Risky zone avoidance strategies for drones

Michel Barbeau, Joaquin Garcia-alfaro, Evangelos Kranakis

Risky Zone Avoidance Strategies for Drones

Michel Barbeau
Carleton University, Canada
0000-0003-3531-4926

Joaquin Garcia-Alfaro
Telecom SudParis, France
0000-0002-7453-4393

Evangelos Kranakis
Carleton University, Canada
0000-0002-8959-4428

***Abstract*—We consider the problem of a drone having to traverse a terrain. Traversal of the terrain exposes the drone to certain risks, e.g., concentration of chemicals, severe thunderstorm wind gusts or any disturbing weather phenomenon. The goal of the drone is to navigate the terrain while minimizing the amount of risk. We develop a framework for quantifying the exposure to risk factors in a circular zone model. We propose risky zone avoidance navigation strategies for rectilinear or curvilinear drone trajectories. We validate the work using numeric simulations.**

***Index Terms*—Drone, collision avoidance, obstacle avoidance, zone avoidance, risk evaluation, risk mitigation.**

I. Introduction

Communications with drones Beyond Visual Line of Sight (BVLOS) are expected to be one of the important applications of 5G networks [1], [2]. With BVLOS, pilots maintain communications with drones across arbitrarily large distances over cellular networks. It is a significant technological advancement. An enabler for drone applications requiring wide geographical area coverage such as the delivery of parcels or medical supplies. However, this new technology has its own challenges. Among others, safety is essential. Several incidents can be prevented by taking into account the existence of risky zones and avoiding to fly over them as much as possible. This includes collisions with flying birds or other flying objects. Thus, drones may avoid traversing zones with concentrations of wild bird populations or heavy air traffic.

In this paper, we address the following problem. Consider a drone having to fly over a given terrain. While traversing the terrain, the drone is exposed to risks related to the geographical locations of factors. For example, a relative risk can be estimated as a function of the 2D density of a population of individuals or as a function of the 3D concentration of chemicals or disturbing weather phenomena. The goal is to navigate the terrain so as to find a path of minimum risk.

We develop a metric for quantifying the exposure of a drone to risk factors in a circular zone model. We design new zone avoidance strategies for drones traversing terrains comprising several risky zones while following either a rectilinear (i.e., straight-line) or a curvilinear (i.e., non straight-line) trajectory.

Related work is briefly reviewed in Section II. In Section III, we describe details of a metric for quantifying the exposure to a risk factor suiting a circular zone model. Risky zone avoidance techniques for rectilinear trajectories are presented in Section IV, while they are defined for curvilinear strategies in Section V. Simulation work is presented in Section VI. We conclude with Section VII.

II. Related Work

The need for zone avoidance emerges from the requirement to avoid collisions or obstacles. Collision avoidance algorithms tend to focus on reaction rather than prevention. Low latency reaction to events observed on cameras or detected with sensors is of paramount importance [3], [4]. In contrast to our concept of zone, obstacles are solid structures that cannot be traversed. We assume, however, structures that are not necessarily bounded by rigid compounds They can be traversed by drones, but with a certain risk. Traditional obstacle avoidance techniques can be adapted to control the way how drones can prevent exposure to such intangible areas. We adapt strategies from the field of robotics [5], e.g., potential-function-based control [6], [7], curve following [8] strategies and bug algorithms [9]. There are also studies (some of which are very recent) on various collision and obstacle avoidance approaches for drones [10]–[13]. In all these situations, one assumes the presence of a physical boundary, like wall or fence, which is to be avoided. Our work is complementary to the research on collision and obstacle avoidance. In contrast to obstacles, zones are also relatively large geographical areas. Our goal is to minimize risk and travel time through risky zones, if they cannot be avoided entirely.

III. Measuring Risk Exposure

We define a new metric for measuring the amount of risk resulting from a selected path. This leads to methods for navigating a terrain. We develop strategies for navigating a terrain comprising risky zones.

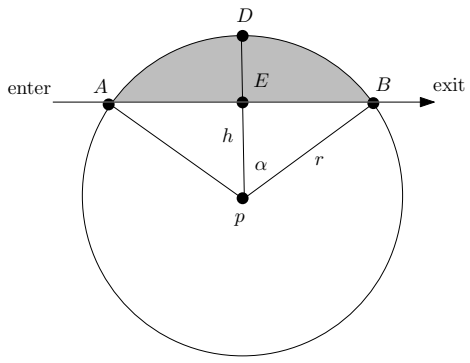


Fig. 1. A drone is traversing a circular zone with a rectilinear trajectory from A (entrance) to B (exit) at distance h from the source point p. The shaded area represents the total amount of exposure to risk by the drone.

A. Circular Zone Model

As depicted in Figure 1, a zone is determined by a source point p in the plane together with a radius r, greater than zero. The value r is called the *sensitivity distance*. Its value depends on the intensity of the source causing the risk factor. The larger the r, the farther the range of the source. A drone traversing a zone is exposed to some risk. The risk level depends on the distance between the drone and source point. Beyond distance r, the source represents no risk. The sensitivity distance is used to define a risk evaluation metric.

B. Risk Evaluation Metric

When traversing a zone, it is assumed that the closer a drone is to the source point p, the greater is the exposure to risk. Let us assume that when a drone is at distance x from a source point, with $x \leq r$, it gets gets an amount of risk exposure defined by the function $V(r, x)$. There are several possibilities for choosing the function V. For example, in the simple linear model we employ in our present work, we may assume that $V(r, x) = r - x$ units. This leads to assume that the total amount of the risk incurred by a drone during its entire trajectory within p's range from A to B is measured by the area of the circular segment $AEBD$ (represented by the shaded domain in Figure 1).

The following lemma formally captures this idea and provides a metric for measuring the risk.

Lemma 1. *Assume a drone is traversing a rectilinear trajectory at distance h from a source point p with sensitivity distance r, where $r \geq h \geq 0$. The total amount $V(r, h)$ of risk exposure that the drone incurs for its entire trajectory within the zone is equal to*

$$V(r, h) = \arccos\left(\frac{h}{r}\right) \cdot r^2 - h \cdot \sqrt{r^2 - h^2}. \tag{1}$$

Proof. Referring to Figure 1, a drone is traversing a rectilinear trajectory whose distance h from the source point is p; the drone is entering the circle at A and exiting it at B. We are interested to measure the risk exposure for the duration of the entire trajectory from A to B. We calculate the total risk exposure of a drone traversing a zone on a rectilinear trajectory from A to B. Let pDB denote the circular sector and $\alpha = \angle(BpD)$. Note that

$$Area(AEBD) = 2 \cdot (Area(Sector(pDB)) - Area(\triangle(BpE)) \tag{2}$$

Since $\cos\alpha = \frac{h}{r}$. we see that

$$Area(Sector(pDB)) = \frac{\alpha}{2} \cdot r^2 = \frac{\arccos\left(\frac{h}{r}\right)}{2} \cdot r^2 \tag{3}$$

and

$$\begin{aligned} Area(\triangle(BpE)) &= \frac{1}{2} \cdot h \cdot r \cdot \sin\alpha \\ &= \frac{1}{2} \cdot h \cdot r \cdot \sqrt{1 - \left(\frac{h}{r}\right)^2} \\ &= \frac{1}{2} \cdot h \cdot \sqrt{r^2 - h^2} \end{aligned} \tag{4}$$

Using Equations (2), (3), (4) we conclude that

$$Area(AEBD) = \arccos\left(\frac{h}{r}\right) \cdot r^2 - h \cdot \sqrt{r^2 - h^2},$$

which proves the lemma. □

C. Risk Associated with Multiple Zones Traversal

When a drone is traversing a terrain within which k source points $p_1, p_2, \ldots, p_k$ reside with respective sensitivity distances $r_1, r_2, \ldots, r_k$, the total risk exposure accumulated by a drone will be equal to

$$\sum_{i=1}^{k} V(r_i, h_i), \tag{5}$$

where h_i is the distance of the trajectory of the drone from source point p_i, for $i = 1, \ldots, k$, respectively. Figure 2 depicts

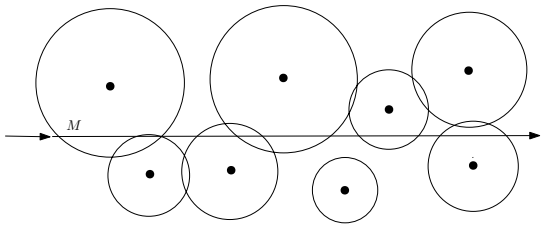


Fig. 2. A drone M is traversing a terrain following a rectilinear trajectory. Its exposure during the trajectory is the sum of the exposures to every source.

the rectilinear trajectory of a drone M in a given terrain and its total risk exposure from sources in its proximity. The latter is measured as the sum of the risks for all the sources encountered, given by Equation (5).

IV. Strategies for Rectilinear Trajectories

We consider the problem of navigating a terrain while at the same time minimizing the amount of risk that a flying drone is exposed to, when following a rectilinear trajectory. The main problem of interest can be specified as follows:

> Determine the rectilinear trajectory a drone should follow so as to minimize the risk when traversing a rectangular terrain.

More precisely, consider a rectangular terrain represented by two parallel lines at distance one from each other. Between the two lines there are source points of some risk. In the first type of trajectories considered, we assume the drone keeps a given fixed direction (see Figure 3) and accumulates the minimum amount of risk for this direction, For example, the trajectory depicted in Figure 3 represents a straight line that is parallel to the two sides of the rectangular area.

We use Lemma 1 to determine the trajectory of drone M which ensures minimal risk exposure for the passage of a drone in two situations: first when the direction of movement of the drone is known, and second when the starting point of the drone is known. To be more specific, we establish the following statement.

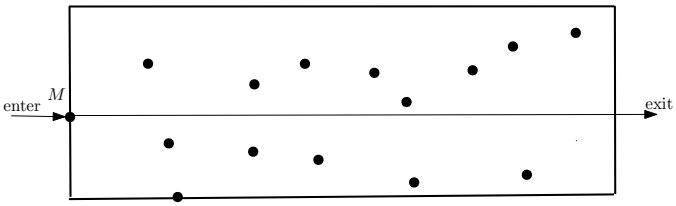


Fig. 3. A rectangular terrain delimited on its length by two parallel lines at distance one from each other. The rectilinear trajectory of the drone M is assumed to be parallel to the two parallel lines along the length of the terrain.

Theorem 2 (Exposure in Directional Trajectory)**.** *Delimited along its length by two parallel lines, consider a rectangular terrain enclosing k source points $p_1, p_2, \ldots, p_k$, of respective sensitivity distances $r_1, r_2, \ldots, r_k$. When moving in parallel to the delimiting lines, for any choice of entry point, in time $\mathcal{O}(k)$ we can compute a trajectory for the drone that minimizes its total risk exposure.*

Proof. Without loss of generality, we assume that the trajectory of the drone is parallel to the length of rectangular area, see Figure 3. The proof is similar for any choice of entry point.

Consider the scenario depicted in Figure 4 with k source points $p_1, p_2, \ldots, p_k$ placed within a rectangular area delimited by two parallel lines.

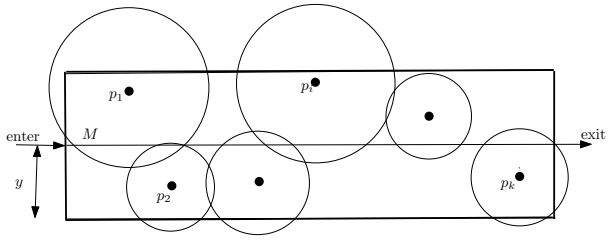


Fig. 4. A drone M is traversing a rectangular terrain following rectilinear trajectory, at distance y from the bottom delimiting line.

We want to find a rectilinear trajectory parallel to the terrain frontiers that minimizes the drone's risk. Without loss of generality, let us assume that the width (vertical distance of the two parallel horizontal delimiting lines) of the area is 1, and let the bottom horizontal line be the line at $y = 0$ (x-axis) and top horizontal line be the line at $y = 1$. Let us assume that a potential drone trajectory is along a horizontal line where $0 \leq y \leq 1$. The problem reduces to finding the y that minimizes the total risk of a drone that traverses the terrain following the trajectory determined by y.

Assume a horizontal line is vertically sliding, with range 0 to 1. We consider events that occur as the line intersects various circular zones (see Figure 5). Consider a circular zone with the source point p and sensitivity distance r. As y is increasing, the line scans the zone. It first intersects the circle at the tangent point p_b, then passes through its center p, and finally intersects the circle at the tangent point p_a.

Consider scanning with a horizontal line crossing the disk centered at p between the tangent points p_b and p_a on its perimeter. Recall that h is the distance of the horizontal line

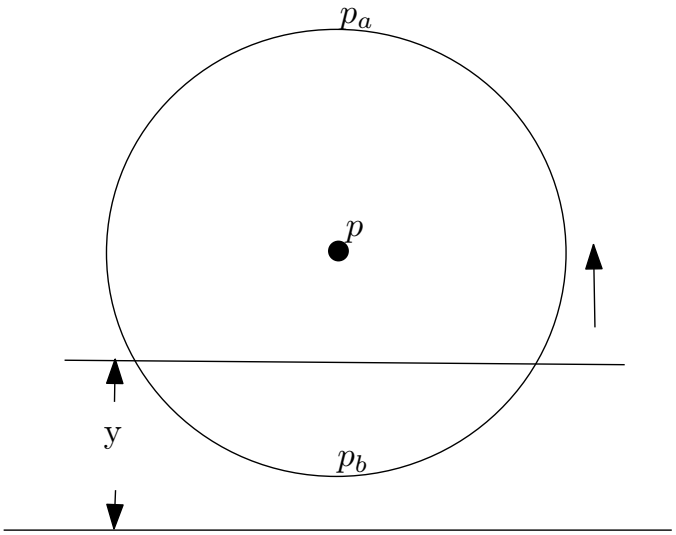


Fig. 5. For any source point p consider the circle with center p and sensitivity r. As y is increasing, the line scans the circle area. It first intersects the circle at the tangent point p_b, then passes through the center p, and finally intersects the circle at the tangent point p_a.

to the center p. The evaluation of function $V(r, h)$ starts at 0 with $y = p_b$ and $h = r$. It then keeps increasing; until the horizontal line passes through the center p of the circular zone ($y = p$). As the line is moving towards the tangent point p_a ($y = p_a$), the evaluation of function $V(r, h)$ keeps decreasing and assumes again the value 0 when the line passes through the point p_a, in which case $y = p_a$ and $h = r$. It is clear that the function $V(r, h)$ is maximized when $y = p$.

Consider a trajectory for the drone M that is parallel to the side length of the terrain and at distance y from the bottom line. Let $I, J \subseteq \{1, 2, \ldots, k\}$ be the subsets of indices of nodes such that

- for each $i \in I$ the line y intersects the circle centered at p_i, below p_i, and
- for each $j \in J$ the line y intersects the circle centered at p_j, above p_j.

The drone accumulates a total risk amount for its entire trajectory equal to

$$\sum_{i \in I} V(r_i, |y - p_i|) + \sum_{j \in J} V(r_j, |y - p_j|), \tag{6}$$

where r_i, r_j denote the respective ranges of the circular zones centered at p_i, p_j where $i \in I$, and $j \in J$, and $V(r, h) := \arccos\left(\frac{h}{r}\right) \cdot r^2 - h \cdot \sqrt{r^2 - h^2}$ is the function in Equation (1) that measures the amount of risk of the drone M. While the configuration of sets of indices I, J remains unchanged during the scanning of the line, the first summand in Equation (6) is monotone increasing in y. The second summand in Equation (6) is monotone decreasing in y. Therefore while the configuration I, J remains the same, we can compute the minimal value of the sum in Equation (6). A change occurs when an event occurs as depicted in Figure 5 in which case the configuration of sets I, J changes.

Our strategy is described in more detail in Algorithm 1.

Algorithm 1 Rectilinear traversal (Source points: $p_1, \ldots, p_k$, Sensitivity distances: $r_1, \ldots, r_k$)

1: Start with the line $y = 0$, where $0 \leq y \leq 1$, from bottom $y = 0$ to top $y = 1$;
2: For each event encountered with line y, determine the risk amount using Equation (6);
3: Determine the next event;
4: Compute the minimum risk amount given by Equation (6) for events between $y = 0$ and $y = 1$;
5: Return minimum risk exposure;

It follows that as we run through our event space, for each pair (I, J) of sets of indices we can determine easily the value of y that minimizes the sum. Since there are $\mathcal{O}(k)$ events there are $\mathcal{O}(k)$ such pairs (I, J). Therefore, we can compute the optimal trajectory of the drone in $\mathcal{O}(k)$ time. This completes the proof of the theorem. □

Although we do not know how to compute the overall optimal rectilinear trajectory, we can approximate it as close to the optimal as required.

Corollary 3 (Approximate Exposure Trajectory)**.** *Consider a rectangular terrain consisting of two delimiting parallel lines and k source points $p_1, p_2, \ldots, p_k$, of respective sensitivities $r_1, r_2, \ldots, r_k$. For any integer $n \geq 1$, in time $\mathcal{O}(nk)$ we can compute a trajectory for a drone whose total risk exposure V is such that $|V_{opt} - V| \leq A/n$, where V_{opt} is the optimal exposure and A is the area of the terrain.*

Proof. Divide the plane into n directions $i\pi/n$ spaced at an angle of π/n radians, where n is sufficiently large. For each $i = 0, 1, \ldots, n-1$ we can run the directional algorithm in this direction and compute the minimum value, say V_i. Let $V_{\min}$ be defined as $\min_{0 \leq I < n} V_i$. The computation of each V_i can be done in $\mathcal{O}(k)$ time and therefore the computation of $V_{\min}$ in $\mathcal{O}(nk)$ time.

Now, consider a rectilinear trajectory that attains the optimal exposure, say V_{opt}. Clearly, there is a direction i such that the $V_i \leq V_{opt} + A/n$, where A is the area of the terrain traversed by the drone. It follows that $V_{\min} \leq V_i \leq V_{opt} + A/n$. Therefore $V_{\min} - A/n \leq V_{opt} \leq V_{\min}$, where $V_{opt} \leq V_{\min}$ follows from the fact that V_{opt} is the optimal exposure of any rectilinear trajectory. This completes the proof. □

In the second type of trajectory the drone must start from a given point S (see Figure 6) and accumulates the minimum amount of risk for this starting position,

A similar algorithm will work if we are looking for rectilinear trajectory starting from a given source location, say S. The event space is now determined by a rotating line emanating from the point S (see Figure 7).

Theorem 4. *Consider a rectangular area consisting of two parallel lines and k source points $p_1, p_2, \ldots, p_k$, delimited on its length by these two lines. Then for any given starting position S, in time $\mathcal{O}(k)$ we can compute a drone trajectory which minimizes its total risk when starting from position S.*

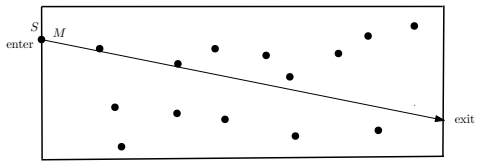


Fig. 6. A rectangular terrain delimited on its length by two parallel lines at distance 1 from each other. The rectilinear trajectory of the drone M is assumed to emanate from a given origin point S and lie between the two parallel lines of the area.

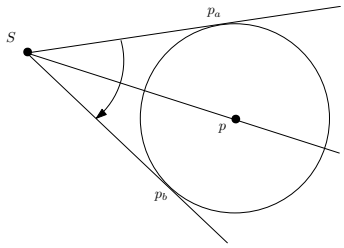


Fig. 7. For any source point p consider the zone with center p and sensitivity distance r. A line emanating from the source S scans the area rotating clockwise: it first intersects the zone circle at the tangent point p_b, then passes through the center p, and finally intersects the zone circle at the tangent point p_a.

V. Strategies for Curvilinear Trajectories

In Section IV, we provided an algorithm that yields optimal trajectories, in a given direction, and asymptotically optimal rectilinear ones, overall. However, reduction of the amount of risk incurred is possible by adopting general curvilinear trajectories.

The core idea developed in this section is the following. During the traversal of a rectilinear trajectory ℓ, when a risky zone is encountered, the drone abandons ℓ. It flies along the perimeter of the risky zone until it encounters ℓ again. Then, it resumes the traversal of the rectilinear trajectory ℓ, see Figure 8. Because it follows its perimeter, it is evident that

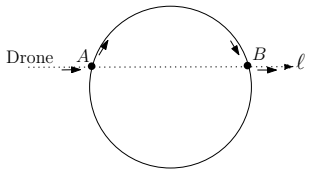


Fig. 8. The drone abandons its rectilinear trajectory ℓ (dotted line). Starting at A, it traverses clockwise the arc AB. It resumes its rectilinear trajectory at B. Note that the drone has two choices at A: it may traverse the perimeter either clockwise (depicted in the figure) or counter clockwise.

the drone reduces the risk from the encountered zone. Note that when the drone reaches the perimeter of a risky zone at A for the first time it has two choices. It may traverse the perimeter either clockwise (depicted in the figure) or counter clockwise. Although the two paths may differ in the time, it takes to complete the traversal, the choice does not affect the

ultimate goal of reaching the destination T and minimizing the risk for the zone encountered.

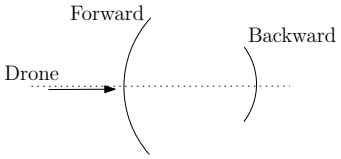


Fig. 9. During its trajectory (dotted line) the drone may encounter either a forward (left) arc, that brings the drone closer, or backward (right) arc, that brings the drone away from the target.

Let S be the starting position of a drone and T its destination. The main difficulty arises in making sure that the drone is not lost in the maze of risky zones and reaches its target. The drone knows the coordinates of both S and T and can remember the straight line ST. In addition, it knows (or can determine) the source point of a risky zone when it encounters its perimeter during its trajectory. During its traversal, the drone may encounter either *forward* circles, that may lead closer to the target T, or *backward* circles, that never lead closer to the target T, see Figure 9. The drone accepts a forward circle only if T does not lie within the zone, but it never accepts a backward circle. We now describe in more detail the update rule for the trajectory of the drone in the Algorithm 2 for curvilinear traversal.

Algorithm 2 Curvilinear traversal from S to T

```
Start at S and follow the straight line ST
repeat
    When a forward circle C is encountered...
    if C contains T then
        Follow the straight line ST
    else
        Follow the perimeter of C until meeting line ST
        Follow the straight line ST
    end if
until target T is reached
```

Using Algorithm 2, two example curvilinear trajectories are depicted in Figure 10.

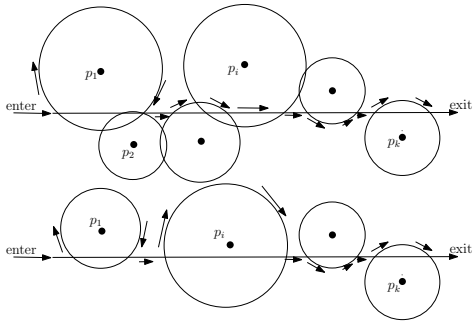


Fig. 10. Curvilinear trajectory following Algorithm 2: for zones with overlaps (top), and without overlaps (bottom).

Theorem 5. *Algorithm 2 always finds a trajectory from the source S to the target T. In a sparse environment of pairwise non-overlapping circles so that S, T do not lie in the interior of any circle the trajectory is also of zero risk.*

Proof. Observe that there is a linear number of circles that the drone may encounter during its traversal. Therefore, the proof follows from the fact that the algorithm always moves forward and closer to the target. The second assertion follows from the fact that the circles are pairwise non-overlapping. □

Algorithm 2 can be realized using attraction and repulsion potential functions [6], [7]. To follow the straight-line ST, the drone is attracted to landmarks on ST. To minimize contact with risky zones, the drone is repelled from source points. Although Algorithm 2 is best effort, we can enhance the performance of the algorithm in selecting trajectories by scanning many possible directions and accepting the optimal one, at the cost of additional complexity. Implementation using attraction and repulsion forces, performance and simulation of Algorithm 2 are discussed further in Section VI.

VI. Simulation results

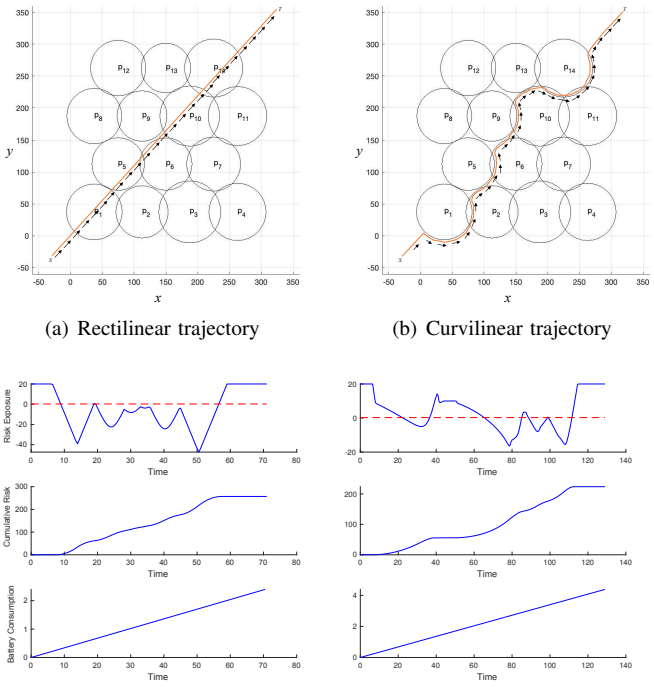
We implemented Algorithms 1 and 2 using and extending SwarmLab [14], a Matlab™ simulation environment for swarms of drones supporting object collision avoidance algorithms, based on attraction and repulsion functions authored by Olfati-Saber and Murray [12] and Vásárhelyi et al. [13]. We also review our simulations results.

The SwarmLab simulation environment captures physical properties of quadcopters and fixed-wing drones, based on software implementation of the models in Refs. [15], [16]. The models include mass, aerodynamics and control parameters; path planning variables represented by a series of waypoints (e.g., starting position S and intermediate waypoints in rectangular terrains); and graphic tools to plot state variables associated to the drones and obstacles.

We have extended the obstacle avoidance algorithms in Ref. [14], and their implementation in Swarmlab, to experiment with risky zone avoidance algorithms applying curvilinear trajectory strategies. We adapted the original potential functions for the attraction between agents and repulsion from obstacles, by assuming that solid obstacles are now non-solid zones, hence relaxing the threshold potential schemes in [12], [13]. The idea is as follows. Drones, modeled in Swarmlab as mobile agents, can pass through what were originally considered to be obstacles, but minimizing the overlap. Drones are repelled from a risky zone by applying repulsion potential functions, and attracted to stationary landmarks, represented as non-mobile agents (already modeled as attraction potential functions in Refs. [12], [13]). As a result, we can support both rectilinear and curvilinear itineraries, trough a series of intermediate waypoints, in addition to source starting position S and target exit position T, as discussed in Sections IV and V.

Monte Carlo simulation results obtained with our extended version of Swarmlab allows us to track and compare the amount of cumulative risk levels (w.r.t. Eq. (5)) vs. battery

consumption of drones using either rectilinear or curvilinear trajectories. Battery consumption functionality comes directly from the Swarmlab Matlab™ code, available in the github repository cited in Ref. [14]. Simulations use randomized configurations with regard to zone sparsity and overlaps (cf. for instance results plotted in Figure 11). We validate that curvilinear trajectories minimize the risk, at the cost of increasing the battery consumption. Additional results, together with the code of our modifications w.r.t. the original Swarmlab simulator, are available in a companion github repository [17].



(a) Rectilinear trajectory (b) Curvilinear trajectory

(c) Rectilinear results (d) Curvilinear results

Fig. 11. Simulation results using the new features of our modified SwarmLab [14] simulation environment, available in a companion github repository [17]. (a) Rectilinear single drone simulation. (b) Curvilinear trajectory simulation (following curvilinear references adapted from the collision avoidance algorithms in [12], [13]) (c) Simulation results for rectilinear trajectory simulations, plotting risk exposure (top), cumnulative risk level w.r.t. Equation (5) (middle), and battery consumption (bottom). (d) Simulation results for curvilinear trajectory simulations.

VII. Conclusion

We have considered the problem of drones traversing terrains that may expose them to certain risks, e.g., disturbing weather phenomena. The goal is to provide means to drones to navigate terrains while minimizing the amount of risk. We have developed a model for quantifying the exposure to a risk factor in a circular zone model. We proposed two main risk reduction strategies, for rectilinear or curvilinear trajectories. We have validated the work by extending SwarmLab [14], a Matlab™ environment for swarms of drones. We have shown that curvilinear trajectories minimize the risk, at the cost of increasing the energy consumption. In our analysis, we use a circular zone model. We note that a similar analysis is also valid if instead of the area of the sector $ADBE$ the length of the chord AB (cf. Figure 1) is being used as risk metric. It is also possible to extend the analysis in Section IV to arbitrary convex (or even polygonal) zones. However, the cost of searching for an optimal traversal algorithm becomes higher as it depends on the total number of vertices of the polygonal zones. More complicated curvilinear strategies are possible. One has to consider the resulting trade-off beradeween minimizing the risk and the the time of arrival to the target. This would be an interesting question to consider in future work.

Acknowledgments — Work partially supported by the Natural Sciences and Engineering Research Council of Canada and the European Commission (H2020 SPARTA project, grant agreement 830892).

References

[1] M. Mozaffari, A. T. Z. Kasgari, W. Saad, M. Bennis, and M. Debbah, "3D Cellular Network Architecture with Drones for beyond 5G," in *2018 IEEE Global Communications Conference (GLOBECOM)*, 2018, pp. 1–6.

[2] H. Ullah, N. Gopalakrishnan Nair, A. Moore, C. Nugent, P. Muschamp, and M. Cuevas, "5G Communication: An Overview of Vehicle-to-Everything, Drones, and Healthcare Use-Cases," *IEEE Access*, vol. 7, pp. 37 251–37 268, 2019.

[3] T. Mori and S. Scherer, "First results in detecting and avoiding frontal obstacles from a monocular camera for micro unmanned aerial vehicles," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 1750–1757.

[4] D. Falanga, K. Kleber, and D. Scaramuzza, "Dynamic obstacle avoidance for quadrotors with event cameras," *Science Robotics*, vol. 5(40), 2020.

[5] M. Ben-Ari and F. Mondada, *Elements of Robotics*. Springer International Publishing, 2017.

[6] O. Khatib, *Real-Time Obstacle Avoidance for Manipulators and Mobile Robots*. New York, NY: Springer New York, 1990, pp. 396–404.

[7] A. A. A. Rizqi, A. I. Cahyadi, and T. B. Adji, "Path planning and formation control via potential function for uav quadrotor," in *2014 International Conference on Advanced Robotics and Intelligent Systems (ARIS)*, 2014, pp. 165–170.

[8] A. Sankaranarayanan and I. Masuda, "A new algorithm for robot curve-following amidst unknown obstacles, and a generalization of maze-searching," in *IEEE International Conference on Robotics and Automation*, 1992, pp. 2487–2494.

[9] D. Maravall, J. de Lope, and J. P. Fuentes, "Navigation and self-semantic location of drones in indoor environments by combining the visual bug algorithm and entropy-based vision," *Frontiers in neurorobotics*, vol. 11, p. 46, 2017.

[10] S. U. Kamat and K. Rasane, "A survey on autonomous navigation techniques," in *2018 Second International Conference on Advances in Electronics, Computers and Communications (ICAECC)*, 2018, pp. 1–6.

[11] J. N. Yasin, S. A. S. Mohamed, M. Haghbayan, J. Heikkonen, H. Tenhunen, and J. Plosila, "Unmanned aerial vehicles (uavs): Collision avoidance systems and approaches," *IEEE Access*, vol. 8, pp. 105 139–105 155, 2020.

[12] R. Olfati-Saber and R. M. Murray, "Distributed cooperative control of multiple vehicle formations using structural potential functions," in *IFAC world congress*, vol. 15, no. 1. Barcelona, Spain, 2002, pp. 242–248.

[13] G. Vásárhelyi, C. Virágh, G. Somorjai, T. Nepusz, A. E. Eiben, and T. Vicsek, "Optimized flocking of autonomous drones in confined environments," *Science Robotics*, vol. 3, no. 20, 2018.

[14] E. Soria, F. Schiano, and D. Floreano, "SwarmLab: a Matlab Drone Swarm Simulator, https://arxiv.org/abs/2005.02769," 2020.

[15] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Ieee, 2007, pp. 153–158.

[16] R. W. Beard and T. W. McLain, *Small unmanned aircraft: Theory and practice*. Princeton university press, 2012.

[17] M. Barbeau, J. Garcia-Alfaro, and E. Kranakis, "Risky zone avoidance strategies for drones [extended swarmlab simulator github repository]," https://github.com/jgalfaro/mirrored-rswarm, 2021.