

# Probabilistic Roadmaps for Aerial Relay Path Planning

Pham Q. Viet and Daniel Romero

Dept. of Information and Communication Technology, University of Agder, Grimstad, Norway.

Email: {viet.q.pham,daniel.romero}@uia.no.

**Abstract**—Unmanned aerial vehicles (UAVs) with on-board relays can be used to establish multi-hop links that deliver high-speed connectivity beyond cell limits. This is of utmost importance e.g. in remote areas and in emergency scenarios. However, jointly designing the trajectories of multiple such flying relays is a complex task since the dimensionality of the underlying configuration space is too large to allow the direct application of traditional shortest-path methods. To bypass this difficulty, this work proposes a probabilistic roadmap algorithm based on a novel heuristic path design which is guaranteed to provide feasible paths for all UAVs under general conditions. This addresses the limitations of existing algorithms, which are typically based on non-linear optimization and, therefore, entail high complexity and cannot readily accommodate the presence of obstacles such as buildings. As corroborated via numerical experiments in an urban environment, the proposed scheme can establish a high-speed link with a user by means of just two aerial relays in a short time.

**Index Terms**—Flying relays, path planning, probabilistic roadmaps, aerial communications.

## I. INTRODUCTION

The large number of applications of unmanned aerial vehicles (UAV) includes extending the coverage of cellular communication networks beyond cell limits [1]. This is of interest in cases where terrestrial infrastructure is absent, as occurs in remote areas, or damaged, e.g. by a natural disaster or military attack. Depending on the implemented functionality, UAVs that serve this purpose are referred to as *aerial base stations* or as *aerial relays*.

The resulting research problems have attracted great interest in the last few years. Since aerial base stations typically hover at fixed positions, the problem is to determine a suitable set of locations [2]. In contrast, aerial relays typically move to address the needs of the users. A large number of works focus on the case of a single relay; see e.g. [3], [4]. However, long distances and obstructions generally require multiple relays. In this context, a substantial body of literature considers the dissemination or collection of delay-tolerant information, which is not relayed in real time but carried from one place to another by means of the movement of the UAVs; see e.g. [5]. Other works consider utilizing aerial relays to establish real-time communication links. The most common approaches

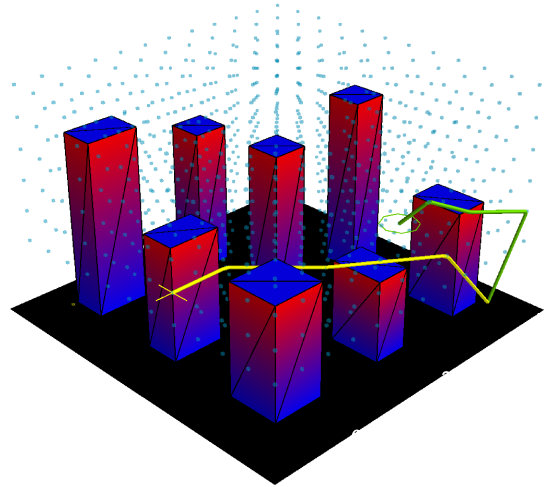


Fig. 1: Trajectories of two relay UAVs obtained with the proposed PRFI algorithm. Red/blue boxes represent buildings. The yellow point on the left represents the UE location. The flight grid points are represented as blue dots. The green line is the trajectory of UAV-1, whose final position is at the green circle. The yellow line represents the trajectory of UAV-2, whose final position is the yellow cross.

involve non-linear optimization formulations over continuous variables, which include the spatial coordinates of all UAVs; see e.g. [6], [7]. Unfortunately, the resulting problems are non-convex, the solvers involve high complexity, and cannot generally accommodate no-flying zones and obstacles such as buildings. Other schemes, such as those based on mixed integer linear programs [8], [9] and the Steiner tree problem [10] also exhibit similar limitations.

Due to obstacles and other spatial constraints, the most common approaches for UAV path planning outside the literature of aerial relays are based on discretizing the airspace and applying a shortest-path algorithm on the resulting graph. This kind of approaches have been applied in the context of UAV communications e.g. to plan a path through coverage areas [11], but not to plan the trajectories of multiple aerial relays. This is because the large number of degrees of freedom renders the complexity of shortest-path algorithms prohibitive.

Inspired by the literature of motion planning for robotic manipulators, this work sidesteps this difficulty by building upon

This work has been funded by the IKTPLUS grant 311994 of the Research Council of Norway.

the celebrated *probabilistic roadmap* (PR) algorithm [12], where a shortest-path primitive is applied on a graph whose nodes are randomly generated *configuration points*. It is worth noting that there have been other works where PR has been applied to UAV path planning (see e.g. [13] and references therein) but never for aerial relays and, to the best of our knowledge, only in 2D or 3D spaces.

The main contribution of this work is a path planning algorithm for two relays that maintain connectivity among themselves and with a terrestrial base station (BS) and navigate through the airspace to establish a communication link with a user in an approximately minimal time. This is illustrated in Fig. 1. The algorithm can accommodate obstacles and no-fly zones, which renders it especially useful in urban scenarios, e.g. for emergency response after an earthquake. It improves upon conventional PR by drawing configuration points that rely on a heuristic waypoint sequence that can be shown to be feasible and meet the target user rate under general conditions. Besides, it is optimal in certain important cases.

After presenting the model and the problem formulation in Sec. II, the proposed algorithm is described in Sec. III. Finally, numerical experiments and conclusions are respectively presented in Secs. IV and V.

## II. MODEL AND PROBLEM FORMULATION

Consider a spatial region  $\mathcal{S} \subset \mathbb{R}^3$  and let  $\mathcal{F} \subset \mathcal{S}$  denote the set of points outside any building or obstacle and above the ground. For simplicity, it is assumed that if  $[x, y, z]^\top \in \mathcal{F}$ , then  $[x, y, z']^\top \in \mathcal{F}$  for all  $z' \geq z$ , which essentially means that the buildings or obstacles contain no holes or parts that stand out. To establish a link between a base station (BS) at location  $\mathbf{q}_{\text{BS}} := [x_{\text{BS}}, y_{\text{BS}}, z_{\text{BS}}]^\top \in \mathcal{S}$  and user equipment (UE) at  $\mathbf{q}_{\text{UE}} := [x_{\text{UE}}, y_{\text{UE}}, z_{\text{UE}}]^\top \in \mathcal{S}$ ,  $K$  aerial relays are deployed. Let  $\bar{\mathcal{F}} \subset \mathcal{F}$  be the set of spatial locations where the UAVs can fly, which is determined by the minimum and maximum allowed altitudes as well as no-fly zones. The position of the  $k$ -th UAV at time  $t$  is represented as  $\mathbf{q}_k(t) \in \bar{\mathcal{F}}$  and the positions of all UAVs at time  $t$  are collected into the  $3 \times K$  matrix  $\mathbf{Q}(t) := [\mathbf{q}_1(t), \dots, \mathbf{q}_K(t)]$ , referred to as the *configuration point* (CP) at time  $t$ . The set of all matrices  $\mathbf{Q}(t)$  whose columns are in  $\bar{\mathcal{F}}$  is the so-called *configuration space* (Q-space) and will be denoted as  $\mathcal{Q}$ . The UAVs collectively follow a trajectory  $\mathcal{T} := \{\mathbf{Q}(t), t \geq 0\}$ . For simplicity, they all take off at the BS, i.e.  $\mathbf{q}_k(0) = \mathbf{q}_{\text{BS}} \forall k$ .

The targeted link must convey information in both ways. In the downlink, UAV-1 relays the signal from the BS to UAV-2, UAV-2 relays the signal from UAV-1 to UAV-3, and so on until UAV- $K$  relays the signal from UAV- $(K-1)$  to the UE. To simplify the exposition, the focus here will be on the downlink. Relays employ the decode-and-forward strategy with capacity-attaining codes and the interference between hops is ignored.

Besides the data to be relayed towards the UE, each UAV consumes a rate of  $r_{\text{CC}}$  for command and control. This means that the achievable rate between the BS and UAV- $k$  for a generic CP  $\mathbf{Q} := [\mathbf{q}_1, \dots, \mathbf{q}_K]$  can be recursively obtained as

$$r_k(\mathbf{Q}) = \max(0, \min(r_{k-1}(\mathbf{Q}) - r_{\text{CC}}, c(\mathbf{q}_{k-1}, \mathbf{q}_k))), \quad (1)$$

where  $c(\mathbf{q}, \mathbf{q}')$  denotes the channel capacity between a UAV at  $\mathbf{q}$  and a UAV at  $\mathbf{q}'$  and  $r_1(\mathbf{Q}) = c(\mathbf{q}_{\text{BS}}, \mathbf{q}_1)$ . The achievable rate of the UE is

$$r_{\text{UE}}(\mathbf{Q}) = \max(0, \min(r_K(\mathbf{Q}) - r_{\text{CC}}, c(\mathbf{q}_K, \mathbf{q}_{\text{UE}}))). \quad (2)$$

Throughout the trajectory, the UAVs must have connectivity with the BS, which means that  $r_k(\mathbf{Q}(t)) \geq r_{\text{CC}} \forall k, t$ . Meanwhile, the UE is said to have connectivity with the BS if  $r_{\text{UE}}(\mathbf{Q})$  exceeds a given rate  $r_{\text{UE}}^{\min}$ .

To plan the trajectory, it is necessary to know function  $c(\mathbf{q}, \mathbf{q}')$ . In practice, one needs to resort to some approximation. For example, one can use a channel-gain map [14], a 3D terrain/city model with a ray-tracing algorithm, a line-of-sight (LOS) map, a set of bounding boxes known to contain the buildings and other obstacles, specific models for UAV channels, etc.

The problem addressed in this paper is to design a trajectory such that connectivity is established between the UE and the BS as soon as possible. Formally, given  $\mathbf{q}_{\text{BS}}, \mathbf{q}_{\text{UE}}, K, c(\mathbf{q}, \mathbf{q}'), r_{\text{CC}}, r_{\text{UE}}^{\min}$ , and the maximum UAV fly speed  $v_{\text{max}}$ , the problem is to solve

$$\underset{\mathcal{T}}{\text{minimize}} \quad T_c(\mathcal{T}) \quad (3a)$$

$$\text{s.t.} \quad \mathbf{Q}(t) \in \mathcal{Q} \forall t \quad (3b)$$

$$\mathbf{Q}(0) = [\mathbf{q}_{\text{BS}}, \dots, \mathbf{q}_{\text{BS}}] \quad (3c)$$

$$r_k(\mathbf{Q}(t)) \geq r_{\text{CC}} \forall k, t \quad (3d)$$

$$\|\dot{\mathbf{q}}_k(t)\| \leq v_{\text{max}} \forall k, t, \quad (3e)$$

where  $T_c(\mathcal{T}) := \inf\{t : r_{\text{UE}}(\mathbf{Q}(t)) \geq r_{\text{UE}}^{\min}\}$  is the *connection time* and  $\dot{\mathbf{q}}_k(t)$  is the velocity vector of UAV- $k$ . Note that, consistent with the standard convention for the infimum,  $T_c(\mathcal{T}) = \infty$  if  $r_{\text{UE}}(\mathbf{Q}(t)) \geq r_{\text{UE}}^{\min}$  does not hold for any  $t$ . A trajectory  $\mathcal{T}$  is said to be *valid* if it is feasible and  $T_c(\mathcal{T}) < \infty$ . Note that a minimum distance between UAVs is not explicitly enforced but it can be readily accommodated in the proposed scheme.

## III. PATH PLANNING VIA PROBABILISTIC ROADMAPS

Since the optimization variable in Problem (3) is a function  $\mathbf{Q} : \mathbb{R}_+ \rightarrow \mathbb{R}^{3 \times K}$ , finding the exact solution to (3) will not be generally possible by numerical means. For this reason, the goal in this paper will be finding a valid trajectory with a reasonably short connection time.

More specifically, the focus will be on the case  $K = 2$  since, as shown next, this suffices to guarantee the existence of a valid trajectory under general conditions. For theoretical purposes, consider a simple channel model where the capacity  $c(\mathbf{q}, \mathbf{q}')$  is a decreasing function  $c(d)$  of the distance  $d = \|\mathbf{q} - \mathbf{q}'\|$  whenever there is LOS between  $\mathbf{q}$  and  $\mathbf{q}'$ .

*Proposition 1:* Let  $h$  denote the height of the highest obstacle and suppose that the UAVs can fly above  $h$ . Furthermore, let  $d := \sqrt{(x_{\text{UE}} - x_{\text{BS}})^2 + (y_{\text{UE}} - y_{\text{BS}})^2}$  denote the horizontal distance between the BS and the UE. If  $h < \min(z_{\text{UE}} + c^{-1}(r_{\text{UE}}^{\min}), z_{\text{BS}} + c^{-1}(r_{\text{UE}}^{\min} + 2r_{\text{CC}}))$  and  $d \leq c^{-1}(r_{\text{UE}}^{\min} + r_{\text{CC}})$ , then there exists a valid trajectory  $\mathcal{T}$  for Problem (3).

*Proof:* Let  $z = \min(z_{\text{UE}} + c^{-1}(r_{\text{UE}}^{\min}), z_{\text{BS}} + c^{-1}(r_{\text{UE}}^{\min} + 2r_{\text{CC}}))$ . It is easy to show that if UAV-1 navigates to  $\mathbf{q}_1 := [x_{\text{BS}}, y_{\text{BS}}, z]^\top$  and UAV-2 navigates first to  $\mathbf{q}_1$  and then to  $\mathbf{q}_2 := [x_{\text{UE}}, y_{\text{UE}}, z]^\top$ , then there is LOS at all hops,  $\mathcal{T}$  is feasible, and  $T_c(\mathcal{T}) < (z - z_{\text{BS}} + d)/v_{\text{max}}$ . ■

Thus, so long as  $d$  and  $h$  are not too large relative to  $r_{\text{UE}}^{\min}$  and  $r_{\text{CC}}$ , a valid trajectory exists with only 2 relays.

Following standard practice in UAV path planning, both spatial and temporal discretizations will be introduced. Specifically, the flight region  $\bar{\mathcal{F}}$  is discretized into a regular 3D *flight grid*  $\bar{\mathcal{F}}_G \subset \bar{\mathcal{F}} \subset \mathbb{R}^3$ , whose points are separated along the  $x$ ,  $y$ , and  $z$  axes by  $\delta_x$ ,  $\delta_y$ , and  $\delta_z$  respectively; see Fig. 1. This spatial discretization also induces a discretization of the  $Q$ -space, which for  $K = 2$  results in a grid  $\mathcal{Q}_G := \bar{\mathcal{F}}_G \times \bar{\mathcal{F}}_G$ . Regarding the time-domain discretization, the trajectory  $\mathcal{T}$  will be designed by first finding a waypoint sequence  $\mathcal{T}_W := \{\mathbf{Q}[0], \dots, \mathbf{Q}[N-1]\}$  through the grid  $\mathcal{Q}_G$ .

Conventional algorithms for planning the path of a single UAV create a graph whose nodes are the points of  $\bar{\mathcal{F}}_G$  and where an edge exists between two nodes if the associated points are adjacent on the grid. In a 3D grid like  $\bar{\mathcal{F}}_G$ , each point may have at most 26 adjacent points, which makes the application of shortest-path algorithms viable. However, the grid  $\mathcal{Q}_G$  has many more points than  $\bar{\mathcal{F}}_G$ . For example, if  $\bar{\mathcal{F}}_G$  is a (small)  $10 \times 10 \times 10$  grid, then  $\mathcal{Q}_G$  has  $10^6$  points. Besides, since each point has  $27^2 - 1 = 728$  adjacent points, solving Problem (3) via shortest-path algorithms is challenging.

To bypass this kind of difficulties, the seminal paper [12] proposed the PR algorithm, which consists of 3 steps: Step 1: a node set  $\mathcal{N} \subset \mathcal{Q}$  with a much smaller number of nodes than  $\mathcal{Q}_G$  is generated at random. Step 2: the edge set  $\mathcal{E} \subset \mathcal{N} \times \mathcal{N}$  is constructed by connecting each point in  $\mathcal{N}$  with its nearest neighbors. Step 3: a shortest-path algorithm is found on this graph.

Unfortunately, the complexity reduction of PR comes at a cost: the number of nodes in  $\mathcal{N}$  necessary to find a feasible path may be very large. The key idea in the present paper is to apply PR to solve Problem (3) but a modification is introduced to avoid the aforementioned limitation: the node generation (sampling) in Step 1 is carried out in such a way that Step 3 can always find a valid waypoint sequence. This technique is described in Sec. III-B and is based on a heuristic path planning technique presented in Sec. III-A.

### A. Valid Path Design

This section proposes a heuristic approach that is guaranteed to find a valid path under general conditions. In many situations, this path will be nearly optimal and will typically result in a connection time much shorter than the one of the path in Proposition 1; cf. Sec. IV. The idea is to first generate the path for UAV-2. Then, a path is found for UAV-1 to serve UAV-2 throughout. If this is not possible, the path of UAV-2 is *lifted* until it becomes possible.

Upon introducing the notation  $\mathcal{R}(\mathbf{q}, r) := \{\mathbf{q}' \in \bar{\mathcal{F}}_G : c(\mathbf{q}, \mathbf{q}') \geq r\}$ , it is clearly necessary (but not sufficient) that UAV-1 is in  $\mathcal{R}(\mathbf{q}_{\text{BS}}, 2r_{\text{CC}})$  throughout the path and in

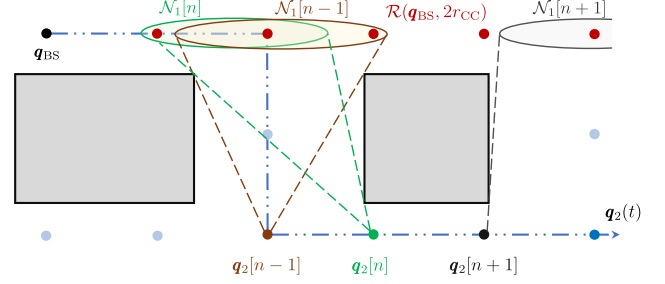


Fig. 2: Top view of an example case where no path through adjacent points exists that allows UAV-1 to serve UAV-2 throughout the tentative path of the latter. At some point, UAV-2 may need to wait so that UAV-1 can gain altitude to maintain service to UAV-2. Grey boxes represent buildings and dots are grid points.

$\mathcal{R}(\mathbf{q}_{\text{BS}}, 2r_{\text{CC}} + r_{\text{UE}}^{\min})$  at the moment of establishing connectivity with the UE. Similarly, with notation  $\mathcal{R}(\mathbf{q}, r, r') := \{\mathbf{q}'' \in \bar{\mathcal{F}}_G \mid \exists \mathbf{q}' \in \mathcal{R}(\mathbf{q}, r) : c(\mathbf{q}', \mathbf{q}'') \geq r'\}$ , it is necessary (not sufficient) that UAV-2 is in  $\mathcal{N}_2 := \mathcal{R}(\mathbf{q}_{\text{BS}}, 2r_{\text{CC}}, r_{\text{CC}})$  throughout the path and in  $\mathcal{D}_2 := \mathcal{R}(\mathbf{q}_{\text{BS}}, 2r_{\text{CC}} + r_{\text{UE}}^{\min}, r_{\text{CC}} + r_{\text{UE}}^{\min}) \cap \mathcal{R}(\mathbf{q}_{\text{UE}}, r_{\text{UE}}^{\min})$  at the moment of establishing connectivity with the UE.

**Tentative Path for UAV-2.** The idea is to start by first planning the path of UAV-2 by finding the shortest path (e.g. via Dijkstra's algorithm) from  $\mathbf{q}_{\text{BS}}$  through a graph with node set  $\mathcal{N}_2$  to the nearest point in  $\mathcal{D}_2 \subset \mathcal{N}_2$ . In this graph, two nodes  $\mathbf{q}$  and  $\mathbf{q}'$  are connected if they are adjacent in  $\bar{\mathcal{F}}_G$ , which will be denoted as  $(\mathbf{q}, \mathbf{q}') \in \mathcal{E}_{\bar{\mathcal{F}}_G}$ , and the weight of an edge  $(\mathbf{q}, \mathbf{q}')$  is the distance  $\|\mathbf{q} - \mathbf{q}'\|$  since it is proportional to the time it takes for UAV-2 to travel from  $\mathbf{q}$  to  $\mathbf{q}'$  and the objective in (3) is the connection time. This procedure produces a waypoint sequence  $\mathbf{q}_2[0], \mathbf{q}_2[1], \dots, \mathbf{q}_2[N-1]$ .

**Path for UAV-1.** If a feasible waypoint sequence exists for UAV-1 that provides a sufficient rate to UAV-2 at all the waypoints  $\mathbf{q}_2[0], \mathbf{q}_2[1], \dots, \mathbf{q}_2[N-1]$ , the combined path will not only be valid but also optimal up to the effects of the discretization. However, such a path for UAV-1 does not always exist, as seen later.

Clearly, for the combined path to be feasible, the position of UAV-1 must satisfy  $\mathbf{q}_1 \in \mathcal{N}_2[n] := \mathcal{R}(\mathbf{q}_{\text{BS}}, 2r_{\text{CC}}) \cap \mathcal{R}(\mathbf{q}_2[n], r_{\text{CC}})$  when UAV-2 is at  $\mathbf{q}_2[n]$ . Besides, for the path to be valid, it is required that  $\mathbf{q}_1 \in \mathcal{D}_1 := \mathcal{R}(\mathbf{q}_{\text{BS}}, 2r_{\text{CC}} + r_{\text{UE}}^{\min}) \cap \mathcal{R}(\mathbf{q}_2[N-1], r_{\text{CC}} + r_{\text{UE}}^{\min}) \subset \mathcal{N}_2[N-1]$  when UAV-2 reaches  $\mathbf{q}_2[N-1]$ . Since the set of positions where UAV-1 must be depends on  $n$ , one needs to plan a path through an *extended graph*. Upon letting  $\bar{\mathcal{N}}_1[n] := \{(n, \mathbf{q}) \mid \mathbf{q} \in \mathcal{N}_1[n]\}$ , the node set of this graph is  $\bar{\mathcal{N}}_1 := \cup_n \bar{\mathcal{N}}_1[n]$ .

As a first approach to plan the path of UAV-1, one can think of finding a path  $(0, \mathbf{q}_1[0]), (1, \mathbf{q}_1[1]), \dots, (N-1, \mathbf{q}_1[N-1])$  where  $(n, \mathbf{q}_1[n]) \in \bar{\mathcal{N}}_1[n] \forall n, \mathbf{q}_1[N-1] \in \mathcal{D}_1$ , and  $(\mathbf{q}_1[n], \mathbf{q}_1[n+1]) \in \mathcal{E}_{\bar{\mathcal{F}}_G} \forall n$ . If this is possible, then the sequence of waypoints  $\mathbf{Q}[n] = [\mathbf{q}_1[n], \mathbf{q}_2[n]]$ ,  $n = 0, \dots, N-1$  is, as indicated earlier, optimal. This is the case when UAV-

1 can maintain connectivity to UAV-2 just by moving to adjacent locations on  $\bar{\mathcal{F}}_G$ . However, this may not be the case: sometimes UAV-1 may need to perform multiple steps through adjacent locations on  $\bar{\mathcal{F}}_G$  to fly around obstacles in order to guarantee connectivity to UAV-2; see Fig. 2. In other words, UAV-2 needs to wait at a certain waypoint until UAV-1 adopts a suitable location. Formally, this means that the path for UAV-1 is of the form  $(n_0, \mathbf{q}_1[0]), (n_1, \mathbf{q}_1[1]), \dots, (n_{\tilde{N}-1}, \mathbf{q}_1[\tilde{N}-1])$  for some  $\tilde{N}$ , where  $(n_i, \mathbf{q}_1[i]) \in \tilde{\mathcal{N}}_1[n_i]$ ,  $n_0 = 0$ ,  $\mathbf{q}_1[\tilde{N}-1] \in \mathcal{D}_1$ ,  $n_{i-1} \leq n_i \leq n_{i-1} + 1$ , and  $(\mathbf{q}_1[i], \mathbf{q}_1[i+1]) \in \mathcal{E}_{\bar{\mathcal{F}}_G}$  for all  $i$ . The corresponding sequence of waypoints for UAV-2 will be  $\mathbf{q}_2[n_0], \mathbf{q}_2[n_1], \dots, \mathbf{q}_2[n_{\tilde{N}-1}]$ , that is, UAV-2 waits at  $\mathbf{q}_2[n_i]$  whenever  $n_i = n_{i+1}$ .

The weights of the edges in the extended (directed) graph can be selected as

$$w((n, \mathbf{q}), (n', \mathbf{q}')) = \begin{cases} \|\mathbf{q} - \mathbf{q}'\| & \text{if } (\mathbf{q}, \mathbf{q}') \in \mathcal{E}_{\bar{\mathcal{F}}_G} \\ & \text{and } n \leq n' \leq n + 1, \\ \infty & \text{otherwise,} \end{cases} \quad (4)$$

where an infinite weight indicates that the corresponding nodes are not connected.

In certain cases, it may happen that no path can be found in the extended graph. To remedy this, one can *lift* the tentative path of UAV-2. To this end, let  $h_{\max}$  be the height of the lowest grid level that is higher than all buildings. The path  $\mathbf{q}_2[0], \mathbf{q}_2[1], \dots, \mathbf{q}_2[N-1]$  can be first replaced with  $L(\mathbf{q}_2[0]), L(\mathbf{q}_2[1]), \dots, L(\mathbf{q}_2[N-1])$ , where  $L(\mathbf{q}) = \mathbf{q} + [0, 0, \delta_z]^T$  if  $[0, 0, 1]\mathbf{q} + \delta_z \leq h_{\max}$  and  $L(\mathbf{q}) = \mathbf{q}$  otherwise. Next, the starting location is appended at the beginning if  $\mathbf{q}_2[0] \neq L(\mathbf{q}_2[0])$ . Similarly, the end location is appended at the end if  $L(\mathbf{q}_2[N-1]) \neq \mathbf{q}_2[N-1]$ . This yields the lifted path  $\mathbf{q}_2[0], L(\mathbf{q}_2[0]), L(\mathbf{q}_2[1]), \dots, L(\mathbf{q}_2[N-1]), \mathbf{q}_2[N-1]$ .

After lifting the path of UAV-2, it is more likely that a feasible path exists for UAV-1. This procedure may need to be repeated multiple times but it will eventually succeed:

*Proposition 2:* Let  $\mathcal{S} = [0, x_{\max}] \times [0, y_{\max}] \times [0, z_{\max}]$  be the region of interest, where a sufficiently dense regular grid  $\bar{\mathcal{F}}_G$  is constructed. With the channel model introduced at the beginning of Sec. III, suppose that  $\sqrt{x_{\max}^2 + y_{\max}^2} \leq c^{-1}(r_{CC})$  and that  $h_{\max} \leq \sqrt{[c^{-1}(2r_{CC})]^2 - [c^{-1}(2r_{CC} + r_{UE}^{\min})]^2}$ . If a valid path for (3) exists through waypoints in  $\mathcal{Q}_G$ , then the procedure described in this section yields a valid waypoint sequence  $\mathcal{T}_W^V := \{\{\mathbf{q}_1[i], \mathbf{q}_2[n_i]\} \mid i = 0, \dots, \tilde{N} - 1\}$ .

*Proof:* The proof is omitted due to space limitations. ■

Thus, if  $r_{UE}^{\min}$  and  $r_{CC}$  are not too large relative to the size of the region, the approach in this section results in a valid path whenever such a path exists. If this condition does not hold, one can design situations where a valid path exists but the algorithm cannot find it. However, this was never observed in our experiments.

### B. Q-space Sampling for Probabilistic Roadmaps

As described earlier, the first step in PR is to randomly generate a set  $\mathcal{N}$  of CPs. The sampled distribution drastically

impacts the optimality and computational burden of the algorithm. In the work at hand,  $\mathcal{N}$  will comprise all the CPs of the path  $\mathcal{T}_W^V = \{\mathbf{Q}[0], \dots, \mathbf{Q}[\tilde{N} - 1]\}$  from Sec. III-A together with  $C$  additional CPs drawn at random around these CPs and likely to be near the optimal path. Specifically, for each  $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2] \in \mathcal{T}_W^V$ , the proposed sampling strategy generates  $\lfloor C/\tilde{N} \rfloor$  configuration points  $\tilde{\mathbf{Q}} = [\tilde{\mathbf{q}}_1, \tilde{\mathbf{q}}_2]$  as follows.

Suppose that both  $\mathcal{R}(\mathbf{q}_{BS}, 2r_{CC})$  and  $\mathcal{R}(\mathbf{q}_{BS}, 2r_{CC}, r_{CC})$  contain more than one element each. Generate  $\tilde{\mathbf{q}}_1$  by drawing a point of  $\mathcal{R}(\mathbf{q}_{BS}, 2r_{CC}) - \{\mathbf{q}_1\}$  with probability proportional to  $1/\|\tilde{\mathbf{q}}_1 - \mathbf{q}_1\|$ . Independently of  $\tilde{\mathbf{q}}_1$ , generate  $\tilde{\mathbf{q}}_2$  by drawing a point of  $\mathcal{R}(\mathbf{q}_{BS}, 2r_{CC}, r_{CC}) - \{\mathbf{q}_2\}$  with probability proportional to  $1/\|\tilde{\mathbf{q}}_2 - \mathbf{q}_2\|$ . If  $\tilde{\mathbf{q}}_2 \notin \mathcal{R}(\tilde{\mathbf{q}}_1, r_{CC})$ , another pair  $(\tilde{\mathbf{q}}_1, \tilde{\mathbf{q}}_2)$  is generated until  $\tilde{\mathbf{q}}_2 \in \mathcal{R}(\tilde{\mathbf{q}}_1, r_{CC})$ , which will eventually happen as  $\tilde{\mathbf{q}}_2 \in \mathcal{R}(\mathbf{q}_{BS}, 2r_{CC}, r_{CC})$ . In case that  $\mathcal{R}(\mathbf{q}_{BS}, 2r_{CC})$  and/or  $\mathcal{R}(\mathbf{q}_{BS}, 2r_{CC}, r_{CC})$  do not contain more than one element,  $\tilde{\mathbf{q}}_1$  and/or  $\tilde{\mathbf{q}}_2$  are taken to be the sole element in the corresponding set.

This procedure, which is based on the distance to CPs in  $\mathcal{T}_W^V$ , ensures that many of the sampled CPs lie close to the path while others may be farther away, thereby increasing the chances for finding near-optimal trajectories in the next step.

### C. Path Planning in Q-Space

Given  $\mathcal{N}$  as generated in Sec. III-B, PR constructs a nearest-neighbor graph and seeks a shortest path through it. To this end, it is necessary to define a distance (metric) between configuration points. Due to the objective in (3), the metric between  $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2]$  and  $\mathbf{Q}' = [\mathbf{q}'_1, \mathbf{q}'_2]$  should be the time that the UAVs require to move from  $\mathbf{Q}$  to  $\mathbf{Q}'$  provided that they can perform this movement without losing connectivity or abandoning  $\bar{\mathcal{F}}$  and  $\infty$  otherwise. In the former case, given the speed constraint (3e), the time is  $\max_k \|\mathbf{q}_k - \mathbf{q}'_k\|/v_{\max}$ , as determined by the UAV that traverses the longest distance. To check numerically whether it is possible to move from  $\mathbf{Q}$  to  $\mathbf{Q}'$  in a straight line, one can check whether the rate constraints are satisfied for a certain set of points between  $\mathbf{Q}$  and  $\mathbf{Q}'$ .

After the nearest-neighbor graph is constructed, the shortest path is sought from the starting location  $[\mathbf{q}_{BS}, \mathbf{q}_{BS}]$  to any of the CPs  $[\mathbf{q}_1, \mathbf{q}_2]$  that satisfies  $\mathbf{q}_1 \in \mathcal{R}(\mathbf{q}_{BS}, 2r_{CC} + r_{UE}^{\min})$ ,  $\mathbf{q}_2 \in \mathcal{R}(\mathbf{q}_1, r_{CC} + r_{UE}^{\min})$ , and  $\mathbf{q}_2 \in \mathcal{R}(\mathbf{q}_{UE}, r_{UE}^{\min})$ . The waypoint sequence  $\mathcal{T}_W^{\text{PR}}$  that results from application of the shortest-path algorithm will never have a greater cost than the feasible path  $\mathcal{T}_W^V$  obtained in Sec. III-A provided that all consecutive CPs in  $\mathcal{T}_W^V$  are connected in the PR graph, which holds so long as  $\bar{\mathcal{F}}_G$  is sufficiently dense.

### D. From the Waypoint Sequence to the Trajectory

Given the waypoint sequence  $\mathcal{T}_W^{\text{PR}} = \{\mathbf{Q}[0], \dots, \mathbf{Q}[N^{\text{PR}} - 1]\}$ , it remains only to obtain the trajectory  $\mathcal{T}^{\text{PR}} = \{\mathbf{Q}(t) \mid t \geq 0\}$ . To this end, the first step is to determine the time at which the UAVs arrive at each of the waypoints  $\mathbf{Q}[n]$ . As indicated in Sec. III-C, the time it takes to arrive at  $\mathbf{Q}[n] = [\mathbf{q}_1[n], \mathbf{q}_2[n]]$  from  $\mathbf{Q}[n-1] = [\mathbf{q}_1[n-1], \mathbf{q}_2[n-1]]$  is  $\max_k \|\mathbf{q}_k[n] - \mathbf{q}_k[n-1]\|/v_{\max}$ . Let  $t_n$  represent the time at which the UAVs arrive

---

**Algorithm 1:** PR with Feasible Initialization (PRFI)

---

- 1: Input:  $\bar{\mathcal{F}}_G, \mathbf{q}_{BS}, \mathbf{q}_{UE}, r_{CC}, r_{UE}^{\min}, c, C$ .
  - 2: Obtain a tentative path for UAV-2.
  - 3: While True
    - Find path for UAV-1 to serve UAV-2.
    - If path found, break.
    - Lift the path of UAV-2.
  - 4: For each  $\mathbf{Q} \in \mathcal{T}_W^V$ , draw  $[C/\tilde{N}]$  CPs  $\rightarrow \mathcal{N}$ .
  - 5: Construct a nearest-neighbor graph.
  - 6: Find the shortest path through  $\mathcal{N}$ .
  - 7: Obtain  $\mathcal{T}^{\text{PR}} = \{\mathbf{Q}(t) \mid t \geq 0\}$  from  $\mathcal{T}_W^{\text{PR}}$ .
  - 8: **return**  $\mathcal{T}^{\text{PR}}$
- 

at  $\mathbf{Q}[n]$  and let  $t_0 = 0$ . In this case, it clearly holds that

$$t_n = t_{n-1} + \frac{\max_k \|\mathbf{q}_k[n] - \mathbf{q}_k[n-1]\|}{v_{\max}}. \quad (5)$$

This provides  $\mathbf{Q}(t_n)$  for  $n = 0, \dots, N^{\text{PR}} - 1$ . For  $t \geq t_{N^{\text{PR}}-1}$ , one can simply set  $\mathbf{Q}(t) = \mathbf{Q}(t_{N^{\text{PR}}-1})$ . To obtain  $\mathbf{Q}(t)$  for other values of  $t$ , one can simply apply an interpolation algorithm. In the experiments of Sec. IV, linear interpolation is used since it ensures that the maximum speed is never exceeded. Although the derivatives of the resulting  $\mathbf{Q}(t)$  will be generally discontinuous at the waypoints, this is not a problem in practice since a flight controller typically requires just a sequence of waypoints, which can be readily found by resampling the interpolated  $\mathbf{Q}(t)$  at uniform intervals.

The complete algorithm is summarized in Algorithm 1 and will be referred to as *PR with feasible initialization* (PRFI).

#### IV. NUMERICAL EXPERIMENTS

This section presents numerical results that validate the efficacy and assess the performance of the proposed PRFI algorithm. The simulator and videos of some trajectories can be found at [https://github.com/uiano/pr\\_for\\_relay\\_path\\_planning](https://github.com/uiano/pr_for_relay_path_planning).

The simulation scenario consists of an urban environment of dimensions  $100 \times 100$  m like the one in Fig. 1. For reproducibility purposes, all buildings are set to have a height of 40 m. The minimum and maximum fly heights are respectively 10 and 70 m. The maximum UAV speed is 2.5 m/s. The carrier frequency and bandwidth of the transmitted signals are respectively 6 GHz and 20 MHz. The transmitted power of the BS and the UAVs is 17 dBm. An antenna gain of 12 dB is used at both the transmit and receive sides to simulate the beamforming gain of an antenna array. The noise power is -97 dBm. To be able to operate, each UAV needs a command-and-control rate of  $r_{CC} = 1$  kbps. Function  $c$  is obtained using the tomographic channel model [15], [16] with an absorption of 1 dB/m inside the buildings and 0 dB/m outside the buildings. This is more favorable than a channel that requires LOS and is chosen this way to favor the benchmarks; the proposed algorithm will assume that LOS is required, which is more restrictive.

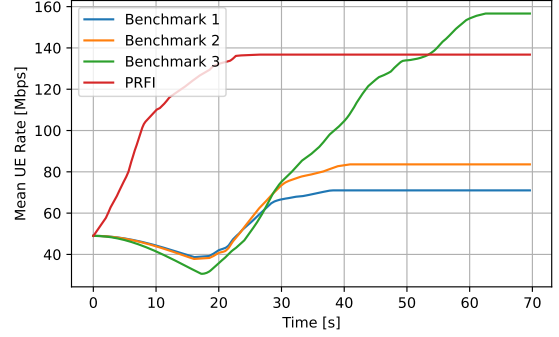


Fig. 3: Expected UE rate  $E[r_{UE}(\mathbf{Q}(t))]$  vs. time averaged across 50 MC realizations ( $r_{UE}^{\min} = 100$  Mbps).

Due to the presence of buildings, no algorithm in the literature that we are aware of can directly accommodate the considered simulation setup. Instead, three benchmarks will be considered: Benchmark 1 is a single UAV that takes off at  $\mathbf{q}_{BS}$  vertically to a height of 41 m and then moves horizontally in straight line to the middle point between the BS and UE, i.e., to the point  $([x_{BS}, y_{BS}, 41]^T + [x_{UE}, y_{UE}, 41]^T)/2$ . In Benchmark 2, two UAVs lift off at the BS to a height of 41 m above it. Then UAV-1 moves to  $(2/3)[x_{BS}, y_{BS}, 41]^T + (1/3)[x_{UE}, y_{UE}, 41]^T$  whereas UAV-2 moves to  $(1/3)[x_{BS}, y_{BS}, 41]^T + (2/3)[x_{UE}, y_{UE}, 41]^T$ . Benchmark 3 is similar to Benchmark 2, but UAV-1 remains at  $[x_{BS}, y_{BS}, 41]^T$  after lift off whereas UAV-2 flies to  $[x_{UE}, y_{UE}, 41]^T$ . Under the hypotheses of Proposition 1, the resulting path is always feasible and yields a finite cost. All three benchmarks stop at the point of their trajectories where the UE rate is maximum. The proposed PRFI algorithm is implemented with a  $12 \times 12 \times 8$  grid  $\bar{\mathcal{F}}_G$ . The number of CPs is 500 and the number of nearest neighbors is limited to 50.

In all experiments, the relevant performance metric is averaged over 50 Monte Carlo (MC) realizations in which  $\mathbf{q}_{BS}$  and  $\mathbf{q}_{UE}$  are randomly generated on the ground and inside  $\bar{\mathcal{F}}$ .

Fig. 3 plots the MC estimate of the expectation  $E[r_{UE}(\mathbf{Q}(t))]$  vs.  $t$  for all compared algorithms when  $r_{UE}^{\min} = 100$  Mbps. Since all UAVs start from the BS, the initial rate is the same for all algorithms. This rate is not 0 because in some MC realizations the BS and the UE may already be close to each other and possibly in LOS. Observe that the proposed algorithm attains the target rate  $r_{UE}^{\min}$  in just a few seconds. The rate continues increasing beyond  $r_{UE}^{\min}$  due to two effects: first, due to the spatial discretization, there are no grid points  $\mathbf{Q}$  that exactly result in  $r_{UE}(\mathbf{Q}) = r_{UE}^{\min}$ . Since the algorithm needs to find destinations with  $r_{UE}(\mathbf{Q}) \geq r_{UE}^{\min}$ , this will generally result in destinations that exceed this minimum. Second, in those MC realizations where UAV-2 needs to turn a corner to reach the UE, the rate increases suddenly. This can be seen by plotting  $r_{UE}(\mathbf{Q}(t))$  for individual MC realizations. Note also that, in accordance with Proposition 1, Benchmark 3 eventually attains  $r_{UE}^{\min}$ . Finally, it is also observed that the rate of the benchmarks initially decreases. This is due to the

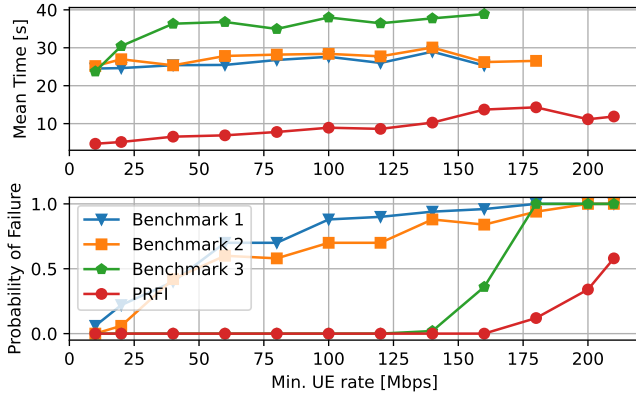


Fig. 4: Mean connection time and probability of failure vs. the UE rate requirement  $r_{UE}^{\min}$  averaged across 50 MC realizations.

take-off in those realizations where there are good propagation conditions directly between the BS and UE locations.

The second experiment studies the influence of  $r_{UE}^{\min}$  onto the expectation of the connection time, which is the cost in Problem (3). To this end, Fig. 4 plots  $E[T_c(\mathcal{T})|T_c(\mathcal{T}) < \infty]$  as well as the probability of failure vs.  $r_{UE}^{\min}$ . The notation  $E[T_c(\mathcal{T})|T_c(\mathcal{T}) < \infty]$  indicates that only those MC realizations where the  $r_{UE}^{\min}$  is attained are considered in the MC average. The probability of failure is defined as the fraction of MC realizations for which  $r_{UE}(\mathbf{Q}(t)) < r_{UE}^{\min} \forall t$ .

The proposed algorithm showcases a 0 probability of failure until  $r_{UE}^{\min} \approx 160$  Mbps and, throughout this interval, its expected connection time is much smaller than for the benchmarks. Benchmark 3 also provides a null probability of failure until  $r_{UE}^{\min}$  is too large, in which case the hypotheses of Proposition 1 no longer hold.

Finally, Fig. 5 plots  $E[T_c(\mathcal{T})|T_c(\mathcal{T}) < \infty]$  and the probability of failure vs. the distance  $\|\mathbf{q}_{UE} - \mathbf{q}_{BS}\|$ . 50 MC realizations are generated for each considered value of  $\|\mathbf{q}_{UE} - \mathbf{q}_{BS}\|$  by first drawing  $\mathbf{q}_{BS}$  uniformly at random out of the buildings and then drawing  $\mathbf{q}_{UE}$  on a circle of the appropriate radius and centered at  $\mathbf{q}_{BS}$ . Consistent with Fig. 4, PRFI yields a zero probability of failure throughout. At this  $r_{UE}^{\min}$ , Benchmark 1 yields a zero connection time but this is because it only succeeds to deliver this rate in the MC realizations where the BS and UE have already good direct communication conditions.

## V. CONCLUSIONS

This paper proposed an algorithm to plan the paths of two aerial relays to establish a link of a given rate with a UE in an approximately minimal time. To cope with the complexity of jointly planning both paths, the PR algorithm was applied with a modification that ensures under general conditions that a valid trajectory can be found if it exists. This circumvents one of the main limitations of plain PR, which is the need for a large number of CPs to find a feasible path. Future work will develop algorithms with more than two flying relays and will investigate alternative sampling strategies.

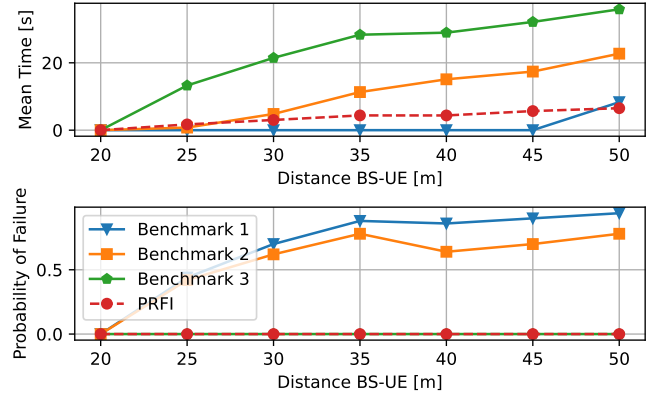


Fig. 5: Mean connection time and probability of failure vs. the distance  $\|\mathbf{q}_{UE} - \mathbf{q}_{BS}\|$  averaged across 50 MC realizations ( $r_{UE}^{\min} = 160$  Mbps).

## REFERENCES

- [1] P. Q. Viet and D. Romero, "Aerial base station placement: A tutorial introduction," *IEEE Commun. Magazine.*, vol. 60, no. 5, May 2022.
- [2] D. Romero, P. Q. Viet, and R. Shrestha, "Aerial base station placement via propagation radio maps," *arXiv preprint arXiv:2301.04966*, 2023.
- [3] H. Wang, G. Ren, J. Chen, G. Ding, and Y. Yang, "Unmanned aerial vehicle-aided communications: Joint transmit power and trajectory optimization," *IEEE Wireless Commun. Lett.*, vol. 7, no. 4, pp. 522–525, 2018.
- [4] K. Anazawa, P. Li, T. Miyazaki, and S. Guo, "Trajectory and data planning for mobile relay to enable efficient internet access after disasters," in *IEEE Global Commun. Conf.*, 2015, pp. 1–6.
- [5] H. Asano, H. Okada, C. B. Naila, and M. Katayama, "Flight model using Voronoi tessellation for a delay-tolerant wireless relay network using drones," *IEEE Access*, vol. 9, pp. 13064–13075, 2021.
- [6] G. Zhang, H. Yan, Y. Zeng, M. Cui, and Y. Liu, "Trajectory optimization and power allocation for multi-hop UAV relaying communications," *IEEE Access*, vol. 6, pp. 48566–48576, 2018.
- [7] G. Zhang, X. Ou, M. Cui, Q. Wu, S. Ma, and W. Chen, "Cooperative UAV enabled relaying systems: Joint trajectory and transmit power optimization," *IEEE Trans. Green Commun. Netw.*, vol. 6, no. 1, pp. 543–557, 2022.
- [8] H. Ghazzai, M. Ben-Ghorbel, A. Kessler, and M. J. Hossain, "Trajectory optimization for cooperative dual-band UAV swarms," in *IEEE Global Commun. Conf.*, 2018, pp. 1–7.
- [9] J. Lee and V. Friderikos, "Trajectory planning for multiple UAVs in UAV-aided wireless relay network," in *IEEE Int. Conf. Commun.*, 2022, pp. 1–6.
- [10] E. Yanmaz, "Positioning aerial relays to maintain connectivity during drone team missions," *Ad Hoc Networks*, vol. 128, pp. 102800, 2022.
- [11] H. Yang, J. Zhang, S. H. Song, and K. B. Lataief, "Connectivity-aware UAV path planning with aerial coverage maps," in *IEEE Wireless Commun. Netw. Conf.*, 2019, pp. 1–6.
- [12] L. E. Kavraki, P. Svestka, J. C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Trans. Robot. Autom.*, vol. 12, no. 4, pp. 566–580, 1996.
- [13] O. V. P. Arista, O. R. V. Barona, and R. S. Nuñez-Cruz, "Development of an efficient path planning algorithm for indoor navigation," in *Int. Conf. Electr. Eng., Comput. Sci. Automat. Control*, 2021, pp. 1–6.
- [14] D. Romero and S.-J. Kim, "Radio map estimation: A data-driven approach to spectrum cartography," *IEEE Signal Process. Mag.*, Nov. 2022.
- [15] N. Patwari and P. Agrawal, "NeSh: a joint shadowing model for links in a multi-hop network," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Las Vegas, NV, Mar. 2008, pp. 2873–2876.
- [16] N. Patwari and P. Agrawal, "Effects of correlated shadowing: Connectivity, localization, and RF tomography," in *Proc. Int. Conf. Info. Process. Sensor Networks*, St. Louis, MO, Apr. 2008, pp. 82–93.