Generating Safe Corridors Roadmap for Urban Air Mobility

Jakub Sláma Petr Váňa Jan Faigl

***Abstract*— Personal air transportation on short distances, so-called Urban Air Mobility (UAM), is a trend in modern aviation that raises new challenges as flying in urban areas at low altitudes induces an additional risk to people and properties on the ground. Risk-aware trajectory planning can mitigate the risk by detouring and flying over less populated and thus less risky areas. Existing risk-aware trajectory planning approaches are computationally demanding single-query methods that are impractical for online usage. Moreover, coordinated planning for multiple aircraft is prohibitively expensive. Therefore, we propose to reduce computational demands by determining low-risk areas called safe corridors and creating a roadmap of safe corridors based on multiple least risky trajectories. The created roadmap can be used in graph-based multi-agent planning methods for coordinated trajectory planning. The proposed method has been evaluated in a realistic urban scenario, suggesting a significant computational burden reduction and less risky trajectories than the current state-of-the-art methods.**

I. Introduction

The *Urban Air Mobility* (UAM) is an emerging trend in the aerospace industry [1]. The UAM aims to fill the current lack of affordable aerial transportation on short distances, mainly in urban areas, and serve as a personal air taxi. An increasing number of small aircraft flying within urban areas can be expected [2] that can consequently increase the risk of an accident within urban areas. Any accident threatens the aircraft, passengers, and people on the ground. Due to the high population density in urban areas, any such accident could have immense consequences, such as the number of casualties and caused material damage.

As proposed in [3], risk-aware trajectory planning is a possible approach to minimize the risks in the case of a malfunction. An aircraft trajectory can be planned such that the induced risk in the case of a malfunction is minimized. However, such an approach can be computationally demanding, and a motion planning roadmap created during planning may be hardly reusable, not to mention the support of multi-aircraft planning. On the other hand, the least risky trajectories tend to be located over less risky areas such as rivers or brownfields [4]. Such low-risk segments of the trajectories, called safe corridors, are commonly used in the least risky paths. Thus, a roadmap of safe corridors can be created. Then, standard graph-based path planning techniques can be employed to plan risk-aware trajectories quickly. Expected start and goal configurations can be inserted in the precomputed roadmap, and the least risky trajectory through the graph can also be found for multiple aircraft.

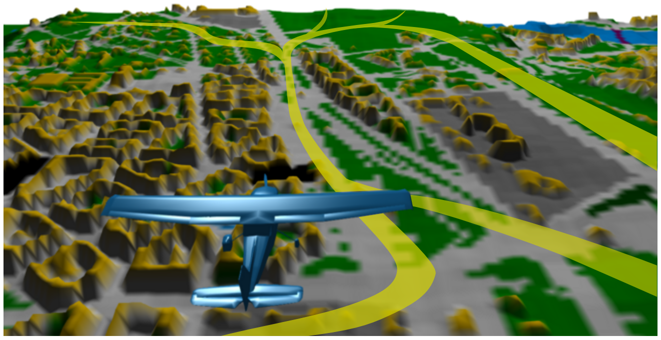


Fig. 1. Least risky trajectories pass through low-risk areas such as above a river, forest, or brownfield. Such commonly low-risk areas called safe corridors are likely to be used if flying through nearby areas. Hence, a roadmap of safe corridors (in yellow) can be created to plan the least risky trajectory instead of extensive planning of a new trajectory.

In this paper, we present a method to create a roadmap of safe corridors based on the analysis of the least risky trajectories. The method identifies safe corridors and creates a roadmap for efficient risk-aware path planning. The approach is based on computing risk-aware trajectories that minimize possible risk in the case of a failure and extracting the safe corridors that cover areas with dense, safe trajectories that would be most likely above less populated areas. The idea of the corridors is visualized in Fig. 1.

The rest of the paper is organized as follows. An overview of the related work is provided in the following section. The formal definition of the studied problem is given in Section III. The proposed method is introduced in Section IV. Results of the performed empirical evaluation of the proposed solution are presented in Section V, and the conclusion and final remarks are in Section VI.

II. Related Work

A possible approach to increase the safety of the UAM is risk-aware trajectory planning can be to provide the least risky trajectories. The existing risk-aware trajectory planning methods are based on single-query randomized motion planning methods that consider risk aspects and possible failures of the aircraft. The most related methods are overviewed in this section to provide background on risk-aware trajectory planning inherently needed in the proposed generation of safe corridors.

First, an aircraft model is required to satisfy vehicle motion constraints. The Dubins airplane [5] is a simplified model of an aircraft based on the Dubins vehicle model [6].

The authors are with the Faculty of Electrical Engineering, Czech Technical University in Prague, Technická 2, 166 27 Prague, Czech Republic {slamajak|vanapet1|faiglj}@fel.cvut.cz

The presented work has been supported by the Czech Science Foundation (GAČR) under the research project No. 22-30043S. The support of the Ministry of Education Youth and Sports (MEYS) of the Czech Republic under project No. LTAIZ19013 is also acknowledged.

The Dubins airplane model is modeled as a vehicle with a constant forward velocity, limited turning radius, and pitch angle. Although the Dubins maneuver is proven to be the shortest unambiguous connection of two points with prescribed heading angles in 2D for Dubins vehicle, such a maneuver cannot be found in the 3D space for Dubins airplane model due to the limited pitch angles. Thus, several trajectory planning methods have been proposed for finding the shortest 3D Dubins maneuver in [7], [8], [9] that can be employed in risk-aware trajectory planning.

The least risky trajectory might not necessarily be the shortest but minimizes possible casualties in the case of aircraft failure. According to the safety report [10], the loss of thrust is the most likely failure type with a significantly higher probability than a failure of the aircraft followed by an uncontrolled fall. The risk-aware planning approaches presented in [3], [11], [12] allow risk mitigation for aircraft flying at a fixed altitude, but the possibility of an emergency landing after loss of thrust is not considered at all. In [13], [14], the authors addressed trajectory planning with the guaranteed possibility of an emergency landing in the case of the loss of thrust. In [15], a risk-aware trajectory planning is proposed that eliminates the risk induced by the loss of thrust by guaranteeing the possibility of an emergency landing while minimizing the risk induced by the total failure of the aircraft. However, the methods are too computationally demanding for quick on-demand queries.

The aforementioned risk-aware trajectory planning methods are single-query methods providing a solution for a single planning scenario. In contrast, multi-query methods, such as the PRM* [16], can re-use the already determined roadmap for multiple planning instances and become more efficient. However, these methods may not always be usable. For example, numerical optimization-based path planning as in [17] or neural network-based path planning as in [18] are purely single-query, and utilizing multi-query approaches would be difficult. Hence, learning the topology of least-risky trajectories that we can call corridors is desirable.

To the best of the authors' knowledge, there is no method for creating such a roadmap of safe corridors from some existing safe trajectories. Therefore, we present the most related work that can be extended to generate safe corridors in the rest of this section.

A flight path based on pre-determined flyable areas is presented in [19] using the skeletonization of a flyable area to determine candidate points interlaced by the B-splines to assure the feasibility of the path. The work could be extended to extract safe corridors by assuming known paths are the flyable area and performing the skeletonization. However, spatial information defines the flyable area only, and heading is not considered. Thus, the method is not capable of creating directional corridors.

The desired roadmap of safe corridors can be built by clustering trajectories into corridors. Clustering [20] allows dividing objects into groups with a high similarity among objects in the same group. Therefore, known trajectories can be clustered, creating edges in the safe corridor roadmap.

Based on the presented literature review, we consider the Dubins airplane model and existing risk-aware trajectory planning for determining safe trajectories. We propose to employ the general idea of clustering trajectories to determine a roadmap of safe corridors. The studied problem with the considered constraints and assumptions is formally defined in the following section.

III. Problem Statement

The addressed problem is determining a roadmap of the least risky trajectories to speed up the planning process. At the same time, the quality of the solution should be affected as little as possible. The risk-aware trajectory planning problem stands to find the least risky trajectory from an initial to a final configuration, and its formal definition can be found in [15]. However, the necessary background on the considered motion constraints and assumptions is briefly introduced here to make the paper self-contained.

A fixed-wing aircraft is modeled as the Dubins airplane model [5]. The vehicle configuration q consists of its position $(x, y, z) \in \mathbb{R}^3$, heading angle $\theta \in \mathbb{S}$, and pitch angle $\psi \in \mathbb{S}$, i.e., $q = (x, y, z, \theta, \psi)$, that define the configuration space $\mathcal{C} = \mathbb{R}^3 \times \mathbb{S}^2$. The Dubins Airplane moves as

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta} \end{bmatrix} = v \begin{bmatrix} \cos\theta\cos\psi \\ \sin\theta\cos\psi \\ \sin\psi \\ u_\theta \rho^{-1} \end{bmatrix}, \tag{1}$$

where the vehicle forward velocity is v, the control input $u_\theta \in [-1, 1]$ changes the heading angle θ, and ρ denotes the minimum turning radius. The pitch angle ψ is considered to change significantly faster than the heading angle θ in the Dubins airplane model. Thus, abrupt changes in the pitch angle are allowed, but the pitch angle must be within the given limits $\psi \in [\psi_{\min}, \psi_{\max}]$. The environment can contain obstacles $\mathcal{O}$, and therefore, the vehicle trajectory is planned in the collision-free part of the configuration space $\mathcal{C}_{\text{free}}$.

A risk r to people on the ground induced by a point-to-point trajectory $\Gamma : [0, T_\Gamma] \to \mathcal{C}_{\text{free}}$ from $q_i \in \mathcal{C}_{\text{free}}$ to $q_f \in \mathcal{C}_{\text{free}}$ at the end time T_Γ is given as

$$r(\Gamma) = \int_0^{T_\Gamma} \mathcal{M}\left(\Gamma(t)\right) \, \mathrm{d}t, \tag{2}$$

where $\mathcal{M}$ is an aircraft-dependent risk map, and $\mathcal{M}(q)$ denotes a risk associated with a configuration q. The risk-aware trajectory planning problem of finding the least risky trajectory can be formally defined as determining a trajectory with minimal trajectory-induced risk r. The optimal path Γ^* for the given endpoints is defined as

$$\Gamma^* = \arg\min_{\Gamma} r(\Gamma), \tag{3}$$

$$\text{s.t.} \qquad \Gamma^*(0) = q_i, \quad \Gamma^*(T_{\Gamma^*}) = q_f, \tag{4}$$

and the corresponding risk induced by such a trajectory is

$$r^*(q_i, q_f) = r\left(\Gamma^*\right). \tag{5}$$

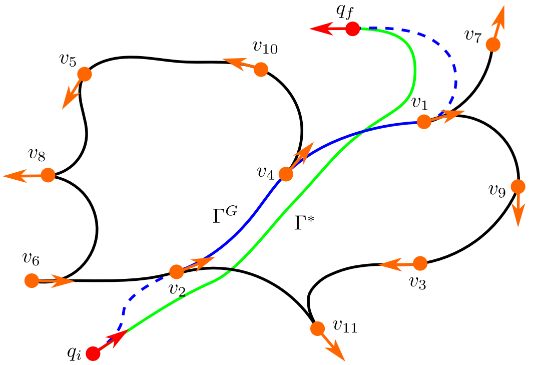


Fig. 2. An example of the roadmap G with vertices $\boldsymbol{V}$ (orange) and edges $\boldsymbol{E}$ (black/blue). The high-quality/optimal path Γ^* between the selected configurations q_i and q_f (red) is depicted in green. The least risky trajectory Γ^G obtained from the roadmap G is depicted in blue, connections of q_i and q_f with the roadmap are denoted by dashed lines. Because Γ^G is only an approximation of Γ^*, its overall risk r^G can be higher than the risk r^* of the individually determined (high-quality/optimal) trajectory Γ^*.

The herein studied problem is to find a roadmap G with vertices $\boldsymbol{V}$ and edges $\boldsymbol{E}$ usable in multi-query planning of the least risky trajectories. The roadmap G can be considered an approximation of the optimal risk-aware trajectories, and a trajectory Γ^G determined in G approximates the optimal trajectory Γ^*, which is visualized in Fig. 2. Thus, we aim to find an approximation that would minimize the relative risk of the roadmap trajectories compared to the optimal ones.

Let Γ^G be the least risky trajectory in G between configurations q_i and q_f. The trajectory goes through intermediate configurations $v_i \in \boldsymbol{V}$. The indices of such vertices can be expressed as $\Sigma = \{\sigma_1, \ldots, \sigma_{|\Sigma|}\}$. The risk r^G of Γ^G from q_i to q_f in the roadmap G can be then expressed as

$$r^G(q_i, q_f) = \min_{\Sigma} \left[r(q_i, v_{\sigma_1}) + \sum_{l=1}^{|\Sigma|-1} r(v_{\sigma_l}, v_{\sigma_{l+1}}) + \right. \\ \left. + r(v_{\sigma_{|\Sigma|}}, q_f) \right]. \quad (6)$$

Now, we can express the creation of safe corridors as determination of the roadmap G with k vertices using h trajectories that would cover the operational area such that the mean relative risk induced by the trajectories from G is minimized. The optimization can be formally expressed as

Problem 3.1 (Determining Safe Corridors Roadmap G)

$$\underset{G}{\text{minimize}} \ \frac{1}{h} \sum_{j=1}^{h} \frac{r^G(q_i^j, q_f^j)}{r^*(q_i^j, q_f^j)}. \quad (7)$$

In theory, we can consider an infinite number of trajectories that would cover the whole $\mathcal{C}_{\text{free}}$ by optimal trajectories with the minimal risk r^* and consider the limit in (7) for $h \to \infty$. In practice, we are limited by the available computational time; therefore, only a limited number of trajectories can be utilized. Besides, the optimal trajectories found by the asymptotically optimal motion planner, such as RRT*-based [15], would provide trajectories whose quality depends on the provided computational time. However, for a particular set of the least risky trajectories Γ_i, we aim to determine G that would be sufficiently sparse to support a quick determination of trajectories Γ_i^G with a similar risk as it would have for Γ_i.

IV. Proposed Method

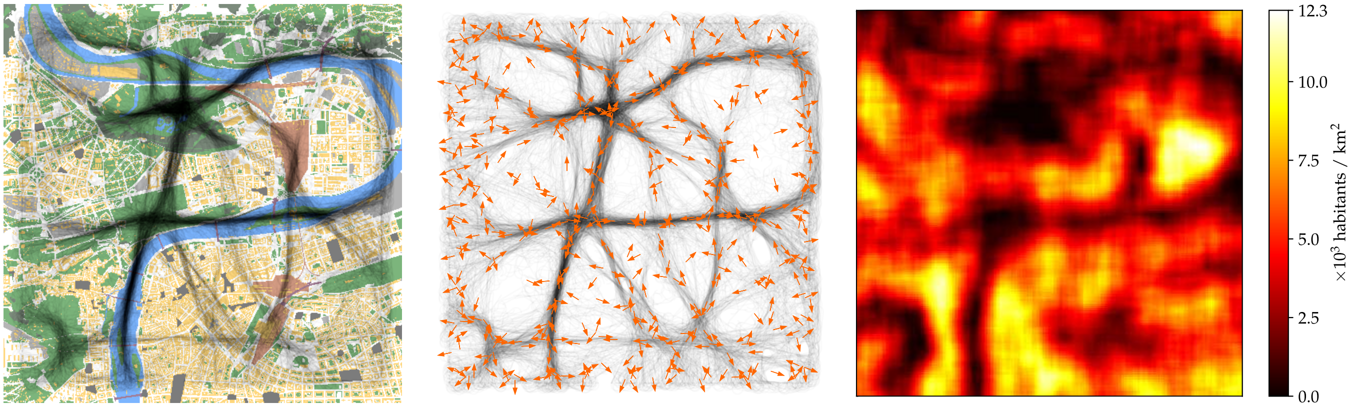
The proposed method creates a roadmap of safe corridors based on processing a huge number of trajectories of the point-to-point risk-aware planning instances. Therefore, the first phase of the method is a solution to point-to-point planning instances. Then, the determined trajectories are processed into a roadmap of safe corridors. The presented method is focused on generating the corridors from trajectories that can be determined by an existing risk-aware trajectory planning method. Thus, only a brief overview of the point-to-point planning is presented. Finally, the whole proposed method is summarized in Algorithm 1 that is further described in detail in the following parts.

Algorithm 1: Safe Corridors Roadmap Creation

```
Input: $\mathcal{M}$ – Risk map.
Parameter: $\mathcal{A}$ – Aircraft model.
Parameter: $k$ – Number of clusters.
Parameter: $n$ – Number of scenarios.
Output: $G$ – Roadmap of safe corridors.
1  Function CreateCorridors($\mathcal{M}$):
2      $\mathcal{Q} \leftarrow \emptyset$
3      for $j \leftarrow 1$ to $n$ do
4          $(q_i, q_f) =$ GenerateScenario()
5          $\Gamma_j \leftarrow$ PlanTrajectory($q_i, q_f, \mathcal{M}$)
6          $\{q_1^j, q_2^j, ..., q_{m_j}^j\} \leftarrow$ SamplePath($\Gamma_j, \delta$)
7          $\mathcal{Q} \leftarrow \mathcal{Q} \cup \{q_1^j, q_2^j, ..., q_{m_j}^j\}$
8      $\mathcal{C} \leftarrow$ Cluster($\mathcal{Q}, k$)
9      $G \leftarrow \{\boldsymbol{V} \leftarrow \mathcal{C}, \boldsymbol{E} \leftarrow \emptyset\}$
10     for $i \leftarrow 1$ to $n$ do
11         for $j \leftarrow 1$ to $m_i - 1$ do
12             $v' \leftarrow$ NearestCenter$(q_j^i, \boldsymbol{V})$   // Eq. 12
13             $v'' \leftarrow$ NearestCenter$(q_{j+1}^i, \boldsymbol{V})$ // Eq. 12
14             if $v' \neq v''$ then
15                 $\boldsymbol{E} \leftarrow \boldsymbol{E} \cup \{(v', v'')\}$
16     return $G$
```

A. Generation of Trajectories and Input Samples

The roadmap is created from the solution of the point-to-point planning instances that are generated randomly such that the pairs of start and goal configurations uniformly cover the whole given map of the operational area. Once a planning scenario is generated by the method `GenerateScenario` (Algorithm 1, Line 4), it is solved by the `PlanTrajectory` routine using the RRT*-based risk-aware trajectory planning [15] that provides the least risky trajectory Γ_j for the given scenario. The obtained trajectory consists of consequent Dubins maneuvers with various radii adopted from [21]. The trajectory is sampled by the `SamplePath` with



(a) Trajectories over the city map

(b) Centers of the clusters

(c) Population density

Fig. 3. A heatmap of the least risky trajectories obtained by [15] and results of their clustering: (**a**) 3 678 trajectories, and the city map; (**b**) resulting cluster centers (orange); (**c**) population density layer of the map. The darker the area is, the more trajectories are passing through it. A strong network of highly used areas can be noticed; however, trajectory direction is omitted, and only a 2D projection is shown. The trajectories have been sampled and clustered into $k = 500$ clusters. Notice that the cluster centers are located along the most commonly used areas by the least risky trajectories.

the sampling step δ (Algorithm 1, Line 6), and m_j obtained samples are added into the set of all samples $\mathcal{Q}$. The process is repeated until the desired number of scenarios is solved.

B. Generation of Safe Corridors Roadmap

Once the determined trajectories are sampled, the samples are clustered by the `Cluster` routine providing the centers of clusters (Algorithm 1, Line 8). The goal is to substitute all near samples with a single vertex in the roadmap. The used clustering is based on k-means algorithm [22] modified for $SE(2)$ by the distance function between two samples q_1 and q_2 as

$$\text{dist}(q_1, q_2) = \max\left(\left\| q_1^{\text{2D}} - q_2^{\text{2D}} \right\|, \frac{d_{\max}}{\Delta_{\max}^{\theta}} \left| \angle(q_1^{\theta}, q_2^{\theta}) \right| \right), \quad (8)$$

where $\left\| q_1^{\text{2D}} - q_2^{\text{2D}} \right\|$ is the Euclidean distance between the samples, $\angle(q_1^{\theta}, q_2^{\theta})$ is the angular difference between their headings, and $\Delta_{\max}^{\theta}$ and $d_{\max}$ are the maximum heading difference and Euclidean distance between two samples to be considered as near, respectively. The mean configuration $c_j = (x_j, y_j, z_j, \theta_j, \psi_j)$ from a set of configurations Ω_j belonging to the j-th cluster is determined as

$$(x_j, y_j, z_j) = \frac{1}{|\Omega_j|} \sum_{q \in \Omega_j} (q.x, q.y, q.z), \quad (9)$$

$$\theta_j = \texttt{atan2} \left[\sum_{q \in \Omega_j} \left(\sin\left(q.\theta\right), \cos\left(q.\theta\right) \right) \right], \quad (10)$$

$$\psi_j = \texttt{atan2} \left[\sum_{q \in \Omega_j} \left(\sin\left(q.\psi\right), \cos\left(q.\psi\right) \right) \right]. \quad (11)$$

Once the samples are clustered, the obtained centers of the clusters $\mathcal{C}$ form vertices $\boldsymbol{V}$ of the safe corridors roadmap G (Algorithm 1, Line 9). The edges can be created between all vertices within some neighborhoods, similar to the PRM*. However, it would result in unnecessary connections never used in the least risky trajectory. Besides, the complexity of graph-based planning algorithms depends on the number of edges. Thus, connecting all vertices would unnecessarily increase the overall computational demands. Therefore, the edges $\boldsymbol{E}$ of the roadmap G are determined based on the original samples' connections as follows.

The nearest cluster center is found by

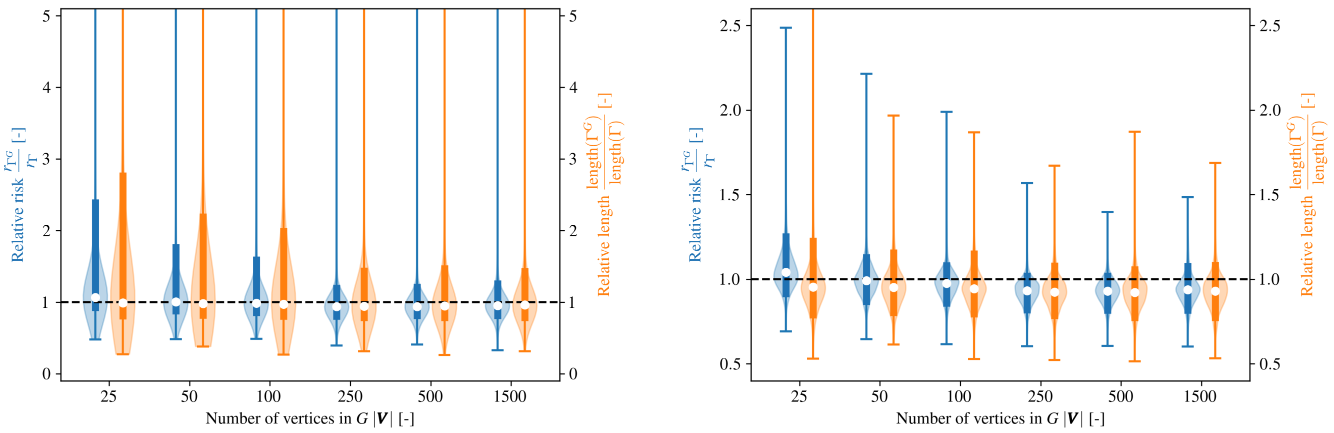
$$\texttt{NearestCenter}(q, \boldsymbol{V}) = \arg\min_{v \in \boldsymbol{V}} \text{dist}(q, v) \quad (12)$$

for two consecutive samples q_j^i and q_{j+1}^i of the same trajectory Γ_i (Algorithm 1, Lines 12 and 13). If the samples are from different clusters, an edge between the centers of these clusters, denoted as v' and v'', is created in the roadmap G.

V. Results

The proposed safe corridors roadmap determination method has been evaluated in a realistic urban scenario to demonstrate its performance and behavior. The assumed urban area is 5 km×5 km large, based on the Prague city center. The aircraft model utilized in the evaluation is Cessna 172, adopted from [13], with a minimum turning radius of 65.7 m.

First, 3 678 pairs of the start and goal configurations have been generated such that these pairs uniformly cover the whole map. Then, the least risky trajectory has been found for each generated planning instance using [15], which represents the reference method. Based on therein reported results, the planning time for each instance has been set to 750 s. The resulting trajectories are visualized in Fig. 3a. Finally, the found trajectories have been sampled, and the samples have been clustered using k-means clustering with the proposed distance function (8). The maximum angular difference $\Delta_{\max}^{\theta} = 30°$ and Euclidean distance $d_{\max} = 250$ m have been used in (8). Various numbers of the clusters $k \in \{25, 50, 100, 250, 500, 1500\}$ have been examined to evaluate their influence on the roadmap quality. Note that k defines the number of vertices in the created roadmap G. An example of found roadmap vertices (cluster centers) for $k = 500$ is depicted in Fig. 3b. A risk associated with the found connections in the safe corridors roadmap has been evaluated by a risk function adopted from [15].



(a) All scenarios

(b) Scenarios where $\|q_i^{2\mathrm{D}} - q_f^{2\mathrm{D}}\| > 2000\,\mathrm{m}$

Fig. 4. Influence of the number of vertices in the roadmap G on the relative risk and relative length (normalized with respect to the reference method): (**a**) results from all solved scenarios; (**b**) results only from scenarios in which start q_i and goal q_f configurations are distant. Medians (white dots) and 90% non-parametric confidence intervals (thick vertical lines) are shown. The relative risk decreases with an increasing number of vertices as denser roadmaps are obtained. Once the area is sufficiently covered ($|\boldsymbol{V}| = 250$), a further increase in the number of vertices does not improve the roadmap quality. However, for too many vertices ($|\boldsymbol{V}| = 1\,500$), the roadmap quality decreases slightly as the clusters used for vertex generation become too small.

The proposed method has been implemented in Julia ver. 1.6.2 [23] and executed on a single core of the Intel® Core™ i7-9750H CPU. The generated scenarios have been solved using the found roadmap; the overall trajectory risk has been compared to the original risk-aware planning results to evaluate the quality of the found roadmap as follows. The start and goal configurations are inserted into the roadmap, and the planning instance is solved by Dijkstra's algorithm [24], a graph-based planning algorithm.

TABLE I

Success rate of the proposed method on 3 678 scenarios

No. of vertices k	25	50	100	250	500	1500
Success rate [%]	99.27	99.54	99.46	99.73	99.95	99.97

Based on the success rate depicted in Table I, only a few scenarios have not been solved. It is because the start and goal configurations are inserted into the roadmap by a single Dubins maneuver. Therefore, a feasible insertion may not be possible in cases where all the possible connections violate no-flight zones. A more complex maneuver would be needed, which is considered out of the paper's scope.

Evaluation results on the induced risk and trajectory length of paths obtained by the proposed safe corridors roadmap compared to the reference method [15] are depicted in Fig. 4. The induced risk generally decreases with the increasing number of vertices as the safe corridors are better covered. For sufficient corridor coverage, the roadmap quality does not increase with the number of vertices. However, the risk starts to slightly increase from a certain number of vertices, suggesting the clusters become too small, which is supported by the increasing length of the obtained solutions.

Surprisingly, the proposed method provides better results than the reference in many cases. The reference method is asymptotically complete, meaning that the quality of its solution increases with increasing planning time. Although the planning time 750 s seemed sufficient, the provided results are still only close to optimum. The proposed method creates safe corridors from a huge number of close-to-optimum trajectories by the reference method; safe corridors closer to the optimum can be built. The risk reduction would most likely be less significant for longer planning times. The roadmap quality as defined in (7) based on $h = 3\,678$ trajectories is depicted in Table II. Based on the results, $k = 500$ vertices in the roadmap have been selected for further detailed analysis as a suitable trade-off between the computational demands and the roadmap quality.

TABLE II

Roadmap quality based on 3 678 trajectories

No. of vertices k	25	50	100	250	500	1500
Roadmap quality [-]	1.33	1.16	1.11	1.00	1.00	1.02

A. Detail Performance Analysis

For $k = 500$ vertices, the median of risk reduction is 6 % compared to the reference method with 90 % non-parametric confidence interval $[0.76, 1.25]$ of the relative risk. The median of the length reduction is 5 %, and 90 % non-parametric confidence interval of relative length is $[0.73, 1.51]$. On the contrary, the trajectory risk and length increase significantly in some cases. As it can be seen in Fig. 5, it is only the case for instances where the start and goal configurations are close to each other; hence, connecting to the roadmap from the start configuration and from the roadmap to the goal configuration significantly increases the path length, and thus its risk. We argue that such short flights are very unlikely in real life; hence, the practical impact would be minimal.

In addition to reducing trajectory risk and length, the proposed method significantly reduces the computational demand for solving a single query, see computational times depicted in Table III. Although the time to create the roadmap increases with the number of roadmap vertices, the average time of single query decreases. It is because a dense roadmap allows shorter insertion maneuvers of the start and

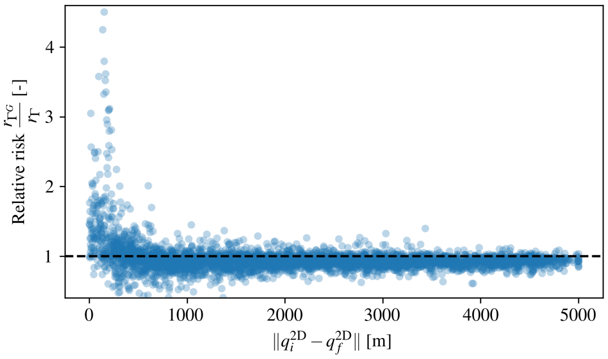


Fig. 5. The relative risk (normalized by the reference method) depends heavily on the distance between the start and goal configurations. The closer they are, the more significant role of the extra connection from the start to the roadmap and from the roadmap to the goal is. However, the very short flights are very unlikely in a real-life scenario.

TABLE III

COMPUTATIONAL DEMANDS OF THE PROPOSED METHOD

No. of vertices k	25	50	100	250	500	1500
Pre-generation of $\mathcal{Q}$	28.5 CPU days					
Clustering time [s]	24	71	77	264	480	954
Roadmap creation [s]	451	416	551	488	573	782
Avg. query time [s]	2.9	1.9	1.9	1.1	1.1	0.8

goal configurations, and the used risk evaluation method [15] scales with the length of the maneuvers. The average time of the single query decreases with shorter insertion maneuvers and thus with the increased number of roadmap vertices.

The safe corridor roadmap is determined in several minutes for the utilized computational environment with a single core CPU. However, a solution query is obtained in the order of units of seconds on average compared to 750 s of the single-query reference method [15].

VI. Conclusion

In this paper, we address a generation of safe corridors roadmap for risk-aware trajectory planning in UAM scenarios. The addressed problem is motivated by decreasing computational demands of the existing single-query risk-aware trajectory planning methods. A single-query approach is relatively computationally demanding and can only provide the least risky trajectory for a given scenario. The proposed method determines commonly used areas called safe corridors among the least risky trajectories determined for various scenarios. The created roadmap of safe corridors can be utilized in multi-query risk-aware trajectory planning using quick graph-based search algorithms that provide the trajectory within a few seconds compared to hundreds of seconds needed by the single-query approach. Furthermore, the proposed method finds less risky trajectories than the reference method. It is because the single-query approach determines a single trajectory while the proposed method aggregates a huge number of trajectories, each determined for the same planning time. Once the roadmap is created, the proposed approach provides better risk-aware trajectories with significantly lower computational requirements than the previous approach.

References

[1] S. Hasan, "Urban air mobility (uam) market study," National Aeronautics and Space Administration (NASA), Tech. Rep., 2019.

[2] M. Moore, "21st century personal air vehicle research," in *AIAA International Air and Space Symposium and Exposition: The Next 100 Years*, 2003, p. 2646.

[3] S. Primatesta, A. Rizzo, and A. la Cour-Harbo, "Ground risk map for unmanned aircraft in urban environments," *Journal of Intelligent & Robotic Systems*, vol. 97, no. 3, pp. 489–509, 2020.

[4] R. Melnyk, D. Schrage, V. Volovoi, and H. Jimenez, "A third-party casualty risk model for unmanned aircraft system operations," *Reliability Engineering & System Safety*, vol. 124, pp. 105–116, 2014.

[5] H. Chitsaz and S. M. LaValle, "Time-optimal paths for a dubins airplane," in *IEEE Conference on Decision and Control*, 2007, pp. 2379–2384.

[6] L. E. Dubins, "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *American Journal of mathematics*, vol. 79, no. 3, pp. 497–516, 1957.

[7] G. Ambrosino, M. Ariola, U. Ciniglio, F. Corraro, A. Pironti, and M. Virgilio, "Algorithms for 3d uav path generation and tracking," in *IEEE Conference on Decision and Control*, 2006, pp. 5275–5280.

[8] S. Hota and D. Ghose, "Optimal path planning for an aerial vehicle in 3d space," in *IEEE Conference on Decision and Control*, 2010, pp. 4902–4907.

[9] J. Herynek, P. Váňa, and J. Faigl, "Finding 3d dubins paths with pitch angle constraint using non-linear optimization," in *European Conference on Mobile Robots (ECMR)*. IEEE, 2021, pp. 1–6.

[10] J. D. Kenny, *26th Joseph T. Nall Report*. Richard G. McSpadden, JR., 2017.

[11] S. Primatesta, G. Guglieri, and A. Rizzo, "A risk-aware path planning strategy for uavs in urban environments," *Journal of Intelligent & Robotic Systems*, vol. 95, no. 2, pp. 629–643, 2019.

[12] S. Primatesta, M. Scanavino, G. Guglieri, and A. Rizzo, "A risk-based path planning strategy to compute optimum risk path for unmanned aircraft systems over populated areas," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2020, pp. 641–650.

[13] P. Váňa, J. Sláma, J. Faigl, and P. Pačes, "Any-time trajectory planning for safe emergency landing," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 5691–5696.

[14] P. Váňa, J. Sláma, and J. Faigl, "Surveillance planning with safe emergency landing guarantee for fixed-wing aircraft," *Robotics and Autonomous Systems*, vol. 133, p. 103644, 2020.

[15] J. Sláma, P. Váňa, and J. Faigl, "Risk-aware trajectory planning in urban environments with safe emergency landing guarantee," in *IEEE International Conference on Automation Science and Engineering (CASE)*, 2021, pp. 1606–1612.

[16] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011.

[17] C. Sun, Y.-C. Liu, R. Dai, and D. Grymin, "Two approaches for path planning of unmanned aerial vehicles with avoidance zones," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 8, pp. 2076–2083, 2017.

[18] R. Glasius, A. Komoda, and S. C. Gielen, "Neural network dynamics for path planning and obstacle avoidance," *Neural Networks*, vol. 8, no. 1, pp. 125–133, 1995.

[19] T.-Y. Sun, C.-L. Huo, S.-J. Tsai, and C.-C. Liu, "Optimal uav flight path planning using skeletonization and particle swarm optimizer," in *IEEE Congress on Evolutionary computation*, 2008, pp. 1183–1188.

[20] L. Rokach and O. Maimon, "Clustering methods," in *Data mining and knowledge discovery handbook*. Springer, 2005, pp. 321–352.

[21] K. Kučerová, P. Váňa, and J. Faigl, "On finding time-efficient trajectories for fixed-wing aircraft using dubins paths with multiple radii," in *Annual ACM Symposium on Applied Computing*, 2020, pp. 829–831.

[22] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Berkeley symposium on mathematical statistics and probability*, vol. 1, no. 14, 1967, pp. 281–297.

[23] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, "Julia: A fresh approach to numerical computing," *SIAM review*, vol. 59, no. 1, pp. 65–98, 2017.

[24] E. W. Dijkstra *et al.*, "A note on two problems in connexion with graphs," *Numerische mathematik*, vol. 1, no. 1, pp. 269–271, 1959.