INL/JOU-24-80952-Revision-0

Minimizing Ground Risk in Cellular-Connected Drone Corridors With mmWave Links

December 2023

Simran Singh, Mihail L. Sichitiu, Ismail Guvenc, Arupjyoti Bhuyan

Changing the World's Energy Future

INL is a U.S. Department of Energy National Laboratory operated by Battelle Energy Alliance, LLC

INL/JOU-24-80952-Revision-0

Minimizing Ground Risk in Cellular-Connected Drone Corridors With mmWave Links

Simran Singh, Mihail L. Sichitiu, Ismail Guvenc, Arupjyoti Bhuyan

December 2023

Idaho National Laboratory
Idaho Falls, Idaho 83415

http://www.inl.gov

Prepared for the
U.S. Department of Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

Minimizing Ground Risk in Cellular-Connected Drone Corridors with mmWave Links

Simran Singh†, Mihail L. Sichitiu†, İsmail Güvenç†, and Arupjyoti Bhuyan*†Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC
*Idaho National Laboratory, Idaho Falls, ID
{ssingh28,mlsichit,iguvenc}@ncsu.edu, arupjyoti.bhuyan@inl.gov

***Abstract*—Unmanned aircraft systems (UAS) have been receiving significant interest and support from academia, industry, and regulatory bodies over the past decade due to their various use cases. To safely integrate UAS operations into the national airspace, particularly over populated regions, the risk posed to ground users, buildings, and vehicles due to unmanned aerial vehicle (UAV) flight should be minimized. This risk can be represented by a numerical metric, which we refer in this paper as the "ground risk". Many UAS applications also depend on the presence of a reliable wireless communication link between the UAV and a control station for the transmission of UAV position, surveillance video, UAV payload commands, and other mission-related data. Such wireless communication requirements also need to be considered in the design of UAS operations. In this work, we consider both these aspects and study the design of non-intersecting trajectories for UAS operations to minimize ground risk, subject to constraints on the wireless signal strength and geometry of the trajectory, specified in terms of: i) an enclosing cylinder within which the trajectory must lie, and ii) an integrated angular change along the UAV's trajectory. The performance of a computationally expensive optimal algorithm is compared with two computationally faster heuristic approaches within the dense urban environment of Manhattan, NY. Performance evaluation using ray tracing simulations shows that the heuristic approaches perform close to the optimal algorithm at a reduced computation cost. This research can be utilized to make UAS operations safe and reliable, and accelerate their adoption.**

***Keywords*—A* search, beamforming, conflict based search (CBS), drone corridor, ground risk, mmWave, multi agent path finding (MAPF), unmanned aircraft system (UAS), UTM.**

I. Introduction

Drone corridors, imagined as highways in the sky [1], are regions of airspace designated for supporting unmanned aerial vehicle (UAV) operations. Such corridors can support various commercial and civil applications, for example to connect production lines and reduce transportation costs [2]. To develop a formal architectural framework for such drone corridors, the National Aeronautics and Space Administration (NASA) developed the Unmmaned Aircraft System Traffic Management (UTM) framework between 2014 to 2021 [3]. This concept is also being adopted globally, with implementations reported in Europe, India, Singapore, Japan, and Australia. The UTM framework integrates UAV flight at low altitudes into the national airspace, and the Federal Aviation Authority (FAA) in the United States (US) has started implementing this framework. For instance, the FAA implemented the Low Altitude Authorization and Notification Capability (LAANC) at 400 air traffic facilities covering 600 US airports [4]. Through LAANC, registered commercial drones are granted access to the controlled airspace under 400 feet, and pilots are instructed as to where they can and cannot fly.

Various proof-of-concept drone corridors have been designed across the globe. For instance, the states of Michigan in the US and Ontario in Canada announced a feasibility study on the creation of a drone corridor for just-in-time delivery and medical transport across the US-Canada border [5]. This project is envisioned to ensure faster product deliveries and reduce supply chain disruptions. Besides this initiative, seven drone corridor test sites have been designed in the US, under the Unmanned Aircraft System (UAS) Test Site Program, to test and accelerate the integration of drone corridors into the national airspace. In the United Kingdom (UK), the National Beyond Visual Line of Sight (BVLOS) Experimentation Center (NBEC) created a drone corridor for UAV experiments and to test their integration to controlled and uncontrolled airspace [6]. In January 2022, a UAV flew beyond the line of sight for a distance of 16 km along this corridor, as part of a trial run [7]. It is predicted that 76,000 UAVs will be flying across the UK skies by 2030 [8]. In the US, the drone industry is predicted to set to be worth $92 billion by 2030. Given the effort and resources spent by the government and the industry to make drone corridors operational as well as their economic and humanitarian potential, it is timely to study how they can be designed, especially to ensure reliability and safety.

In January 2022, the FAA granted permission for BVLOS UAV operations within 35 miles of the 50 mile long drone corridor in New York [9]. BVLOS operations are essential to make commercial drone operations economically viable. Uses of BVLOS drone flights include wind turbine blade inspection, traffic monitoring, search-and-rescue, power line surveys, and medical and package deliveries [7]. UAVs can be tracked wirelessly rather than visually in BVLOS operations, requiring the presence of a reliable wireless communication link between the UAV and the ground station, to ensure command and control. For this purpose, 5G cellular technology is being evaluated for suitability to the UAS industry [10], [11]. The

This work is supported in part by the INL Laboratory Directed Research Development (LDRD) Program under DOE Idaho Operations Office Contract DEAC07-05ID14517, and by NSF under the grant numbers CNS-1453678 and CNS-1738093.

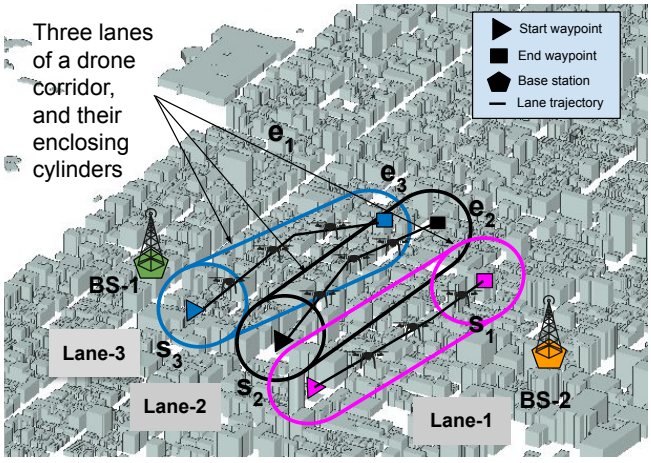


Fig. 1: A drone corridor with three lanes. The start and end coordinates of each lane and their enclosing cylinders are shown, along with two BSs serving the corridor. While the enclosing cylinders overlap partially, the lane trajectories do not intersect with each other. The corridor lies within the dense urban environment of Manhattan.

millimeter wave (mmWave) frequencies used in 5G, along with achieving very high data rates, also allow smaller form factor for antenna elements due to the high frequencies employed. This smaller form factor, in turn, allows the use of massive antenna arrays, and digital beam-forming techniques to track UAVs accurately. Hence, it is necessary to research the use of mmWave frequencies for establishing wireless broadband communication links in drone corridors.

When UAVs fly in drone corridors over populated regions, it is important to minimize the risk to ground users, vehicles, and pedestrians. In Michigan, more than 2.7 million residents and 283 cities and townships live and operate under controlled airspace. One such community, the city of Taylor, is home to 62,000 people and operates completely within federally controlled airspace [12]. Considering ground risk when designing drone corridors in the airspace over such residential regions would alleviate privacy and safety concerns and make the concept more likely to gain acceptance.

For the purpose of this work, we define a drone corridor to be a set of non-intersecting three dimensional (3D) lanes along which UAVs fly. We introduce a term *elasticity*, denoted as ϵ, and define it as the radius of a cylinder within which each lane is constrained to lie. The axis of this cylinder is along the line joining the start and end waypoints of that lane. We study the design of the trajectory of each lane to minimize the sum of the integrated ground risk across all lanes, while satisfying constraints on the received signal strength (RSS) along each lane, on the integrated angular change along each lane, and while ensuring that each lane lies within its enclosing cylinder. Additionally, we also study wireless coverage in drone corridors, when using BSs that operate at mmWave frequencies. An example of a drone corridor with three lanes is shown in Fig. 1, along with the enclosing cylinders of each lane and two serving base stations (BS). Imposing a constraint on the integrated angular change along a lane enables the drone corridor designer to indirectly influence the shape and total curvature of the trajectory. The maximum integrated angular change can be tuned by considering UAV manoeuvrability and it can be used to prevent sharp abrupt changes in the trajectory, which may result if the optimization algorithm purely focuses on minimizing ground risk.

The rest of the paper is organized as follows. Our work is compared with other related research in the literature in Section II and the theoretical drone corridor framework is developed in Section III. The drone corridor trajectory optimization problem and trajectory search algorithms are presented in Section IV. In Section V, we provide insights on wireless coverage within the corridor, and in Section VI, we evaluate the performance of the proposed solutions in terms of cost and performance. Finally, Section VII concludes our work.

II. Literature Review

In the literature, UAV trajectory design has been studied for various objectives in the literature [13]–[17]. The trajectory of a single UAV was optimized in [13], to maximize the number of Internet of Things (IoT) devices served, such that the target data upload deadline of each IoT device is met. The energy expended by a UAV was minimized in [14], while ensuring that the signal to noise ratio (SNR) is high enough at the desired ground node, and low enough at malicious nodes to avoid eavesdropping. A massive machine type communication system (mMTC) was considered in [16], wherein the trajectory for a single UAV was optimized for various objectives, such as to minimize the transmit energy of mMTC nodes and maximize the number of tasks completed by the UAV before running out of battery. UAV trajectory planning with 6G communications was studied in [15] to maximize the sum upload rate of all devices in an IoT system. Dynamic programming was used in [17] to design the trajectory of a UAV acting as a relay between ground users, to maximize the spectral efficiency of the network, while taking into account backhaul constraints and the 3D antenna radiation pattern. However, in each of the above studies, only a single UAV was considered.

Multi-UAV trajectory design has been studied in [18]–[21], [26]. Authors in [19] considered a cellular Internet of UAVs wherein UAVs equipped with vision-based sensors detect stationary targets and transmit these measurements to a BS. UAVs were used similarly in [26] to gather data from road-side units, with the objective of minimizing the variation in flight time across all UAVs. Deep learning techniques were utilized to calculate the trajectory of multiple UAVs to maximize the freshness of sensing results, using a metric termed the accumulated age of information. The use of UAVs as cellular BSs to serve ground users was studied in [18], [20], [21], all of which calculated the optimum trajectories of multiple UAVs. The studies [20], [21] maximized the downlink throughput of ground users. While [20] optimized UAV trajectory in conjunction with user association and resource allocation, [21]

TABLE I: Literature review on UAV trajectory optimization and drone corridors.

Ref.	Trajectory design optimization criteria	Trajectory design constraints	mmWave frequencies	Kinematic constraints	Multiple, distinct trajectories	Ground risk	Role of UAV
[13]	Maximize number of IoT devices served	Data uploaded by each IoT device should be greater than a threshold	×	×	×	×	BS
[14]	Minimize energy expended by UAV	SNR at ground node to be protected should be greater than a threshold, while SNR at eavesdroppers, both individually and with maximal ratio combining, should be less than a threshold	×	×	×	×	BS
[15]	Maximize the sum upload rate of all IoT devices in the network	Upload rate of each IoT device should be greater than a threshold	×	×	×	×	BS
[16]	Minimize transmit energy of mMTC nodes, minimize energy consumed by UAV, minimize latency of all nodes served by UAV, maximize number of tasks completed by UAV within its battery-constrained flight time	SNR at UAV, from the node, should be greater than a threshold	×	×	×	×	BS
[17]	Maximize the sum of spectral efficiency of all served users, calculated over the UAV flight time	Maximum UAV velocity, start and end waypoints of UAVs path, UAV height	×	×	×	×	Relay
[18]	Minimize total energy consumption of all UAVs	QoS requirement, specified in terms of amount of content to be delivered within a specific time interval, of each ground user should be met	×	×	✓	×	BS
[19]	Minimize age of information.	IoT nodes should be within maximum sensing angle of UAV	×	×	×	×	Relay
[20]	Maximize minimum throughput across all ground users	Transmission power of UAVs should be less than a threshold	×	×	✓	×	BS
[21]	Maximize sum of instantaneous transmission rates	Transmission rate of each user should be greater than a threshold	×	×	✓	×	BS
[22]	Maximize sum of downlink rates, calculated over all IoT devices	Transmission rate of each user should be greater than a threshold, sum of uplink transmission powers of all devices should be less than a threshold	✓	×	×	×	BS
[23]	Maximize ratio of uplink data rate and distance to the destination	None	✓	×	×	×	UE
[24]	Minimize flight time	Network throughput should be greater than a threshold	✓	×	×	×	UE
[25]	Not applicable (NA)	NA	×	×	×	✓	NA
This work	Minimize ground risk	Angular change over trajectory must be less than a threshold, RSS along the trajectory must be greater than a threshold	✓	✓	✓	✓	UE

optimized UAV transmit power along with trajectory. The study in [18], on the other hand, minimized the total energy consumption of all UAVs under QoS constraints, in unknown environments using decentralized reinforcement learning (RL) techniques. However, none of the above works considered ground risk, nor did they utilize mmWave frequencies.

The integration of UAVs into the 5G cellular network was summarized in [27]–[29] and UAV trajectory optimization, when using mmWave communication links, was studied in [22]–[24]. UAV path planning algorithms were surveyed and categorized in [27], and a smart city system was proposed, which utilized UAVs and IoT devices to mitigate the effect of disasters. The review in [28] focused on the integration of UAVs into the 5G network, categorizing existing work and frameworks into either UAV-assisted wireless communications or cellular-connected UAVs. Physical and network layer aspects of the use of UAVs in 5G systems was summarized in [29]. An IoT system was considered in [22], wherein UAVs acted as BSs, and the downlink sum rate was maximized by optimally placing these UAVs, tuning their transmission power and adapting their beam pattern. Deep Q learning was used in [23] to calculate the trajectory of a UAV, within the coverage region of an mmWave BS, to maximize the ratio of the uplink data rate and distance to the destination. Authors in [24] planned the path of a UAV to meet the Quality of Service (QoS) requirements of surveying an area using a

camera on board the UAV. While these studies considered mmWave frequencies, they did not incorporate ground risk in their model.

Risk assessment models for UAV operations over roads were developed in [25]. These models calculated the probability that a UAV crashing onto a road results in the crash of a road user, while taking into account the road type and the time of the day. A simple risk index was calculated and used to identify risky areas. This study was extended in [30] to generate ground impact probability maps using Monte Carlo simulations and neural network models, while also taking into account wind conditions. While these studies did not optimize UAV trajectory to minimize ground risk, their proposed models can be leveraged by future work. The comparison of the related work on drone corridors and trajectory design is summarized in Table I.

III. System Model

In this section, we develop a mathematical model of the drone corridor and introduce the concept of ground risk. Then, the uplink wireless communication model is presented.

A. Drone Mobility Model

We consider the region of study that is in the shape of a cuboid, denoted as ν, of dimensions ν_x, ν_y, and ν_z. This region of space is sampled at discrete intervals of δ_x along the x axis, δ_y along the y axis, and δ_z along the z axis. This sampling interval determines both the granularity at which communication parameters such as the RSS are calculated and the number of points in the 3D grid for path search. Thus, the sampling interval determines computational complexity of analysis. With each 3D coordinate, denoted as $\mathbf{x} = (x, y, z) \in \nu$, we associate a cost $c_{\mathrm{g}}(\mathbf{x})$, as a measure of the ground risk at that location. There may be regions of space within ν where UAVs are not permitted to fly, termed as no-fly-zones, such as above government buildings and airports, among other areas. In such regions, $c_{\mathrm{g}}(\mathbf{x}) = \infty$.

The drone corridor is assumed to consist of N_{L} lanes (volumes of cylindrical segments as shown in Fig. **??**, where the i^{th} lane ($1 \leq i \leq N_{\mathrm{L}}$) is denoted as L_i. Each lane in the corridor is defined as a set of waypoints, where the source and destination waypoints of each lane are pre-determined, and the path-finding algorithm chooses intermediate waypoints within that lane that satisfy the geometry and wireless communication constraints. We denote the set of waypoints as $\mathbf{x}_{\mathrm{W},i}$ for the i^{th} lane. The ground risk for the i^{th} lane, denoted as $c_{\mathrm{g},i}$ is then calculated as:

$$c_{\mathrm{g},i} = \int_{\mathbf{x}_{\mathrm{W},i}} c_{\mathrm{g}}(\mathbf{x}) \mathrm{d}\mathbf{x}. \tag{1}$$

It is assumed that UAVs fly in a straight line path between any two successive waypoints.The geometric constraints that are associated with the lanes are described next.

To allow for multiple angular changes, from any given waypoint (x, y, z), we allow the UAV to transition to $(x + n_{\mathrm{x}}\delta_x, y + n_{\mathrm{y}}\delta_y, z + n_{\mathrm{z}}\delta_z)$, where n_{x}, n_{y}, and n_{z} are integers

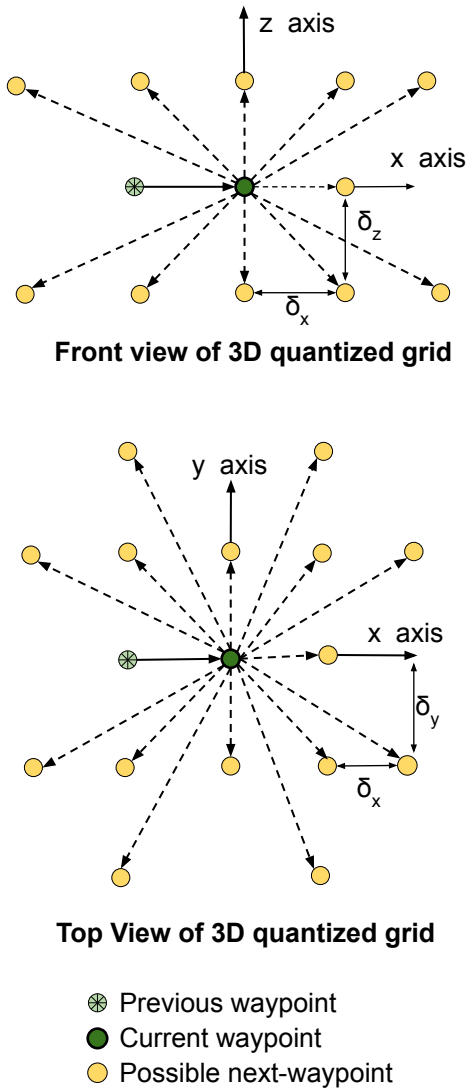


Fig. 2: 3D quantized grid for $N_{\mathrm{wp}} = 2$. Both the front and top views are shown.

under certain constraints. This decision goal can be expressed as follows:

$$\begin{aligned}
&\text{Find } (n_{\mathrm{x}}, n_{\mathrm{y}}, n_{\mathrm{z}}), \text{s. t. :} \\
&\qquad (-N_{\mathrm{wp}} \leq n_{\mathrm{x}} \leq N_{\mathrm{wp}}), \\
&\qquad (-N_{\mathrm{wp}} \leq n_{\mathrm{y}} \leq N_{\mathrm{wp}}), \\
&\qquad n_{\mathrm{z}} \in \{-1, 0, 1\}, \\
&\qquad n_{\mathrm{x}} \in \{-1, 1\}, \text{ for all } n_{\mathrm{y}} = 0, \\
&\qquad n_{\mathrm{y}} \in \{-1, 1\}, \text{ for all } n_{\mathrm{x}} = 0, \\
&\qquad n_{\mathrm{x}} \neq n_{\mathrm{y}}, \text{for all}(|n_{\mathrm{x}}| > 1, |n_{\mathrm{y}}| > 1),
\end{aligned} \tag{2}$$

where N_{wp} is an integer that represents the number of hops considered when choosing the next waypoint. The constraints

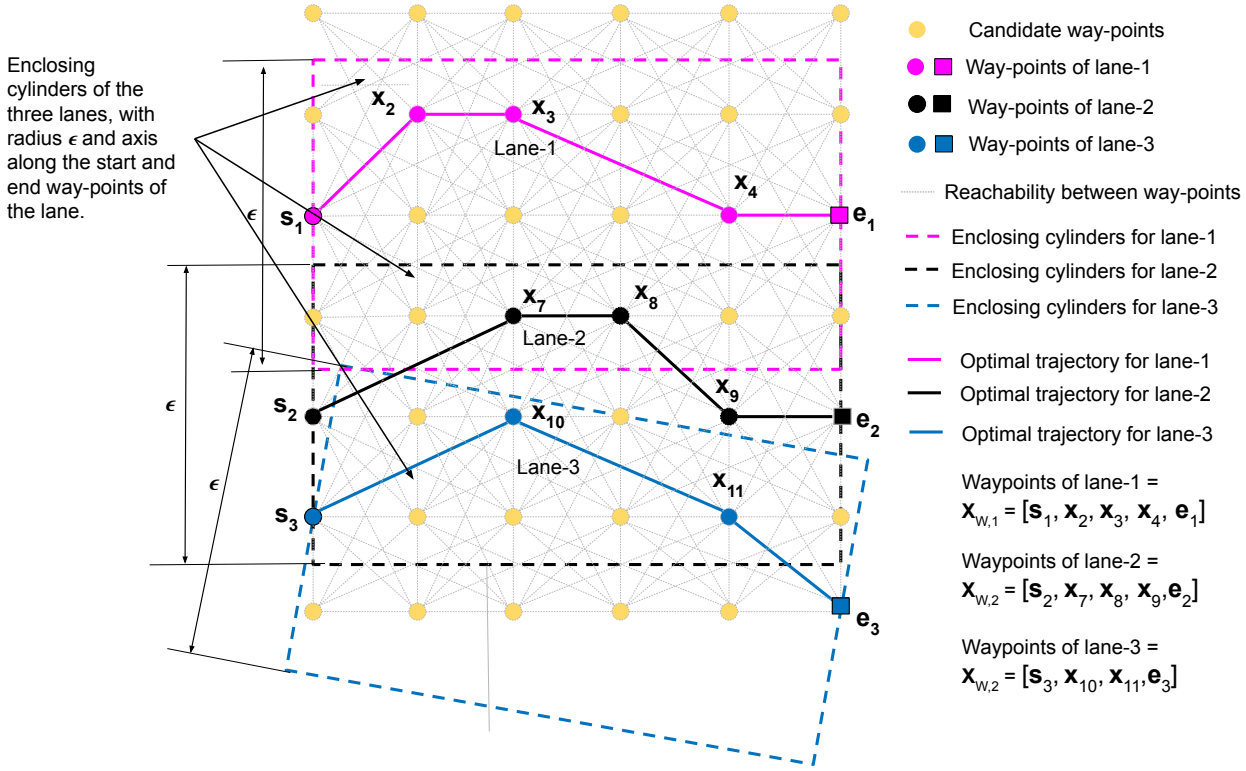


Fig. 3: Top view of a drone corridor with three lanes. The reachability grid is shown, for $N_{\mathrm{WP}} = 2$. The start and end coordinates of each lane and their enclosing cylinders are also shown.

imply that the possible next waypoints among all given way-points are those that lie within N_{wp} hops along the x and y, and are along the z axis or one hope above or below it. The last constraint in (2) removes redundant next waypoints. Increasing N_{wp} provides for a larger set of valid next waypoints, and thus more options of angular change, allowing greater flexibility in designing the trajectory, at the cost of higher computational complexity. Further, we assume each lane to lie within a cylinder of radius ϵ (see Fig. 3), whose axis is formed by the line joining the start and end waypoints of that lane. Let this cylinder, for the i^{th} lane, be denoted as $\mathcal{C}_{\epsilon,i}$ and its axis length as l_i. Then, for any waypoint $\mathbf{x}$ in the i^{th} lane, the constraint is specified as follows:

$$\mathbf{x} \in \mathcal{C}_{\epsilon,i}, \tag{3}$$

for all $\mathbf{x} \in \mathbf{x}_{\mathrm{W,i}}$. We term ϵ as the *elasticity* of the lane. We denote the angular change at any waypoint $\mathbf{x}$ along the i^{th} lane as $\Theta(\mathbf{x}, i)$, and impose a constraint on the integrated angular change along that lane as follows:

$$\int_{\mathbf{x}_{\mathrm{W,i}}} \Theta(\mathbf{x}, i)\mathrm{d}\mathbf{x} \leq \Theta_{\max}. \tag{4}$$

This constraint implies that UAVs travelling along the lane cannot turn more than $\Theta_{\max}$ in that lane, over the entire lane. While (4) imposes a constraint on the deviation from the straight line path in terms of angular curvature, (3) determines the extent to which the lane can deviate from the straight line path in terms of euclidean distance.

Fig. 3 shows the top view of a drone corridor with three lanes, the reachability grid for $N_{\mathrm{WP}} = 2$, and the enclosing cylinder of each lane. Add more.

B. Wireless Coverage in Drone Corridor

Wireless service is provided to the drone corridor by N_{BS} cellular BSs on the ground. Let $\mathbf{x}_{\mathrm{BS}} \in \mathbb{R}^{N_{\mathrm{BS}} \times 3}$ represent the 3D coordinates of these BSs. These BSs are assumed to operate in the mmWave spectrum centered at f_{c} and with a bandwidth of B. Each BS is equipped with three $N_{\mathrm{a}} \times N_{\mathrm{a}}$ universal planar antenna arrays (UPAs). Let $k \in [1, 2, 3]$ identify each of these three antenna arrays. These three UPAs are separated by 120° in the horizontal plane to create three sectors, thus providing 360° coverage. Each antenna array at a BS is fixed at a deterministic direction with no tilting (no mechanical steering is considered) and we assume that each UAV has a single omni-directional antenna.

Assume that the k^{th} antenna array of BS-j is using digital beamforming to collimate a beam to a UAV located at $\mathbf{x}$, and that the instantaneous channel state information (CSI) is known at the BS, i.e. BS-j knows the downlink channel matrix in (5) from each of its three antenna arrays to the UAV located at $\mathbf{x}$. The downlink channel matrix from the k^{th} antenna array of

BS-j to a UAV located at $\mathbf{x}$ is denoted as $\mathbf{h}_{j,k}(\mathbf{x}) \in \mathbb{C}^{N_{\mathrm{a}}^2 \times 1}$ and the corresponding beamforming vector to coordinate $\mathbf{x}$ is given by $\mathbf{w}_{j,k}(\mathbf{x}) \in \mathbb{C}^{N_{\mathrm{a}}^2 \times 1}$. The channel matrices from each antenna array of each BS to all possible receiver locations are obtained by performing raytracing simulations. Then, the signal strength received from the k^{th} antenna array of BS-j at this UAV located at $\mathbf{x}$, denoted as $S_{j,k}(\mathbf{x})$, can be calculated as

$$S_{j,k}(\mathbf{x}) = \left| [\mathbf{w}_{j,k}(\mathbf{x})]^{\mathrm{H}} \mathbf{h}_{j,k}(\mathbf{x}) \right|^2 , \tag{5}$$

where the Hermitian (conjugate transpose) of a vector is represented by $[.]^{\mathrm{H}}$. A UAV located at $\mathbf{x}$ associates with the strongest sector amongst all the three sectors of all BSs, i.e.

$$S(\mathbf{x}) = \max_{j,k} \; S_{j,k}(\mathbf{x}) . \tag{6}$$

Based on this association rule, the set of all coordinates that are served by the k^{th} antenna array of BS-j can be found, which we denote as $\mathbf{a}_{j,k}$. The drone corridor communication system was analyzed in the downlink in [31] by the authors. The uplink analysis is presented in this work.

Assume that a UAV positioned at location $\mathbf{x}$ is associated with, and transmitting to the k^{th} antenna array of BS-j on the uplink. We assume that orthogonal uplink multiplexing is used, such that interference to the k^{th} antenna array of BS-j comes from UAVs associated with and transmitting on the uplink to antenna arrays of other BSs. For simplicity, all UAVs are assumed to transmit at a power level P watts, using an omnidirectional antenna, and we assume wireless communication channel reciprocity, i.e. the uplink channel is the same as the downlink channel. We denote these BSs, to which interfering UAVs transmit, as BS-m and the resultant interference to k^{th} antenna array of BS-j as $I_{j,k}$. Then, $I_{j,k}$ can be calculated as follows:

$$I_{j,k} = \sum_{k,m \neq j} P[\mathbf{h}_{j,k}(\mathbf{x}_{m,k})]^{\mathrm{H}} \mathbf{h}_{j,k}(\mathbf{x}_{m,k}) , \tag{7}$$

where $\mathbf{x}_{m,k} \in \mathbf{a}_{m,k}$ identifies the location of the UAV currently being served by the k^{th} antenna array of BS-m.

As the interference is a function of the number of UAVs served in the corridor, we also introduce a parameter, namely traffic density, which we define to be the number of UAVs per meter supported by each lane of the corridor, and which we denote as λ. We denote the number of UAVs served by the k^{th} antenna array of BS-j by $N_{j,k}^{\mathrm{U}}$, which can be calculated by considering the length of all the lanes served by this antenna array, given λ. Assuming that a minimum mean square error (MMSE) receiver is used at each BS, and that achievable data rates are represented by Shannon capacity, the uplink data rate for the UAV at location $\mathbf{x}$, transmitting to the k^{th} antenna array of BS-j, is calculated as:

$$R(\mathbf{x}) = \frac{B}{N_{j,k}^{\mathrm{U}}} \log_2 \left(1 + P \mathbf{h}_{j,k}(\mathbf{x})^{\mathrm{H}} (\sigma_n^2 + I_{j,k})^{-1} \mathbf{h}_{j,k}(\mathbf{x})\right) , \tag{8}$$

where $R(\mathbf{x})$ denotes the data rate at $\mathbf{x}$, and noise is modelled by a zero-mean Gaussian random variable with variance σ_{n}^2. Since $I_{j,k}$ is a random variable, $R(\mathbf{x})$ is also a random variable. The distribution of $R(\mathbf{x})$ can be calculated by considering all possible combinations of interfering UAV locations, $\mathbf{x}_{m,k}$ in (7).

IV. Drone Corridor Trajectory Optimization

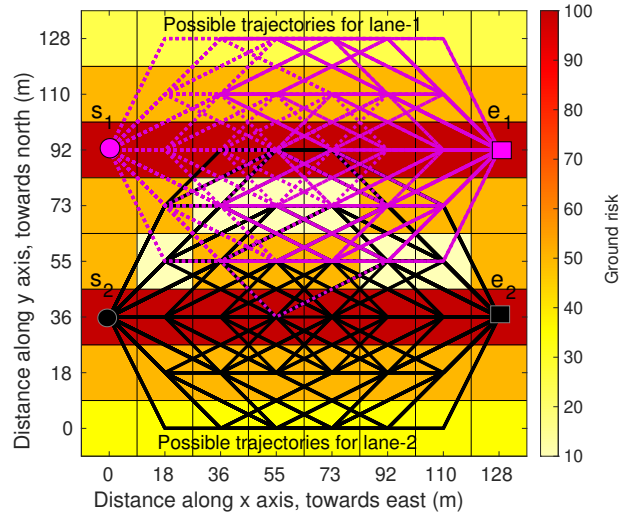
In this section, we study the problem of calculating the intermediate waypoints of each lane given their start and end waypoints. The optimization goal is to minimize the sum of ground risk, calculated along all lanes, while ensuring that the geometric and RSS constraints of each lane are satisfied. This is expressed mathematically as follows:

$$\begin{aligned} \text{minimize } & \sum_{i=1}^{N_{\mathrm{L}}} c_{\mathrm{g,i}}(\mathbf{x}_{\mathrm{W},i}) \\ \text{such that: } & c_1 : S(\mathbf{x}) \geq S_{\mathrm{Thresh}}, \; \forall \, \mathbf{x} \in \mathbf{x}_{\mathrm{W,i}} \\ & c_2 : \int_{\mathbf{x}_{\mathrm{W,i}}} \theta(\mathbf{x}, i) \mathrm{d}\mathbf{x} \leq \theta_{\max} \\ & c_3 : \mathbf{x} \in \mathcal{C}_{\epsilon,i}, \\ & c_4 : \mathbf{x}_{\mathrm{W,i}} \cap \mathbf{x}_{\mathrm{W,j}} = \emptyset, \text{for } j = 1 \text{ to } N_{\mathrm{L}}, \; j \neq i \; , \end{aligned} \tag{9}$$

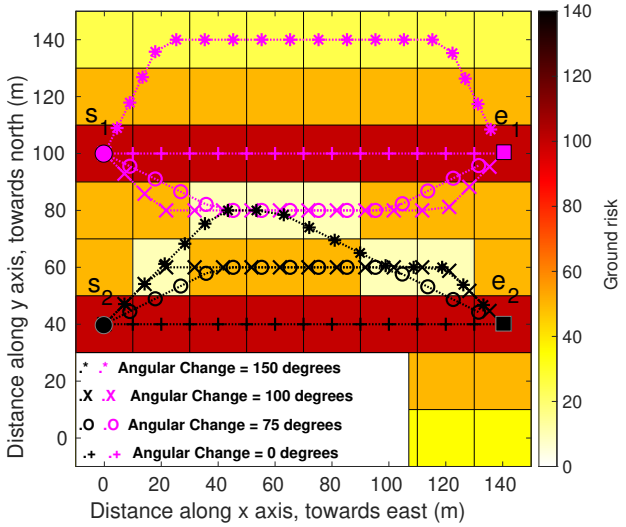
where the constraints c_1 - c_4 need to be satisfied for each lane. The RSS constraint is captured by c_1, while the geometric constraints on angular change and the deviation from the straight-line-path by c_2 and c_3 respectively. The constraint c_4 implies that lanes should not intersect. The optimization problem in (9) is classified as a multi-agent path finding problem (MAPF) with kinematic constraints [32]. Constraint c_2 implies that the possible next waypoints from the current waypoint also depends on the trajectory followed by the agent to reach the current waypoint, thus imposing a kinematic constraint. As an example, Fig. 4a shows the various possible trajectories between a set of two start and end waypoints that satisfy an elasticity constraint of 50 m, while Fig. 4b shows the optimum trajectories that minimize ground risk for various angular change thresholds. The trajectories are overlaid on a heatmap of the ground risk. We present optimal and sub-optimal algorithms to solve this MAPF in the next sub-section.

A. Optimal Trajectory Search

Conflict based search (CBS) [33] is an optimal MAPF solver that can be used to solve the trajectory search problem in (9). Solution to MAPF problems using CBS requires two levels of search as follows. On the high level, CBS creates a conflict tree based on conflicts between the trajectories of individual agents. This tree is a binary tree, wherein each parent node has two child nodes. Each node represents a candidate solution to the trajectory optimization problem in (9), and consists of three attributes. The first attribute includes the *intersection constraints*, a set of constraints where each constraint consists of a lane and a line segment lane, such that this lane is constrained not to intersect with the line segment. The second attribute includes the *candidate paths*, a set of N_{L} paths, one for each lane, calculated by using a single-agent path finding algorithm such that each lane satisfies its corresponding intersection constraints, i.e. none of the lanes



(a)



(b)

Fig. 4: A scenario to demonstrate the trajectory optimization problem. (a) shows all possible trajectories for two lanes of a drone corridor that satisfy the elasticity constraint. (b) shows the optimal trajectories, from the previous set of all possible trajectories, that minimize ground risk for various angular change thresholds.

intersect with those line segments of other lanes that are listed in the *intersection constraints*. The third attribute is the *cumulative ground risk cost* of the node, calculated as the sum of the integrated ground risk along the candidate paths of each lane.

Algorithm 1 Conflict based search (CBS), for high level search [33]

Nodes $= \emptyset$.
`Root.IntersectionConstraints` $= \emptyset$.
`Root.CandidatePaths` = Individual lane paths that satisfy `Root.IntersectionConstraints`, found using low level search.
`Root.Cost` = Sum of integrated ground risk cost along `Root.CandidatePaths`.
Add `Root` to `Nodes`.
while *Nodes* $\neq \emptyset$ **do**
  `CurrentNode` = Node with the lowest `Cost` from `Nodes`.
  if no intersections in `CurrentNode` **then**
    return `CurrentNode.CandidatePaths`.
  else
    forall Lane in the first intersection **do**
      `ChildNode.IntersectionConstraints` = `CurrentNode.IntersectionConstraints`.
      Add intersecting line segment to `ChildNode.IntersectionConstraints`.
      `ChildNode.CandidatePaths` = Lane paths that satisfy `ChildNode.IntersectionConstraints`, found using low level search.
      `ChildNode.Cost` = Sum of integrated ground risk cost along `ChildNode.CandidatePaths`.
      Add `ChildNode` to `Nodes`.
    end
  end
end

This conflict tree is generated by first starting with a root node, which consists of an empty set of *intersection constraints*, a set of locally optimal paths for each lane without any intersection constraints, and the corresponding *cumulative ground risk cost*. This root node is then evaluated as follows. The *candidate paths* are evaluated, two at a time, for intersections. If no intersections are found, this is returned as the solution. If an intersection is found, then two new child nodes are generated as follows. The set of *intersection constraints* are copied over to each of the two child nodes. For the first child node, an additional constraint is added, describing one of the lanes and the intersecting line segment of the other lane, and vice-versa for the other child node. The first intersection between the two lanes is considered when adding the above constraints. The *candidate paths* of each of these two child nodes are then calculated by using a single-agent path finding algorithm, while satisfying the corresponding *intersection constraints*. The *cumulative ground risk cost* of each node is updated accordingly. These child nodes are added into an *open set* and sorted as per the *cumulative ground risk cost*. The node with the lower *cumulative ground risk cost* is then evaluated using the same steps as described above for the root node. The process continues till a node with no conflicts is found, which is then returned as the solution. The steps of CBS are listed in Algorithm 1.

Algorithm 2 Modified Heuristic-Based Best-First Search (M-HBFS), for low level search

```
Paths = ∅
Add start waypoint to Paths
while Paths ≠ ∅ do
    CurrPath = Lowest-cost path in Paths, where cost = integrated ground risk over the path + estimated cost from the last waypoint of the path to the end waypoint.
    Remove CurrPath from Paths.
    if Last waypoint of CurrPath is the end waypoint then
        return CurrPath
    end
    CandidatePaths = Set of paths obtained by expanding CurrPath, considering all possible N_WP hop neighbours of the last waypoint of the path.
    forall CandidatePath ∈ CandidatePaths do
        if CandidatePath satisfies constraints c1, c2, c3, c4 then
            Insert CandidatePath to Paths
        end
    end
end
```

At the low level, CBS relies on single-agent path finding algorithms to calculate the path for each agent independently while satisfying the intersection constraints on its motion imposed by the high level conflict tree node. In this work, a modified version of heuristic-based best-first search (M-HBFS) is used as the optimal low-level search, wherein the set of next waypoints from the current waypoint is pruned based on the current path and the constraints c_1-c_4. This algorithm branches out from the start waypoint, finding all valid sub-paths to reach intermediate waypoints. The lowest cost sub-paths are expanded at each iteration till the end waypoint is reached. Sub-paths are ordered based on the sum of the integrated ground-risk of the sub-path and a heuristic estimate of the cost from the end of the sub-path to the goal. To ensure that the algorithm is optimal and complete, this goal estimate must be admissible as well as consistent. This estimate is calculated as the straight line distance to the end waypoint, weighted by the ratio of minimum to maximum ground risk cost within the drone corridor. The steps of M-HBFS are listed in Algorithm 2.

Algorithm 3 Greedy algorithm for high level search

```
LaneTrajectories = [ ]
for i = 0; i < N_L; i++ do
    for j = 0; j < i; j++ do
        Remove locations in LaneTrajectories[j] from the search space.
    end
    LaneTrajectory = Trajectory for the i^th lane, found using low level search.
    Append LaneTrajectory to LaneTrajectories.
end
return LaneTrajectories
```

The worst-case computational complexity of CBS is proven to be exponential in the number of agents and the number of nodes in the graph [33], i.e., for our scenario, the run-time of CBS is exponential in the number of candidate waypoints and the number of lanes in the corridor. In the worst-case, when all the least cost paths do no satisfy this angular change constraint, M-HBFS has to evaluate *all* possible paths to the goal. Thus, its worst-case computational complexity is exponential in the number of candidate waypoints in the corridor, and the number of hops, N_{WP}, considered in the drone mobility model, both of which determine by the number of candidate paths to the goal.

Algorithm 4 Locally backtracking A* (LBA), for low level search

```
WayPtsToExplore = ∅.
WayPtsExplored = ∅.
PathsToWayPts = ∅.
while WayPtsToExplore ≠ ∅ do
    CurrWayPt = Lowest-cost waypoint in WayPointsToExplore, where cost = ground risk from start waypoint to the current point + estimated cost from the current point to end waypoint.
    Move CurrWayPt from WayPtsToExplore to WayPtsExplored.
    if CurrWayPt is the end waypoint then
        return Path to CurrWayPt
    else
        Neighbours = Set of N_WP hop neighbours of CurrWayPt that satisfy constraints c1, c3, c4
        forall Nei ∈ Neighbours do
            PathToNei = Path from start waypoint to Nei via CurrWayPt
            ValidPathExists = False
            if (Angular change over PathToNei ≤ Θ_max) ∧ (Ground risk cost over PathToNei ≤ Current cost to Nei)) then
                Insert PathToNei to PathsToWayPts, as the best path to Nei.
                ValidPathExists = True
            else if Angular change over PathToNei > Θ_max then
                AdjustedPathToNei = BacktrackLocally(PathToNei)
                if AdjustedPathToNei ≠ ∅ then
                    Insert AdjustedPathToNei to PathsToWayPts, as the best path to Nei.
                    ValidPathExists = True
                end
            end
            if (ValidPathExists) ∧ (Nei ∉ WayPtsToExplore) ∧ (Nei ∉ WayPtsExplored) then
                Add Nei to WayPtsToExplore
            end
        end
    end
end
```

B. Heuristic-Based Trajectory Search

As an alternate to CBS high level search, we consider a simple greedy algorithm, which sequentially finds paths for each lane, while ensuring that calculated path does not intersect with previously calculated paths. The greedy algorithm calculates the best path for the first lane. The best path for the second lane is then calculated by removing the waypoints of the first lane from the search space. Similarly, the waypoints

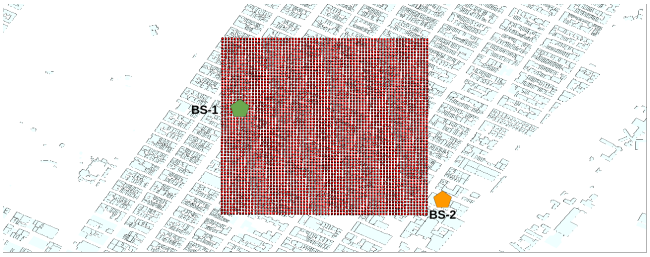
of the third lane are calculated by removing the waypoints of both the second and first lanes from the search space, and so on. Thus, the greedy algorithm invokes the lower level search algorithm exactly N_{L} times always, exhibiting polynomial time complexity, where N_{L} is the number of lanes in the corridor. The pseudocode of the greedy algorithm is presented in Algorithm 3.

Algorithm 5 Function for backtracking locally, to adjust a path to satisfy angular change constraint.

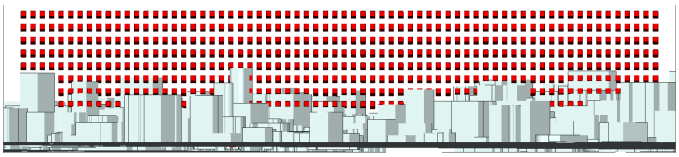
```
Input: Path
Output: AdjustedPath
Function BacktrackLocally(Path):
    MinCost = ∞
    AdjustedPath = ∅
    forall WayPt ∈ Path do
        AltWayPts = Set of alternate positions of WayPt that
          satisfy N_WP-hop neighbourhood with previous and next
          waypoints, and constraints c_1, c_3, c_4.
        forall AltWayPt ∈ AltWayPts do
            CandidatePath = Replace WayPt by
              AdjustedWayPt
            if Ground risk of CandidatePath < MinCost
            then
                AdjustedPath = CandidatePath
                MinCost = Ground risk of CandidatePath
            end
        end
    end
    return AdjustedPath
```

We introduce a non-optimal heuristic algorithm for the low level search, which we term locally backtracking A* (LBA). LBA prioritizes minimizing ground risk over satisfying the angular change constraint, and hence, it is not guaranteed to find a solution since its solution may or may not satisfy the angular change constraint.

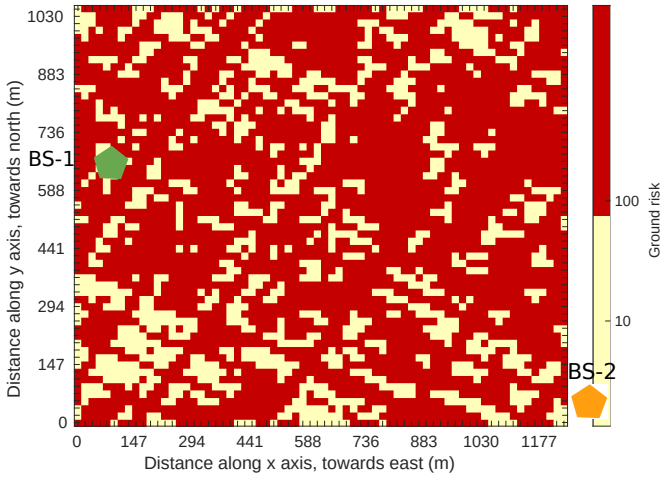
LBA low level search works similar to the conventional A* search, with the modification that waypoints that do not satisfy the RSS or elasticity constraints or those imposed by higher order search are pruned from the search space. When it encounters a waypoint for which the shortest path, as found by A*, does not satisfy the angular change constraint, the algorithm adjusts the shortest path to this waypoint such that the angular change constraint is satisfied. The path re-adjustment is performed as follows. For each waypoint in the shortest path, the algorithm considers all possible alternate N_{WP} positions, keeping all other waypoints in the shortest path fixed. Alternate paths are then generated through these alternate waypoints, and the algorithm chooses the path which results in the minimum ground risk while satisfying the angular change constraint. Essentially, waypoints in the shortest path are adjusted one at a time to generate alternate paths, and the one that has a valid angular change and minimizes ground risk is chosen. In case the angular change constraint cannot be satisfied, that waypoint is considered unreachable. Since this algorithm re-adjusts the path locally within the neighbourhood of the shortest path, it is non-optimal, i.e. it does not consider all possible alternate ways to reach the waypoint under consideration. The pseudocode for LBA is presented in Algorithm 4 and Algorithm 5. Table II summarizes the optimality and level of search of all the trajectory optimization algorithms considered in this work.



(a) Top view of the urban simulation area.



(b) Side view of the urban simulation area.



(c) Ground risk map.

Fig. 5: Simulation environment of Manhattan. (a) Top view, showing the receiver grid overlaid on the buildings and roads; (b) Side view of the receiver grid and the buildings; (c) Ground risk map, with a higher cost assigned to locations directly over streets and lower cost to locations directly over buildings.

TABLE II: Drone corridor trajectory design algorithms

Algorithm	**Search level**	**Optimality**
Conflict based search (CBS)	High level search	Optimal
Modified heuristic based search (M-HBFS)	Low level search	Optimal
Greedy	High level search	Non-optimal
Locally backtracking A* (LBA)	Low level search	Non-optimal

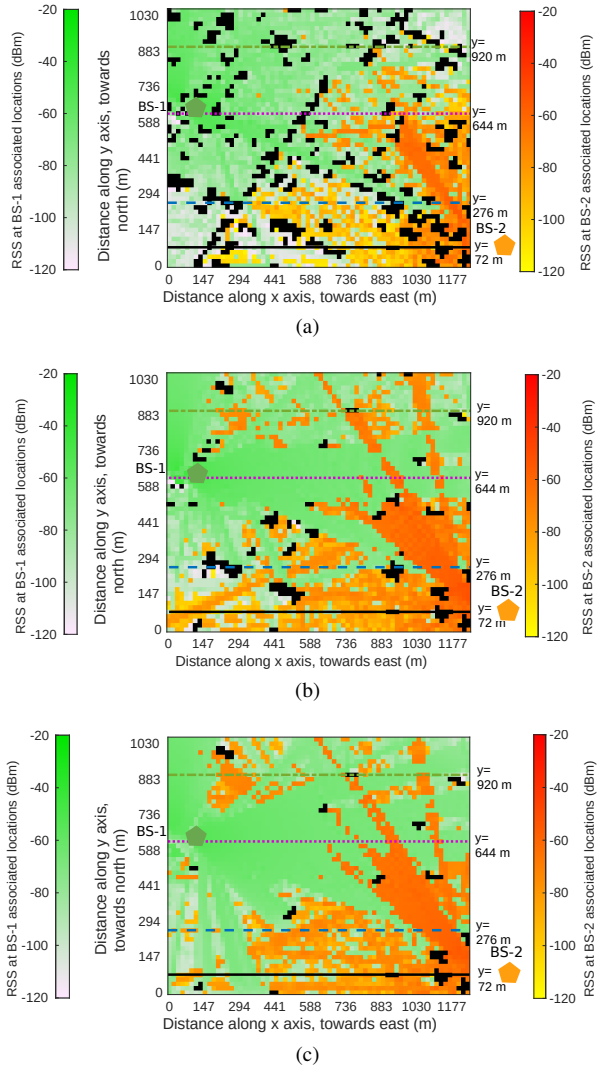


Fig. 6: Heatmap depicting the association of receiver locations with BS antenna arrays at altitudes of (a) 50 m, (b) 75 m, and (c) 100 m. BS locations are shown as green and orange pentagons for BS-1 and BS-2, respectively.

V. Analysis of Urban mmWave Coverage with Ray-Tracing Simulations

Before we present our trajectory optimization results, in this section, we first provide coverage analysis for drones at different altitudes in an urban scenario. We consider a dense urban region in Manhattan with two active BSs, as shown in Fig. 5. The BSs were equipped with three 8×8 antenna arrays operating at mmWave frequencies. Relevant simulation parameters are listed in Table III. The drone corridor was assumed to be enclosed within a cuboid extending from 50 m to 100 m in altitude, and of dimension 1240 m by 1050 m along the x-y plane. This cuboid was sampled at discrete intervals of space to generate a 3D grid of receiver locations. Then, ray-tracing simulations were used to obtain the channel matrix in (5) between each antenna array element of each BS to each receiver location. These channel matrices were processed offline to calculate the RSS at each receiver location after maximum ratio transmit (MRT) digital beamforming. The entire raytracing dataset is shared at [34].

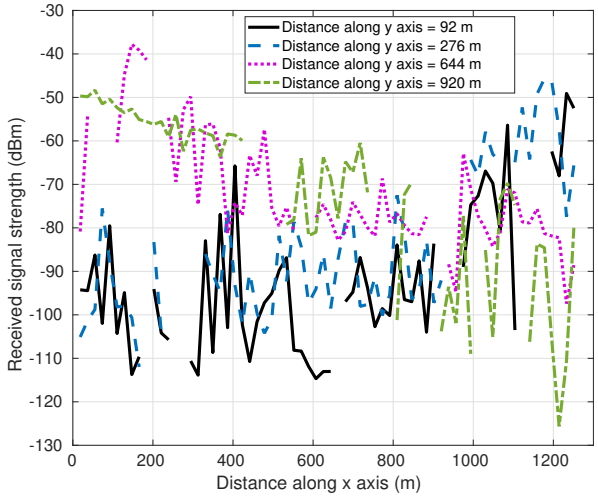
TABLE III: Raytracing simulation parameters.

Parameter	**Value**
Center frequency (f_c)	28 GHz
mmWave bandwidth (B)	1 GHz
BS-1 location	Latitude: 40.771306, Longitude: −73.965167, Height: 63.4 m
BS-2 location	Latitude 40.766250, Longitude: −73.953444, Height: 42.4 m
BS antenna array	8×8
Receiver grid dimensions	1240 m by 1050 m
Adjacent receiver spacing	18.4 m
Receiver altitudes	$[50, 75, 100]$ m

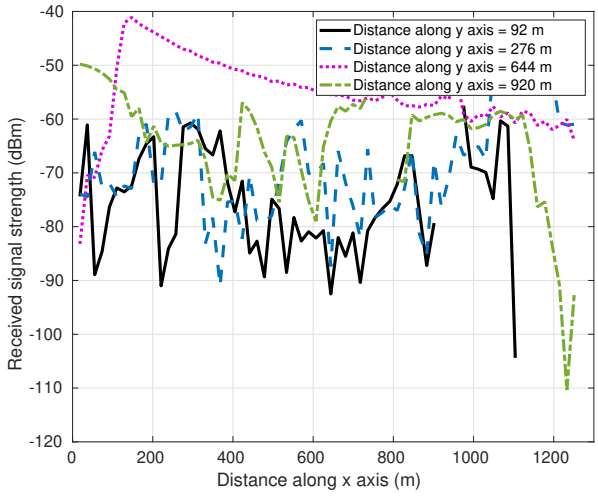
The resultant RSS heatmap at various altitudes is displayed in Fig. 6. The association of locations with BSs is also shown, locations which have various intensities of green and blue are associated with BS-1 while those with various intensities of orange are associated with BS-2. Compared to the RSS at altitudes of 75 m (Fig. 6b) and 100 m (Fig. 6c), the RSS at an altitude of 50 m (Fig. 6a) has more outages, caused by buildings taller than 50 m. Drone corridor lanes can bypass such regions of RSS outage by either bending around them, at the same altitude, or bending over them at higher altitudes where the RSS is higher, or below them at lower altitudes. Such changes in altitude result in an angular change in the trajectory, and the best action is determined by the angular change constraint, the RSS constraint, and the receiver location.

Fig. 7 shows the received power levels at altitudes of 50 m and 100 m, at receiver locations parallel to the x axis, at various distances along the y axis. The corresponding paths are also shown in the heatmaps in Fig. 6, in matching colors and line styles. Solid black curves represent a distance of 72 m along the y-axis towards north, dotted blue curves represent a distance of 276 m along y axis, dashed magenta curves represent a distance of 644 m along y axis, and dash-dot green curves represent a distance of 920 m along the y axis. Individual RSS levels were calculated at altitudes of 50 m, 75 m, and 100 m. Then, the maximum RSS across all altitudes was calculated. Fig. 7a shows the RSS at these receiver locations at an altitude of 50 m, and Fig. 7b at 100 m. It can be seen that there are more discontinues in RSS at an altitude of 50 m compared to 100 m, due to the presence of buildings in the line of sight path. It can also be seen that the RSS at a distance of 920 m along y-axis is better in some locations at an altitude of 50 m than at 100 m, since the path loss is less at lower altitudes. The altitude at which the RSS is maximized is shown in Fig. 8, demonstrating the number of altitude transitions required if the goal is simply to maximize RSS, ignoring all other constraints.

It can also be seen from Fig. 7a that there are some receiver



(a)



(b)

Fig. 7: RSS along the x-axis at various distances along y axis. The plot at altitude of 50 m is shown in (a), at an altitude of 100 m in (b).

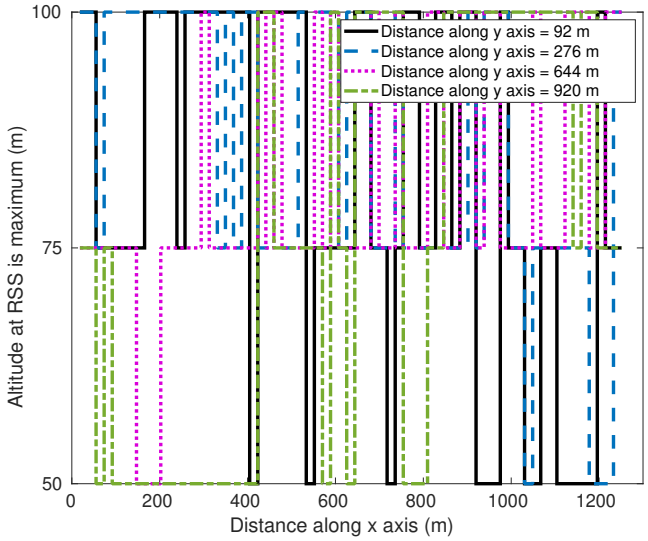


Fig. 8: Altitude at which RSS is maximum, for receiver locations parallel to the x axis, at various distances along the y axis.

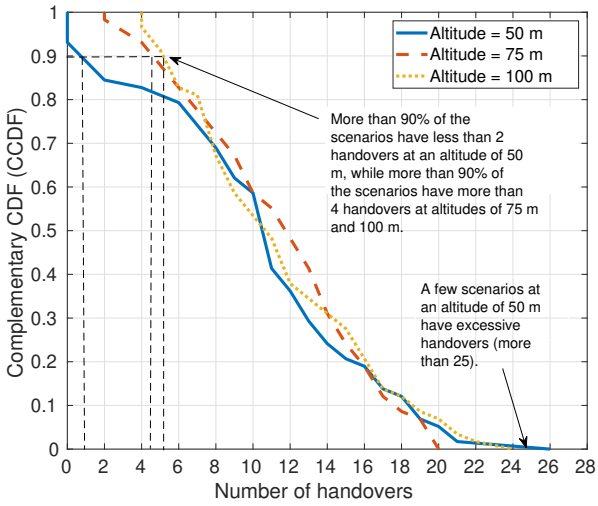


Fig. 9: The complementary CDF (CCDF) of the number of handovers along straight line paths parallel to the x-axis, at altitudes of 50 m, 75 m, and 100 m. The paths are at intervals of 18.4 m along the entire y axis.

locations that are associated with distant BS antenna arrays, and the number of such locations increases with altitude. This behaviour is caused due to the fact that such locations lie in the coverage holes of the closest BS antenna arrays, due to the presence of obstructions in the dense urban environment, and the RSS from the farther BS is still high at higher altitudes. Such locations can cause frequent handovers if they are present with the trajectory of a drone corridor lane. Hence, while higher altitudes may be desirable due to higher RSS, the caveat is that they may cause frequent handovers.

Fig. 9 shows the complementary CDF (CCDF) of the number of handovers at altitudes of 50 m, 75 m, and 100 m, where the distribution was calculated along paths parallel to the x axis, at intervals of $\delta_{\mathrm{y}} = 18.4$ m along the y axis. A handover is assumed when the strongest RSS at a given location, from the six sectors, changes. In practice, the number of handovers can be reduced by considering hysteresis and time to trigger. 90% of the paths require fewer than two handovers at an altitude of 50 m, while at altitudes of 75 m, and 100 m, 90% of

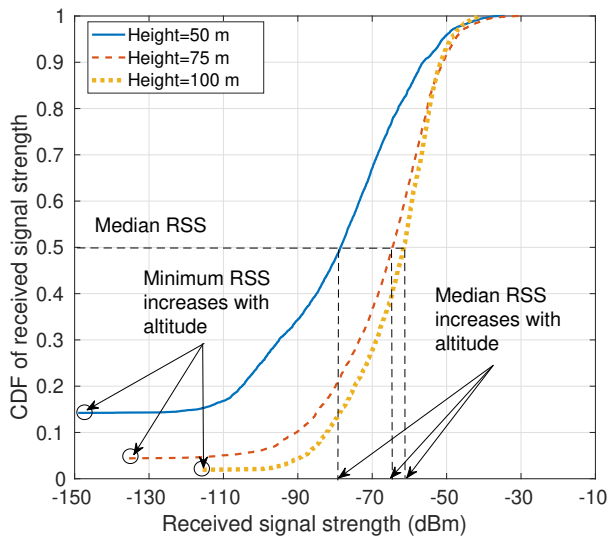


Fig. 10: The CDF of the best RSS, calculated over the receiver grid at altitudes of 50 m, 75 m, and 100 m

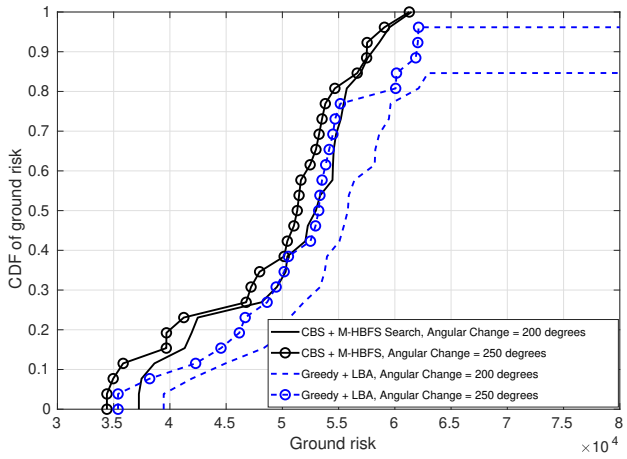


Fig. 11: CDF of the ground risk achieved by the optimal and non-optimal algorithms, calculated over multiple start and end waypoints.

TABLE IV: Drone corridor specifications.

Parameter	**Value**
Number of lanes (N_{L})	3
Start waypoint of the lanes	(9.2, 285.2, 75) m, (9.2, 303.6, 75) m, and (9.2, 322, 75) m
End waypoint of the lanes	(266.8, 285.2, 75) m, (266.8, 303.6, 50) m, and (266.8, 322, 100) m
Number of scenarios	26. Each start and end waypoint is incremented by 18.4 m along the y-axis to create a new scenario, for the same x and z coordinates.
RSS constraint (S_{Thr})	-120 dBm
Maximum angular change ($\theta_{\max}$)	200° or 250°
Elasticity (ϵ)	75 m
Number of hops (N_{WP})	2

the paths require more than four handovers. Also, a few paths at an altitude of 50 m require more than 25 handovers, which is larger than that needed by any path at altitudes of 75 m and 100 m. This signifies that, in some regions at an altitude of 50 m, the RSS from the closer BS fluctuates rapidly with distance, due to the presence of more obstacles in the line of sight path at lower altitudes. This fluctuation in the RSS causes handovers. The cumulative distribution function (CDF) of received power along the x-y grid at different altitudes is shown in Fig. 10. Both the median and minimum RSS increase with altitude, due to fewer obstructions in the line of sight path.

VI. Drone Corridor Results

In this section, we discuss the performance of the proposed algorithms in minimizing ground risk for drone corridors designed within the environment described in Section V.

The CDF of the ground risk in the corridor achieved by the algorithms is shown in Fig. 11, for the drone corridor specifications listed in Table IV. The ground risk was calculated as the sum of integrated ground risk over each lane. The CDF was calculated over a number of scenarios with varying start and end lane waypoints. These start and end waypoints were chosen to be close to each other to increase the probability of inter-lane conflicts and thus, to ensure that the ability of higher level search to resolve these conflicts is tested. The ground risk achieved by the combination of greedy and LBA is 5.77% higher than that achieved by CBS and M-HBFS at an angular change constraint of 200°, and 5.43% higher at an angular change constraint of 250°, i.e. the heuristic approach performs close to the optimal. However, the heuristic approach can find a solution in only 84.6% of the scenarios at an angular change constraint of 200°. When the angular change constraint is relaxed to 250°, the backtracking in LBA has a higher probability of finding a valid path, and thus, the heuristic approach can solve 96.15% of the scenarios.

Insights into the behaviour of the algorithms can be obtained from Fig. 12. The top view of the paths of the three lanes, overlaid on the ground risk map, is shown in Fig. 12. The parameters for this scenario are the same as listed in Table IV, however the start waypoints for the three lanes are $(9.2, 395.6, 75)$ m, $(9.2, 414, 75)$ m, and $(9.2, 432.4, 75)$ m and the end waypoints (1002.8, 395.6, 75) m, (1002.8, 414, 50) m, and (1002.8, 432.4, 100) m. Hence, the lanes have a longer span along the x axis, close to the entire width of the simulation area. The paths found by the CBS and M-HBFS algorithm are represented by the solid line, and those by the greedy and LBA as the dashed line. Magenta curves are for lane-1, black curves for lane-2, and blue curves for lane-3. The CBS algorithm adjusts all lanes, unlike the greedy algorithm which only adjusts the current lane, keeping all previous lanes stationary. This behaviour can be seen from Fig. 12 wherein the trajectory of lane-1 found by the CBS algorithm actually follows a higher ground risk path, of cost 70500.1, than its

TABLE V: Mean and standard deviation of the run time of ground risk minimization algorithms.

Algorithm	Maximum angular change	Mean run time	Standard deviation of run time
CBS + M-HBFS	200°	68.35 min	49.2 min
Greedy + LBA	200°	35.02 min	5.3 min
CBS + M-HBFS	250°	358.99 min	118 min
Greedy + LBA	250°	24.36 min	4.48 min

individually optimal path, of cost 65054.3, as in the case of the greedy algorithm. This re-adjustment creates space for lane-2 and lane-3 to follow lower ground risk paths, of costs 74975.6 and 78887.8 respectively. The paths of lane-2 and lane-3, as found by the greedy algorithm, have higher costs of 85959 and 81531.7 respectively. This approach of CBS results in a lower overall ground risk of 224363.5, when compared to greedy, which achieves a ground risk of 232545, i.e. the CBS algorithm incurs a higher local cost to achieve a lower global cost.

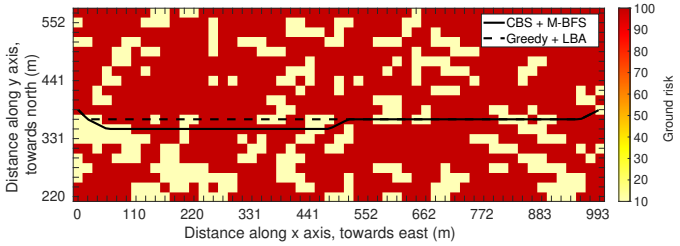
The CDF of run time of the two algorithms is shown in Fig. 13, and the mean and standard deviation values are listed in Table V. The CDF was calculated for the scenarios listed in Table IV, with CBS nodes evaluated using parallel processes. Greedy algorithm performs low level search, by calling LBA, a maximum of N_{L} times, whereas the number of times CBS performs low level search varies and can be more than N_{L}, depending on the number of conflicts between lanes. In the worst case, LBA evaluates all candidate waypoints once, while M-HBFS has to evaluate all possible paths to the goal. Hence, the combination of CBS and M-HBFS has a higher variance and mean run time compared to the combination of greedy and LBA. Further, while relaxing the angular change constraint reduces the mean run time of greedy and LBA (due to less backtracking for LBA), it increases run time of CBS and M-HBFS (since there are now more valid paths for M-HBFS to evaluate). Since CBS calculates the trajectories of N_{L} lanes in parallel, while greedy calculates the lane trajectories sequentially, in scenarios without any conflicts between the individually optimal lane trajectories, CBS and M-HBFS is actually faster than greedy and LBA.

Fig. 14 shows the variation in achievable data rate within the drone corridor for different UAV traffic densities. Since, the achievable uplink data rate at each location within the corridor is a random variable, the 95th percentile of this data rate is calculated, and then CDF is plotted across all locations within the corridor. It can be seen that the achieved data rate reduces with increase in UAV traffic density, since each BS has to serve a larger number of UAVs.

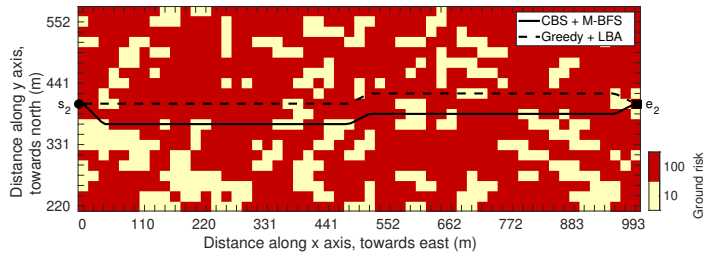
VII. Concluding Remarks

In this work, we modelled the problem of minimizing ground risk drone corridors subject to constraints on the RSS along the lanes of the corridor and on the integrated angular change of each lane. We proposed a heuristic approach and compared its performance with the optimal. It was found that the ground risk cost achieved by a modified version of A* algorithm was 5.77% higher than the optimal, for the considered scenarios. We also presented insights on the wireless coverage and achievable data rates in drone corridors when using ground BSs operating at mmWave frequencies.

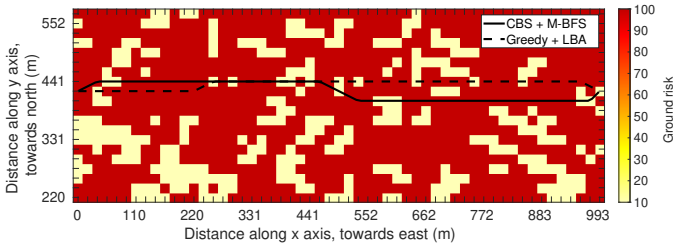
In this work, the drone corridor trajectories are computed offline, and UAVs follow these pre-computed trajectories. A possible extension is to allow UAVs to choose their next waypoint in real-time, while avoiding collisions with other UAVs, within the same and other lanes, using detect and avoid (DAA) techniques.



(a) Path of lane-1



(b) Path of lane-2



(c) Path of lane-3

Fig. 12: Paths of the three drone corridor lanes, as found by the combination of CBS and M-HBFS, and Greedy and LBA algorithms

References

[1] Building the drone highways of the sky. Accessed on 2020, June. [Online]. Available: https://www.smartcitiesdive.com/news/building-drone-highways-smart-city-sky/592581/

[2] How Will Drone Corridors Revolutionise Industrial Estates? Accessed on 2022, Jan. [Online]. Available: https://www.lexology.com/library/detail.aspx?g=6b350655-83ac-4af0-a8a5-8613af248dfd

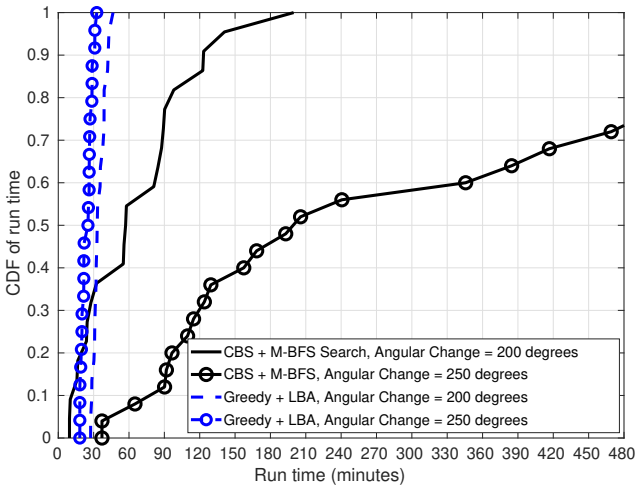


Fig. 13: CDF of the run time of the algorithms, calculated over the scenarios listed in Table IV.

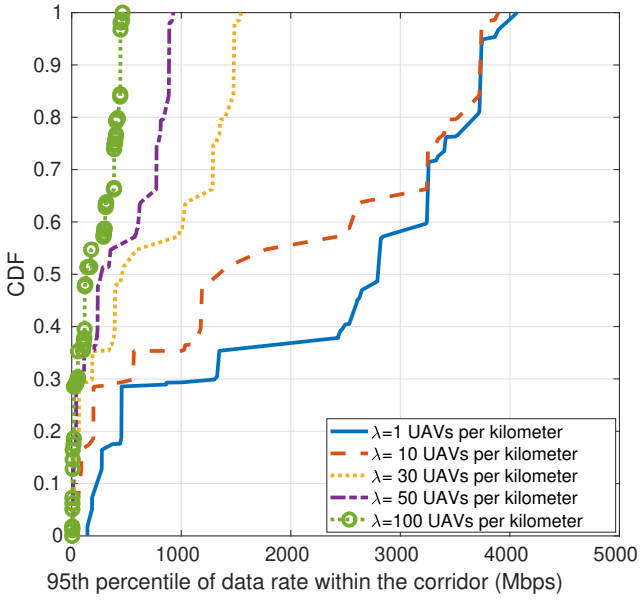


Fig. 14: CDF of the 95th percentile of achievable data rate at each location within the drone corridor, for various UAV traffic densities.

[3] What is Unmanned Aircraft Systems Traffic Management? Accessed on 2020, June. [Online]. Available: https://www.nasa.gov/ames/utm

[4] UAS Data Exchange (LAANC). Accessed on 2020, June. [Online]. Available: https://www.faa.gov/uas/programs_partnerships/data_exchange/

[5] Gov. Whitmer announces air mobility corridor development in Michigan and Ontario. (2022, Jan 10). [Online]. Available: https://www.michigan.gov/mdot/0,4616,7-151--575027--,00.html

[6] NBEC - UK National BVLOS Experimentation Center. Accessed on 2022, Jan. [Online]. Available: http://uknbec.com/

[7] Drone flies out of sight and back again along 16km 'corridor' over land. Accessed on 2022, Jan. [Online]. Available: https://www.imeche.org/news/news-article/drone-flies-out-of-sight-and-back-again-along-16km-corridor-over-land

[8] The impact of drones on the UK economy. Accessed on 2022, Jan. [Online]. Available: https://www.pwc.co.uk/issues/intelligent-digital/the-impact-of-drones-on-the-uk-economy.html

[9] NUAIR and New York UAS Test Site Unlock 35 Miles of Beyond Visual Line of Sight Drone Operations. Accessed on 2022, Jan. [Online]. Available: https://nuair.org/2022/01/04/nuair-nyuasts-unlock-35-miles-of-bvlos-airspace/

[10] FAA Statements on 5G. Accessed on 2022, Jan 13. [Online]. Available: https://www.faa.gov/newsroom/faa-statements-5g

[11] NY to Host First in Nation 5G UAS Test Range. Accessed on 2022, Jan 13. [Online]. Available: http://uasmagazine.com/articles/2371/ny-to-host-first-in-nation-5g-uas-test-range

[12] Using GIS to Build Highways in the Skies. Accessed on 2022, Jan 13. [Online]. Available: https://www.esri.com/en-us/lg/industry/transportation/airspace-link-case-study

[13] M. Samir, S. Sharafeddine, C. M. Assi, T. M. Nguyen, and A. Ghrayeb, "UAV trajectory planning for data collection from time-constrained IoT devices," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 34–46, 2019.

[14] A. V. Savkin, H. Huang, and W. Ni, "Securing UAV communication in the presence of stationary or mobile eavesdroppers via online 3D trajectory planning," *IEEE Wireless Commun. Lett.*, vol. 9, no. 8, pp. 1211–1215, 2020.

[15] Z. Na, Y. Liu, J. Shi, C. Liu, and Z. Gao, "UAV-supported clustered NOMA for 6G-enabled Internet of Things: Trajectory planning and resource allocation," *IEEE Internet of Things Journal*, vol. 8, no. 20, pp. 15 041–15 048, 2021.

[16] K. Zhu, X. Xu, and S. Han, "Energy-efficient UAV trajectory planning for data collection and computation in mMTC networks," in *Proc. IEEE Globecom Workshops*, 2018, pp. 1–6.

[17] M. M. U. Chowdhury, S. J. Maeng, E. Bulut, and I. Güvenç, "3-D Trajectory Optimization in UAV-Assisted Cellular Networks Considering Antenna Radiation Pattern and Backhaul Constraint," *IEEE Trans. Aerospace and Electron. Syst.*, vol. 56, no. 5, pp. 3735–3750, 2020.

[18] C. Zhao, J. Liu, M. Sheng, W. Teng, Y. Zheng, and J. Li, "Multi-UAV Trajectory Planning for Energy-efficient Content Coverage: A Decentralized Learning-Based Approach," *IEEE Select. Areas in Commun.*, vol. 39, no. 10, pp. 3193–3207, 2021.

[19] J. Hu, H. Zhang, L. Song, R. Schober, and H. V. Poor, "Cooperative internet of UAVs: Distributed trajectory design by multi-agent deep reinforcement learning," *IEEE Trans. Commun.*, vol. 68, no. 11, pp. 6807–6821, 2020.

[20] Q. Wu, Y. Zeng, and R. Zhang, "Joint trajectory and communication design for multi-UAV enabled wireless networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 2109–2121, 2018.

[21] X. Liu, Y. Liu, Y. Chen, and L. Hanzo, "Trajectory design and power control for multi-UAV assisted wireless networks: A machine learning approach," *IEEE Trans. Vehic. Technol.*, vol. 68, no. 8, pp. 7957–7969, 2019.

[22] W. Feng, N. Zhao, S. Ao, J. Tang, X. Zhang, Y. Fu, D. K. C. So, and K.-K. Wong, "Joint 3D Trajectory and Power Optimization for UAV-aided mmWave MIMO-NOMA Networks," *IEEE Transactions on Communications*, vol. 69, no. 4, pp. 2346–2358, 2020.

[23] P. Susarla, Y. Deng, G. Destino, J. Saloranta, T. Mahmoodi, M. Juntti, and O. Sílven, "Learning-based trajectory optimization for 5G mmWave uplink UAVs," in *Proc. IEEE Int. Commun. Workshops (ICC Workshops)*, 2020, pp. 1–7.

[24] L. Shi, S. Xu, H. Liu, and Z. Zhan, "QoS-Aware UAV Coverage path

planning in 5G mmWave network," *Elsevier, Computer Netw.*, vol. 175, p. 107207, 2020.

[25] S. Bertrand, N. Raballand, and F. Viguier, "Evaluating ground risk for road networks induced by UAV operations," in *Proc. IEEE Int. Conf. Unmanned Aircraft Syst., Dallas, TX*, 2018, pp. 168–176.

[26] H. Binol, E. Bulut, K. Akkaya, and I. Guvenc, "Time optimal multi-uav path planning for gathering its data from roadside units," in *2018 IEEE Vehic. Technol. Conf. (VTC-Fall)*, 2018, pp. 1–5.

[27] Z. Qadir, F. Ullah, H. S. Munawar, and F. Al-Turjman, "Addressing disasters in smart cities through UAVs path planning and 5G communications: A systematic review," *Elsevier, Computer Communications*, vol. 168, pp. 114–135, 2021.

[28] Y. Zeng, Q. Wu, and R. Zhang, "Accessing from the sky: A tutorial on UAV communications for 5G and beyond," *Proc. of the IEEE*, vol. 107, no. 12, pp. 2327–2375, 2019.

[29] B. Li, Z. Fei, and Y. Zhang, "UAV communications for 5G and beyond: Recent advances and future trends," *IEEE Internet of Things J.*, vol. 6, no. 2, pp. 2241–2263, 2018.

[30] B. Levasseur, S. Bertrand, and N. Raballand, "Efficient generation of ground impact probability maps by neural networks for risk analysis of UAV missions," in *Proc. IEEE Int. Conf. Unmanned Aircraft Syst. (ICUAS)*, 2020, pp. 1398–1406.

[31] S. Singh, U. Bhattacherjee, E. Ozturk, İ. Güvenç, H. Dai, M. Sichitiu, and A. Bhuyan, "Placement of mmWave base stations for serving urban drone corridors," in *Proc. IEEE Vehic. Technol. Conf.*, 2021, pp. 1–6.

[32] R. Stern, N. R. Sturtevant, A. Felner, S. Koenig, H. Ma, T. T. Walker, J. Li, D. Atzmon, L. Cohen, T. S. Kumar *et al.*, "Multi-agent pathfinding: Definitions, variants, and benchmarks," in *Proc. Annual Symp. Combinatorial Search*, 2019.

[33] G. Sharon, R. Stern, A. Felner, and N. R. Sturtevant, "Conflict-based search for optimal multi-agent pathfinding," *Artificial Intelligence*, vol. 219, pp. 40–66, 2015.

[34] Raytracing dataset for drone corridors. [Online]. Available: https://github.com/ssinghjah/DroneCorridorDataset