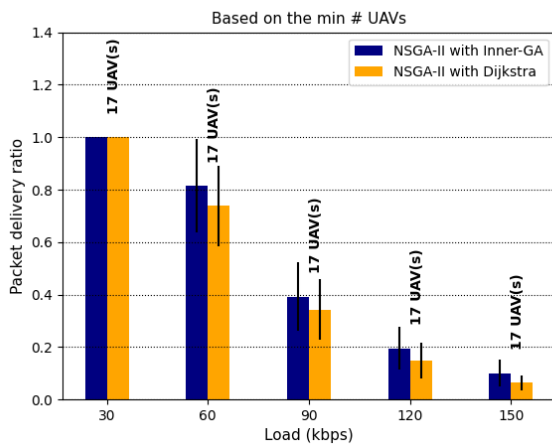


FIGURE 17. Multiple destinations: comparison between inner-GA and Dijkstra performance for Scenario (B).

destinations take less time to converge when compared to the single destination scenario. This is due to the reduction of inter-flow interference, leading to faster FPA convergence. Recall that FPA initialization assumes no contention between flows, which is a closer guess when load is low. It is also

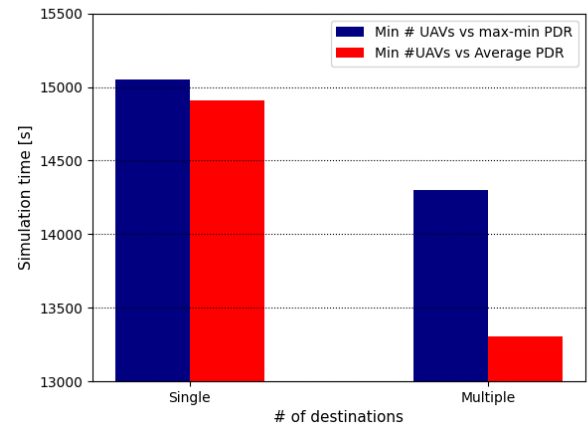


FIGURE 18. Average simulation time for the given simulation environment.

observed that maximizing the average PDR converges relatively faster than maximizing the minimum PDR.

VII. CONCLUSION

This paper presents a joint UAV placement and flow routing optimization scheme based on a nested architecture, where NSGA-II algorithm forms the outer layer, while a single objective GA is used as inner layer. Link performance is estimated by an FPA taking into account MAC layer contention. Significant insights were only possible due to this feature. Two scenarios, each with two optimization objectives, were considered: (1) minimization of the number of UAVs while maximizing the average PDR, and (2) minimization of the number of UAVs while maximizing the minimum PDR. From the simulation results, it was observed that the proposed algorithm is able to provide meaningful Pareto curves, determining a set of non-dominated solutions for UAV placement. Based on the latter, the decision-making entities can choose the one that best fits the application conditions and mission management strategy at hand. Additionally, the performance of the inner-GA routing optimization was compared against the Dijkstra shortest path routing. The results show that the inner-GA achieves better performance due to its capacity of performing load balancing to reduce MAC contention. This advantage is even more significant in complex scenarios comprising more than one sink node. The simulation performance was also studied. It was concluded that scenarios with multiple sink nodes tend to reduce flow interference, speeding up convergence of the FPA algorithm. In general, the average PDR objective function leads to faster convergence than maximin PDR.

In future work, we intend to evaluate the proposed scheme when replacing NSGA-II and inner-GA with other alternative algorithms. We are currently investigating alternative problem formulations that optimize UAV displacement vectors instead of positions, making use on GN movement prediction. Finally, we plan to reduce the overall complexity of the scheme by performing heuristic based adjustments to UAV

positions, as is done in [23], or by using multi-agent Deep Reinforcement Learning, where local adjustment decisions are negotiated among neighboring UAVs. Machine Learning models also have the potential to replace the inner-GA and FPA, trading-off efficiency versus accuracy.

REFERENCES

- [1] S. Morgenthaler, T. Braun, Z. Zhao, T. Staub, and M. Anwander, "UAVNet: A mobile wireless mesh network using unmanned aerial vehicles," in *Proc. IEEE Globecom Workshops*, Dec. 2012, pp. 1603–1608.
- [2] E. Arribas, V. Mancuso, and V. Cholvi, "Coverage optimization with a dynamic network of drone relays," *IEEE Trans. Mobile Comput.*, vol. 19, no. 10, pp. 2278–2298, Oct. 2020.
- [3] D.-Y. Kim and J.-W. Lee, "Joint mission assignment and topology management in the mission-critical FANET," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 2368–2385, Mar. 2020.
- [4] A. Dai, R. Li, Z. Zhao, and H. Zhang, "Graph convolutional multi-agent reinforcement learning for UAV coverage control," in *Proc. Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Oct. 2020, pp. 1106–1111.
- [5] W. Du, W. Ying, P. Yang, X. Cao, G. Yan, K. Tang, and D. Wu, "Network-based heterogeneous particle swarm optimization and its application in UAV communication coverage," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 4, no. 3, pp. 312–323, Jun. 2020.
- [6] C. Esposito and G. Rizzo, "Help from above: UAV-empowered network resiliency in post-disaster scenarios," in *Proc. IEEE 19th Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2022, pp. 477–480.
- [7] C.-C. Lai, L.-C. Wang, and Z. Han, "The coverage overlapping problem of serving arbitrary crowds in 3D drone cellular networks," *IEEE Trans. Mobile Comput.*, vol. 21, no. 3, pp. 1124–1141, Mar. 2022.
- [8] M. Stellin, S. Sabino, and A. Grilo, "LoRaWAN networking in mobile scenarios using a WiFi mesh of UAV gateways," *Electronics*, vol. 9, no. 4, p. 630, Apr. 2020.
- [9] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
- [10] J. S. Baras, V. Tabatabaee, G. Papageorgiou, and N. Rentz, "Performance metric sensitivity computation for optimization and trade-off analysis in wireless networks," in *Proc. IEEE Global Telecommun. Conf. (IEEE GLOBECOM)*, 2008, pp. 1–5.
- [11] S. Sabino and A. Grilo, "NSGA-II based joint topology and routing optimization of mesh networks with flying access points," *Proc. Comput. Sci.*, vol. 160, pp. 165–172, Jan. 2019.
- [12] S. Sabino, N. Horta, and A. Grilo, "Centralized unmanned aerial vehicle mesh network placement scheme: A multi-objective evolutionary algorithm approach," *Sensors*, vol. 18, no. 12, p. 4387, Dec. 2018.
- [13] P. Basu, J. Redi, and V. Shurbanov, "Coordinated flocking of UAVs for improved connectivity of mobile ground nodes," in *Proc. Mil. Commun. Conf. (IEEE MILCOM)*, vol. 3, Oct./Nov. 2004, pp. 1628–1634.
- [14] M. Di Felice, A. Trotta, L. Bedogni, K. R. Chowdhury, and L. Bononi, "Self-organizing aerial mesh networks for emergency communication," in *Proc. IEEE 25th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Sep. 2014, pp. 1631–1636.
- [15] B. Galkin, J. Kibilda, and L. A. DaSilva, "Deployment of UAV-mounted access points according to spatial user locations in two-tier cellular networks," in *Proc. Wireless Days (WD)*, Mar. 2016, pp. 1–6.
- [16] E. Kalantari, H. Yanikomeroglu, and A. Yongacoglu, "On the number and 3D placement of drone base stations in wireless cellular networks," in *Proc. IEEE 84th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2016, pp. 1–6.
- [17] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage," *IEEE Commun. Lett.*, vol. 20, no. 8, pp. 1647–1650, Aug. 2016.
- [18] R. A. Jarvis, "On the identification of the convex hull of a finite set of points in the plane," *Inf. Process. Lett.*, vol. 2, no. 1, pp. 18–21, Mar. 1973.
- [19] D. G. Reina, H. Tawfik, and S. L. Toral, "Multi-subpopulation evolutionary algorithms for coverage deployment of UAV-networks," *Ad Hoc Netw.*, vol. 68, pp. 16–32, Jan. 2018.
- [20] C. Caillouet and T. Razafindralambo, "Efficient deployment of connected unmanned aerial vehicles for optimal target coverage," in *Proc. Global Inf. Infrastruct. Netw. Symp. (GIIS)*, Oct. 2017, pp. 1–8.
- [21] E. N. Almeida, R. Campos, and M. Ricardo, "Traffic-aware multi-tier flying network: Network planning for throughput improvement," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2018, pp. 1–6.
- [22] S.-Y. Park, C. S. Shin, D. Jeong, and H. Lee, "DroneNetX: Network reconstruction through connectivity probing and relay deployment by multiple UAVs in ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 11192–11207, Nov. 2018.
- [23] D.-Y. Kim and J.-W. Lee, "Integrated topology management in flying ad hoc networks: Topology construction and adjustment," *IEEE Access*, vol. 6, pp. 61196–61211, 2018.
- [24] H. R. Hussien, S.-C. Choi, J.-H. Park, and I.-Y. Ahn, "Efficient multi-UAV relay nodes placement scheme in wireless networks," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, Oct. 2021, pp. 893–898.
- [25] A. Bhardwaj and H. El-Ocla, "Multipath routing protocol using genetic algorithm in mobile ad hoc networks," *IEEE Access*, vol. 8, pp. 177534–177548, 2020.
- [26] J. Li and M. Chen, "Multiobjective topology optimization based on mapping matrix and NSGA-II for switched industrial Internet of Things," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1235–1245, Dec. 2016.
- [27] G. Rehani, A. K. Sharma, and K. Verma, "NSGA-II with ENLU inspired clustering for wireless sensor networks," *Wireless Netw.*, vol. 26, no. 5, pp. 3637–3655, Jul. 2020. [Online]. Available: <https://link.springer.com/content/pdf/10.1007/s11276-020-02281-8.pdf> and <https://srmuniversity.ac.in/professors/dr-gunjan-rehani/?dept=computer-science-engineering-cse&pid=318>
- [28] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar. 2000.
- [29] J. S. Baras, V. Tabatabaee, and K. Jain, "A model based platform for design and optimization of multi-hop 802.11 wireless networks," in *Proc. 8th ACM Symp. Perform. Eval. Wireless Ad Hoc, Sensor, Ubiquitous Netw. (PE-WASUN)*. New York, NY, USA: Association for Computing Machinery, 2011, pp. 17–24, doi: 10.1145/2069063.2069067.
- [30] D. Greenhalgh and S. Marshall, "Convergence criteria for genetic algorithms," *SIAM J. Comput.*, vol. 30, no. 1, pp. 269–282, Jan. 2000.
- [31] H. Aytug and G. J. Koehler, "Stopping criteria for finite length genetic algorithms," *INFORMS J. Comput.*, vol. 8, no. 2, pp. 183–191, May 1996.
- [32] M. Garetto, T. Salonidis, and E. W. Knightly, "Modeling per-flow throughput and capturing starvation in CSMA multi-hop wireless networks," *IEEE/ACM Trans. Netw.*, vol. 16, no. 4, pp. 864–877, Aug. 2008.



SÉRGIO E. SABINO received the B.Sc. degree in electronics engineering from Eduardo Mondlane University, Mozambique, in 2008, the M.Sc. degree in informatics engineering from the University of Algarve, Portugal, in 2014, and the Ph.D. degree in electrical and computer engineering from the Instituto Superior Técnico, University of Lisbon, Lisbon, Portugal.

His current research interests include wireless networks and optimization algorithms for network design issues.



ANTÓNIO M. GRILLO (Senior Member, IEEE) received the B.Sc. degree in computer science and engineering, the M.Sc. degree in electrical and computer engineering, and the Ph.D. degree in electrical and computer engineering from the Instituto Superior Técnico, Universidade de Lisboa (IST-UL), Lisbon, Portugal, in 1996, 1998, and 2004, respectively.

Since 1996, he has been working in several European Commission (EC)-funded projects related with communication networks. He is currently an Associate Professor with IST-UL and a Researcher with INESC-ID, Lisbon, where he is also the Coordinator of the Communication Networks Research Area. He is the author or the coauthor of more than eighty scientific articles on subjects related with communication networks. His current research interests include span communication networks for smart utilities, the Internet of Things, and edge computing optimization.

...