Emergency Landing Aware Surveillance Planning for Fixed-wing Planes

Petr Váňa Jan Faigl Jakub Sláma

***Abstract*— In this paper, we introduce the Emergency Landing Aware Surveillance Planning (ELASP) problem that stands to find the shortest feasible trajectory to visit a given set of locations while considering a loss of thrust may happen to the vehicle at any time. Two main challenges can be identified in ELASP. First, the ELASP is a planning problem to determine a feasible close-loop trajectory visiting all given locations such that the total trajectory length is minimized, which is a variant of the traveling salesman problem. The second challenge arises from the safety constraints to determine the cost-efficient trajectory such that its altitude is sufficiently high to guarantee a gliding emergency landing to a nearby airport from any point of the trajectory. Methods to address these challenges individually already exist, but the proposed approach enables to combine the existing methods to address both challenges at the same time and returns a safe, feasible, and cost-efficient multi-goal trajectory for the curvature-constrained vehicle.**

I. Introduction

Loss of thrust is a critical situation for any plane because it forces the (auto)pilot to react promptly and steer the vehicle towards the nearest location where it can land safely. Although planes are usually able to land without any thrust from the motor, such a maneuver is impossible if the initial altitude of the vehicle is too low. Besides, an obstacle or high terrain may block the direct path from the current position to the landing site and make the landing impossible.

A common approach is to plan an emergency landing at a particular moment when *Loss of Thrust* (LoT) occurs. However, LoT can happen at any time without any warning, and there is no guarantee that a safe landing trajectory exists. Therefore, we propose to avoid possibly dangerous situations and consider the option of gliding emergency landing directly in the planning process of the vehicle trajectory.

The introduced *Emergency Landing Aware Surveillance Planning* (ELASP) stands to find the shortest trajectory visiting the given set of locations in 3D while a safe landing is possible at any time during the trajectory execution. Regarding the literature review, we consider the introduced ELASP as a novel problem combining characteristics of the two existing problems. First, the ELASP can be seen as a generalization of the multi-goal motion planning [1] where the sequence of visits to the given locations is determined simultaneously with the feasible trajectories connecting the locations while the motion constraints of the vehicle are satisfied. Secondly, the ELASP is also tightly related to the determination of the gliding trajectories because the existence of the safe emergency landing trajectory has to be guaranteed for any point of the ELASP trajectory. Thus, determining a safe emergency landing trajectory needs to be integrated directly into the multi-goal planning process to determine the most suitable (and cost-efficient) sequence of visits to the given locations of interests.

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

The presented work has been supported by the Czech Science Foundation (GAČR) under research project No. 19-20238S.

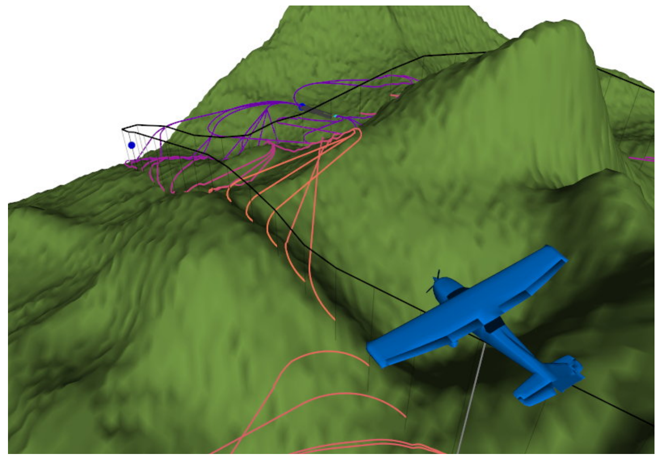


Fig. 1. Visualization of a found ELASP trajectory (black line) for Cessna 172 airplane together with possible emergency landing trajectories (colored lines according the actual altitude) above the mountain terrain.

The proposed solution to the introduced ELASP problem leverages the RRT*-based algorithm [2] for planning emergency landing trajectories that are employed in the determination of the lowest safe altitude at the specific location. The estimation of the safe altitude is used to create a save altitude profile for any trajectory which is further utilized as a travel cost estimation between two locations. Having this estimation, the sequence of visits to the locations can be determined as a solution of the *Traveling Salesman Problem* (TSP) and the final multi-goal trajectory can be then constructed from the corresponding trajectories with the guaranteed emergency landing. Hence, the proposed algorithm provides trajectory satisfying motion constraints of the vehicle while the emergency landing is possible from any point of the final trajectory, e.g., see an example of the found solution in Fig. 1.

The rest of the paper is organized as follows. The related work to the planning of multi-goal and emergency landing trajectories is summarized in the following section. The ELASP problem is formally introduced in Section III and the proposed approach to address the problem is described in Section IV. Results on the empirical evaluation of the proposed algorithm for the ELASP problem are reported in Section V. The final concluding remarks are in Section VI.

II. Related work

The introduced ELASP problem arises from surveillance planning, where the vehicles are requested to visit a set of locations [3]. Thus a solution to ELASP is related to the trajectory planning for curvature-constrained vehicles, but also to the combinatorial multi-goal motion planning for such vehicles. Besides, the problem is also related to emergency landing planning, i.e., the determination of the gliding trajectory. Existing approaches of each particular related area are briefly summarized in the rest of this section.

Trajectory planning for fixed-wing aerial vehicles with curvature-constrained trajectories can be based on the Dubins vehicle model [4], which is advantageous because a closed-form solution is available for the point-to-point trajectory of the 2D plane with the minimum turning radius and defined vehicle heading at the initial and terminal point. The curvature-constrained trajectory planning problem has been extended to the three-dimensional space in [5] where the authors proved the necessary conditions for the optimal path.

A heuristic method for the 3D trajectory generation proposed in [6] can connect arbitrary configurations of the vehicle considering the pitch angle constraints. In [7], such a model of the curvature-constrained trajectory in 3D is called the *Dubins Airplane model*. Implementation details of the generation method are addressed in [8] considering kinematics of a real fixed-wing vehicle in more detail. As an alternative to the aforementioned constructive methods, seventh order Bézier curves are utilized in [9].

Having a method to determine feasible (and eventually optimal) trajectory connecting two configurations of the vehicle, a more general multi-goal trajectory planning can be addressed as a problem to determine a sequence of visits to the given set of locations. The introduced ELASP is a variant of the surveillance planning [3] that stands to determine the sequence of visits together with the determination of the most suitable configurations at which the given locations of interests are visited such that the overall cost of the final trajectory is minimized. In particular, the ELASP problem can be seen as a 3D variant of the existing Dubins Traveling Salesman Problem (DTSP) [10] where the 2D Dubins vehicle model is replaced by the Dubins Airplane model [11].

Several approaches to the DTSP have been proposed that can be roughly divided into decoupled and sampling-based methods [12]. The decoupled methods separate the combinatorial part (to determine the optimal sequence) from the continuous optimization of determining the optimal configurations to visit the locations, e.g., by a solution of the Euclidean TSP followed by determining heading angles at the locations, e.g., by the Alternating Algorithm [10]. In the sampling-based methods, possible heading angles at the given locations are sampled into a finite set of configurations [13] and the problem is formulated as the Generalized Asymmetric TSP that can be transformed by the Noon-Bean transformation [14] to an instance of the regular TSP and solved using existing solvers, e.g., [15]. Alternatively, soft-computing techniques such as genetic [16], memetic [17], or unsupervised learning [18] algorithms can be utilized to address both combinatorial and continuous optimization parts of the DTSP at the same time.

The main difference of the introduced ELASP problem to the DTSP is in the requirements of the emergency landing trajectory from any point of the planned trajectory to guarantee safe landing in the case of loss of thrust. The planning problem of the emergency landing trajectory can be based on a determination of the overall altitude loss of the shortest Dubins maneuver [4] from the particular vehicle location to the closest landing site. Such a concept utilizing the Dubins Airplane model is used in [19] to compute gliding trajectories for the accident on the Hudson River. However, neither obstacles nor altitude of the terrain in the vicinity of the landing site is considered in [19]. A^*-based, genetic, and RRT^* algorithms are proposed in [2], [20], [21], respectively, to address this issue. The emergency landing trajectory can be further generalized by considering the influence of the wind [22], [23]; which however does not address multi-goal planning and nearby terrain.

Based on the literature review, individual sub-problems of the introduced ELASP can be addressed by the existing work. In particular, Dubins Airplane model [8] can be used for the point-to-point 3D curvature-constrained trajectory planning within the context of surveillance planning formulated as the 3D variant of the DTSP [11]. Particular approaches for planning emergency landing trajectories also existing [2], [22], [23]. However, a combination of the surveillance planning with the guarantee on the safe emergency landing has not been reported in the literature. Thus, the main contribution of the proposed approach is that an emergency landing trajectory is guaranteed to exist from any point of the selected trajectory as presented in Section IV.

III. Problem Statement

The ELASP problem, studied in this paper, is to find the shortest trajectory for a fixed-wing vehicle to visit all the given locations such that a safe emergency landing trajectory is guaranteed from any point of the trajectory for the case of the total LoT. ELASP combines three challenges. First, a feasible trajectory satisfying motion constraints of the vehicle, such as the minimum turning radius, or minimum/maximum allowed pitch angle, needs to be determined. Second, an optimal sequence of visits to the locations needs to be determined as well, which makes the ELASP problem NP-hard as it becomes the TSP if motion constraints are relaxed. Finally, the altitude of the trajectory must guarantee safe landing on at least one landing site for any point of the trajectory.

The ELASP problem is considered for a fixed-wing vehicle that is modeled as the Dubins Airplane model [7]. The state q of the vehicle is a combination of its 3D position $(x, y, z) \in \mathbb{R}^3$, heading angle $\theta \in \mathbb{S}$, and pitch angle $\psi \in \mathbb{S}$, i.e., $q = (x, y, z, \theta, \psi)$, and thus the configuration space is $\mathcal{C} = \mathbb{R}^3 \times \mathbb{S}^2$. The Dubins Airplane model can be expressed as the following dynamic system

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

where v stands for the forward velocity and the change of the heading angle θ is controlled by the input $u_0 \in [-1, 1]$ and limited by the minimum turning radius ρ. For the Dubins Airplane model [7], it is assumed that the time constant of changing the pitch angle ψ is significantly lower than for the heading angle, and thus the model allows abrupt pitch changes within the given interval $\psi \in [\psi_{\min}, \psi_{\max}]$.

Safe landing is assumed to be possible on m given landing sites $\Xi = \{\xi_1, \ldots, \xi_m\}$ and for each site, the configuration ξ_j corresponds to the touchdown. The safe emergency landing is guaranteed for the final trajectory $\mathcal{R}$ if for any configuration $q_{\text{act}} \in \mathcal{R}$, there exists an emergency landing trajectory $\Gamma : [0, 1] \to \mathcal{C}_{free}$ which starts at $q_{\text{act}} = \Gamma(0)$ and ends at a particular landing site j or directly above it, i.e., $\Gamma(1)$ is from the set $\widehat{\xi_j}$ of configurations above the landing site ξ_j.[1] Each landing trajectory is strictly above the terrain, and it includes possible altitude loss for the case of LoT.

The multi-goal part of ELASP can be defined as follows. Let $S = \{s_1, s_2, \ldots, s_n\}$ be the n given locations of interest $s_i \in \mathbb{R}^3$. Then the problem is to find a closed-loop trajectory to visits all locations in S such that it satisfies (1). The configuration q_i corresponding to the point of interest $s_i \in S$ is called visiting configuration and the final trajectory $\mathcal{R}$ is composed of n trajectory segments $\mathcal{R}_i : [0, 1] \to \mathcal{C}_{free}$ connecting the corresponding visiting configurations, and thus $\mathcal{R} = \{\mathcal{R}_1, \ldots, \mathcal{R}_n\}$.

The sequence of visits $\Sigma = \{\sigma_1, \ldots, \sigma_n\}$, $\sigma_i \in \{1, \ldots, n\}$ defines a permutation of the points of interest S selected such that the total length of the final trajectory $\mathcal{R}$ is minimized. Since the visiting configuration q_i corresponds to s_i, the trajectory segments $\mathcal{R}_j$ starts at q_{σ_j} and ends at the following configuration of the sequence $q_{\sigma_{j+1}}$ for $i, j \in \{1, \ldots, n\}$. Notice $\sigma_{n+1} \triangleq \sigma_1$, and $\mathcal{R}_{n+1} \triangleq \mathcal{R}_1$, respectively, to simplify the notation for the close-loop trajectory. Having the preliminaries, the *Emergency Landing Aware Surveillance Planning (ELASP)* can be formulated as the optimization Problem 3.1.

Problem 3.1 (Emergency Landing Aware Surveillance Planning (ELASP))

$$\min_{\mathcal{R},\Sigma} \quad \sum_{i=1}^{n} \mathcal{L}(\mathcal{R}_i) \quad (2)$$

s.t. for $i \in \{1, 2, \ldots, n\}$

$$\mathcal{R}_i(0) = \mathcal{R}_{i+1}(1), \mathcal{R}_i(0) = q_{\sigma_i}, \mathcal{R}_i(1) = q_{\sigma_{i+1}} \quad (3)$$

and for all $d \in [0, 1]$ there exists $\Gamma(d)$ such that

$$\Gamma(0) = \mathcal{R}_i(d) \text{ and } \Gamma(1) \in \{\widehat{\xi_1}, \ldots, \widehat{\xi_m}\}, \quad (4)$$

where $\mathcal{L}(\mathcal{R}_i)$ denotes the length of the i-th trajectory segment and $\widehat{\xi_j}$ denotes configuration above the touchdown configuration ξ_j of the corresponding landing site, i.e., it is allowed the vehicle is above the landing site. The existence of the safe emergency landing trajectory at any point of the final trajectory $\mathcal{R}$ is assured by (4).

[1] It is assumed the vehicle can decrease its altitude easily by a specific maneuver type even in the case of LoT [2].

IV. Proposed RRT*-based solution

The introduced ELASP is a challenging problem because it includes a solution of three sub-problems that are solved separately in the literature. Therefore, we propose to address the ELASP problem in the following five steps.

1) A dense graph of possible emergency landing trajectories is generated by Algorithm 1 to support fast queries for the minimum safe altitude of the vehicle at any location of the planning area.
2) Based on the generated graph, minimum altitude profiles of all possible connections between points of interest S are computed in Algorithm 2.
3) The altitude profiles are modified such that their inclination meets the minimum/maximum pitch angle constraint, and thus feasible trajectories between points S are found.
4) The length of the found trajectories is used as the travel cost to determine the sequence of visits to S using an existing TSP-like solver.
5) For the determined sequence of visits, the corresponding trajectories are connected into the final trajectory, and any altitude discontinuities are removed if occur.

The method for constructing the dense graph of possible emergency landing trajectories is adopted from [2] where the RRT*-based algorithm is employed to find a landing trajectory with the minimum altitude loss. The main idea of [2] is growing a tree of possible landing trajectories for each landing site, and thus a forest of the individual trees forms the graph of landing trajectories. The graph supports fast queries for the minimum safe altitude; however, in contrast to [2] where informed sampling is utilized, in the herein addressed ELASP, the location of the vehicle is not known at this stage of the graph generation. Therefore, informed sampling is replaced by the uniform sampling in Algorithm 1.

Utilizing model of the gliding trajectory proposed in [2], the configuration space of the vehicle is simplified to the 2D position (x, y) and the corresponding heading angle θ. Thus, the simplified configuration is $\tilde{q} = (x, y, \theta) \in SE(2)$, i.e., the simplified configuration space is $\tilde{\mathcal{C}} = SE(2)$, which significantly reduces the computational burden. The altitude of $\tilde{q}$ is considered as the minimum altitude for the safe emergency landing that is denoted as the function $\mathcal{A} : \tilde{\mathcal{C}} \to \mathbb{R}$. The particular minimum altitude $\mathcal{A}$ is influenced by the altitude of the selected landing site, altitude loss $\mathcal{H} : \Gamma \to \mathbb{R}$ of the corresponding landing trajectory Γ, and the altitude of the terrain.

In each iteration of Algorithm 1, a random configuration is uniformly sampled over the whole planning area. Then, the nearest configuration in the current graph G is found

Algorithm 1: RRT*-based planning of possible emergency landing trajectories (adopted from [2])

```
Input: Ξ = {ξ_1, ..., ξ_m} – Set of the landing sites
Input: T_alt – Altitude of the terrain (or obstacles)
Output: G – Graph of emergency landing trajectories
Output: A – Minimum safe altitudes for graph nodes
1  G ← {V ← Ξ, E ← ∅}
2  A(ξ_i) ← T_alt(ξ_i), ∀ξ_i ∈ Ξ
3  while t < t_plan do
4      q̃_rand ← SampleUniform()
5      q̃_nearest ← Nearest(q̃_rand, G)
6      q̃_new ← Steer(q̃_nearest, q̃_rand)
7      Q_n ← Near(q̃_new, G)
8      q̃_* ← argmin_{q̃_n ∈ Q_n} [A(q̃_n) + H(q̃_new, q̃_n)]
9      A(q̃_new) ←
10             max [T_alt(q̃_*, q̃_new), A(q̃_*) + H(q̃_new, q̃_*)]
11     V ← V ∪ {q̃_new}; E ← E ∪ {(q̃_*, q̃_new)}
12     G ← Rewire(Q_n, G)
```

by Nearest(), and a connection to the randomly selected node is determined by 2D Dubins maneuver in the Steer() procedure. Further, a part of the trajectory longer than the steer constant is cut off, and a new configuration at the maneuver's end is returned (Line 6, Algorithm 1). Possible parent nodes for the new configuration $\tilde{q}_{new}$ are determined by Near(). The best parent node is selected according to the overall altitude loss to the landing site. Finally, an attempt to optimize the graph G is performed in Rewire() by re-connecting existing edges to lower the minimum safe altitude. The expansion of the graph G is repeated for the dedicated computational time t_{plan}.

Once the graph G of the emergency landing trajectories is constructed, trajectories connecting the locations S are determined together with their minimum safe altitude profiles. Finding a smooth trajectory connecting S is addressed by the sampling-based method [13]. Possible heading angles are discretized using k uniformly selected samples per each location $s_i \in S$ and $\{q_i^1, \ldots, q_i^k\}$ denote the created configurations for the location s_i. The created configurations are connected using the Dubins Airplane model [7], and thus $k^2(n^2 - n)$ trajectories are determined.

Each trajectory is further sampled using selected sampling step d_{step}, i.e., the trajectory segment $\mathcal{R}_i$ is sampled to $\tilde{Q}_i = \{\tilde{q}_i^1, \tilde{q}_i^2, \ldots, \tilde{q}_i^{\text{end}}\}$. A distance in the 2D plane, denoted as d, is lower than the sampling step for any two consecutive samples, $d_i^{j,j+1} = d(\tilde{q}_i^j, \tilde{q}_i^{j+1}) \le d_{\text{step}}$. For each sample, an emergency landing trajectory is determined using Algorithm 2 which determines the minimum landing altitude $\mathcal{A}$ for the generated samples.

The proposed algorithm should guarantee a safe landing possibility from any point of the trajectory, and not only from the selected samples. Thus, a safe altitude $\mathcal{A}_{\text{safe}}$ is determined such that the altitude loss $\mathcal{H}$ of the segment between two samples is added to the minimum altitude of the next samples. Further, the minimum/maximum pitch angle

Algorithm 2: Retrieve the minimum safe altitude

```
Input: G – Graph of emergency landing trajectories
Input: A – Minimum safe altitudes for graph nodes
Input: q̃_act – Simplified configuration to query
Output: A(q̃_act) – Minimum safe altitude
Q_n ← Near(q̃_act, G)
q̃_best ← argmin_{q̃_n ∈ Q_n} [A(q̃_n) + H(q̃_act, q̃_n)]
A(q̃_act) ← max [T_alt(q̃_*, q̃_act), A(q̃_*) + H(q̃_act, q̃_*)]
```

$\psi_{\text{min/max}}$ is utilized to limit a slope of the trajectory. The complete expression for the safe altitude $\mathcal{A}_{\text{safe}}$ is

$$\mathcal{A}_{\text{safe}}\left(\tilde{q}_i^j\right) = \max \begin{bmatrix} \mathcal{A}\left(\tilde{q}_i^{j+1}\right) + \mathcal{H}\left(\tilde{q}_i^j, \tilde{q}_i^{j+1}\right) \\ \mathcal{A}_{\text{safe}}\left(\tilde{q}_i^{j-1}\right) + \tan(\psi_{\min})\, d_i^{j-1,j} \\ \mathcal{A}_{\text{safe}}\left(\tilde{q}_i^{j+1}\right) - \tan(\psi_{\max})\, d_i^{j,j+1} \end{bmatrix}. \quad (5)$$

One may notice that the expression (5) is recursive, and thus the altitude is increased in two passes through all samples: (i) forward for limiting the maximum descent by the minimum allowed pitch angle; and (ii) backward for limiting the ascend by the maximum pitch angle.

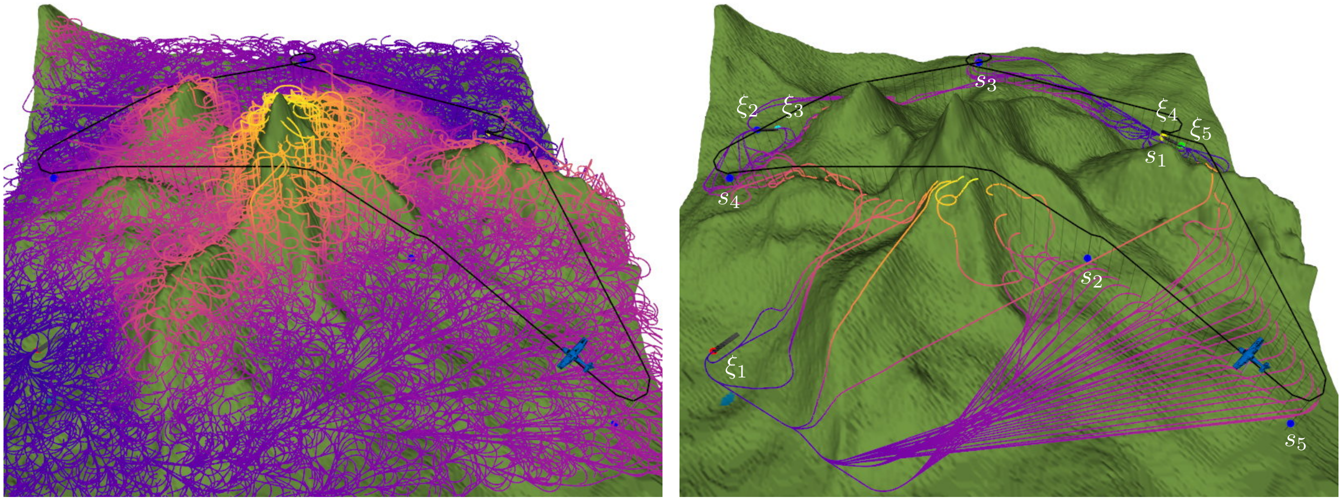
The determined safe altitude $\mathcal{A}_{\text{safe}}$ guarantees safe landing, but the total trajectory length is not minimized as in the ELASP formulation (2). Therefore, the final altitude profile $\mathcal{A}_{\text{F}}$ is required to have a concave property which holds for any sample except the end ones corresponding to the given locations. Otherwise, the altitude profile is not smooth and is prolonged by unnecessary altitude changes. The concavity constraint can be expressed based on the lengths of two consecutive segments $d_i^{j-1,j}$ and $d_i^{j,j+1}$ as

$$\mathcal{A}_{\text{F}}\left(\tilde{q}_i^j\right) \ge \frac{d_i^{j,j+1} \mathcal{A}_{\text{safe}}\left(\tilde{q}_i^{j-1}\right) + d_i^{j-1,j} \mathcal{A}_{\text{safe}}\left(\tilde{q}_i^{j+1}\right)}{d_i^{j-1,j} + d_i^{j,j+1}}. \quad (6)$$

Similarly to the previous case, the final altitude profile is consecutively determined by two-pass algorithm which check the concavity forward and backward.

The fourth step of the proposed ELASP approach is a solution of the sequencing part, and the computed feasible trajectories connecting the visiting configurations are utilized to determine a closed-loop trajectory to visit all the locations S. The sequencing problem is formulated as the Generalized Asymmetric TSP (GATSP), where the sets are defined by the k configurations for each location, and distances are the lengths of the feasible trajectories determined in the previous step. The created instance of the GATSP is transformed to the Asymmetric TSP using Noon-Bean transformation [14] and the sequence is determined optimally using Concorde [24], or LKH Solver [15] is employed for a faster heuristic solution.

The final trajectory is created by concatenating the individual trajectories according to the sequence determined as the GATSP solution. However, a straightforward concatenation may contain discontinuities because of altitude profiles modified to meet the pitch angle constraints. Hence, the trajectory ends may have a higher altitude than the original locations of the interest S, and thus the same procedure for increasing



(a) Pre-generated graph of possible landing trajectories

(b) Emergency landing trajectories for the final trajectory

Fig. 2. Visualization of the graph (left) of possible emergency landing trajectories colored based on the altitude and final trajectory (right) in black that connects the locations $s_i \in S$ with selected landing trajectories to the landing sites ξ_j.

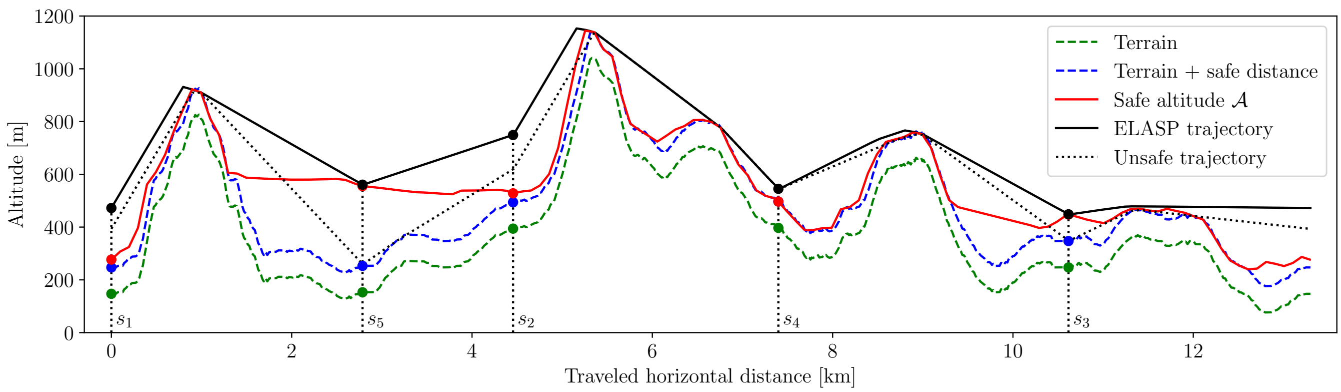


Fig. 3. Comparison of altitude profiles generated by the proposed ELASP algorithm and *Unsafe planning* considering only the pitch angle limits and the concavity constraint. The highlighted points correspond to the locations $s_i \in S$, and the minimum altitude for safety landing is also shown.

the altitude of the samples, as in the third step, is utilized to get the final feasible trajectory.

As a result, the proposed algorithm guarantees that the final trajectory is closed-loop, visits all the given points of interest, meets the pitch angle constraints, and it is possible to land from any point of the trajectory safely.

V. Results

The proposed solution to the introduced ELASP problem is empirically evaluated on a scenario with the mountain terrain, which is visualized in Fig. 2b. The considered planning area is 5×5 km large and the terrain altitude ranges from $[0, 1500]$ meters above the sea level. Two bidirectional and a one single-side landing sites are placed at the lower part of the terrain. Thus, emergency landing trajectories are selected from 5 possible landing sites, i.e., $m = 5$.

The considered vehicle is Cessna 172 because it represents one of the most popular general aviation aircraft. The minimum turning radius of the vehicle is about 65 m and the constant forward velocity is assumed to be $33.4\,\mathrm{ms}^{-1}$. A complex model of the altitude loss during gliding has been adopted from [2], where the minimum angle of the emergency descent is about 4.9°.

The considered ELASP scenario consists of $n = 5$ locations of interest that are 100 m above the terrain. The number of considered heading angles is $k = 4$ and the generated trajectory segments are uniformly sampled with the step size 100 m. The emergency landing trajectory is found for each of the samples and the trajectory influences the altitude of the final trajectory. Further, the necessary altitude for safe landing is increased by 100 m as the selected safety distance above the terrain.

The determined graph of possible landing trajectories is pre-computed and queried to speed up the planning process. The landing trajectories are visualized in Fig. 2a, where the color represents an altitude to highlight costly trajectories. The final planned trajectory is shown in Fig. 2b.

The altitudes for the final found trajectory are further visualized in Fig. 3 to demonstrate a behavior of the proposed algorithm. An altitude of the final trajectory is significantly influenced by the highest points of the terrain. The constraints on the pitch angle are applied, and thus, the maximum slope around the highest points is limited. This causes that the locations s_2 cannot be visited in the requested altitude and the altitude of the final trajectory is significantly increased. The altitude in s_2 is further increased by the requirement for the safe landing because the highest point in the terrain needs to be reached from an even higher altitude to provide the option to over-fly the mountain safely if there is not enough space to turn back.

The final altitude change by the emergency landing planning is mostly influenced at the location s_5 (bottom right in Fig. 2b). Here, the altitude is not affected by the pitch angle constraints nor the terrain, but mostly by the distance to the closest landing site. A similar case can be seen even for the location s_3 with a much lower altitude increase.

TABLE I

COMPUTATIONAL REQUIREMENTS OF THE PROPOSED ELASP SOLUTION

Step	T_{CPU}
1. Graph generation	60 s
2. Compute landing trajectories	131 s
3. Limits the pitch angle	less than 1 ms
4. Solving the GATSP	80 ms
5. Limits the pitch angle	less than 1 ms
Total	191 s

The proposed ELASP algorithm has been implemented in C++ and executed on a single core of the Intel Core i5-7600K running at up to 4.2 GHz. The real computational requirements for the considered mountain scenario are depicted in Table I. The most computationally demanding part is the determination of the landing trajectories because the trajectories are generated for any possible connection between the given locations and sampled heading angles.

VI. CONCLUSION

We propose a solution of the newly introduced ELASP problem that combines finding a feasible multi-goal trajectory to visit a set of locations with the guarantee of safe emergency landing from any point of the trajectory for the case of unexpected loss of thrust. The proposed algorithm leverages on the pre-computed graph of possible landing trajectories within the expected planning area that is later utilized during the multi-goal planning. The proposed algorithmic solution has been empirically evaluated in the mountain scenario to demonstrate the difficulty of the ELASP problem and the effect of the pre-computed landing trajectories. There can be identified several possible directions for further improvement. Beside a possible decrease of the computational requirements by planning landing trajectories only for the most promising areas of the search spaces, we plan to generalize the model of the vehicle for more complex trajectories, and thus further optimize the final length of the trajectory.

REFERENCES

[1] S. Edelkamp and E. Plaku, "Multi-goal motion planning with physics-based game engines," in *IEEE Conference on Computational Intelligence and Games*, 2014, pp. 1–8.

[2] 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.

[3] J. Faigl, P. Váňa, R. Pěnička, and M. Saska, "Unsupervised learning-based flexible framework for surveillance planning with aerial vehicles," *Journal of Field Robotics*, vol. 36, no. 1, pp. 270–301, 2019.

[4] 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*, pp. 497–516, 1957.

[5] H. J. Sussmann, "Shortest 3-dimensional paths with a prescribed curvature bound," in *34th IEEE CDC*, vol. 4, 1995, pp. 3306–3312.

[6] G. Ambrosino, M. Ariola, U. Ciniglio, F. Corraro, A. Pironti, and M. Virgilio, "Algorithms for 3d uav path generation and tracking," in *45th IEEE CDC*, 2006, pp. 5275–5280.

[7] H. Chitsaz and S. M. LaValle, "Time-optimal paths for a dubins airplane," in *46th IEEE CDC*, 2007, pp. 2379–2384.

[8] M. Owen, R. W. Beard, and T. W. McLain, "Implementing dubins airplane paths on fixed-wing uavs," in *Handbook of Unmanned Aerial Vehicles*. Springer, 2015, pp. 1677–1701.

[9] A. A. Neto, D. G. Macharet, and M. F. Campos, "3d path planning with continuous bounded curvature and pitch angle profiles using 7th order curves," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 4923–4928.

[10] K. Savla, E. Frazzoli, and F. Bullo, "On the point-to-point and traveling salesperson problems for dubins' vehicle," in *American Control Conference*. IEEE, 2005, pp. 786–791.

[11] P. Váňa, J. Sláma, and J. Faigl, "The dubins traveling salesman problem with neighborhoods in the three-dimensional space," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 374–379.

[12] D. G. Macharet and M. F. M. Campos, "A survey on routing problems and robotic systems," *Robotica*, vol. 36, no. 12, pp. 1781–1803, 2018.

[13] K. Obermeyer, P. Oberlin, and S. Darbha, "Sampling-based roadmap methods for a visual reconnaissance uav," in *AIAA Guidance, Navigation, and Control Conference*, 2010, p. 7568.

[14] C. Noon and J. Bean, "An Efficient Transformation of the Generalized Traveling Salesman Problem," Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI, USA, Tech. Rep. 91–26, 1991.

[15] K. Helsgaun, "An Effective Implementation of the Lin-Kernighan Traveling Salesman Heuristic," *European Journal of Operational Research*, vol. 126, no. 1, pp. 106–130, 2000.

[16] X. Yu and J. Hung, "A genetic algorithm for the dubins traveling salesman problem," in *IEEE International Symposium on Industrial Electronics*, 2012, pp. 1256–1261.

[17] X. Zhang, J. Chen, B. Xin, and Z. Peng, "A memetic algorithm for path planning of curvature-constrained uavs performing surveillance of multiple ground targets," *Chinese Journal of Aeronautics*, vol. 27, no. 3, pp. 622–633, 2014.

[18] J. Faigl and P. Váňa, "Self-organizing map for the curvature-constrained traveling salesman problem," in *International Conference on Artificial Neural Networks (ICANN)*, 2016, pp. 497–505.

[19] S. Paul, F. Hole, A. Zytek, and C. A. Varela, "Flight Trajectory Planning for Fixed-Wing Aircraft in Loss of Thrust Emergencies," *arXiv preprint arXiv:1711.00716*, 2017.

[20] N. Meuleau, C. Plaunt, D. Smith, and T. Smith, "Emergency landing planning for damaged aircraft," in *Innovative Applications of Artificial Intelligence Conference*, 2009, pp. 3247–3259.

[21] P. Eng, "Path planning, guidance and control for a uav forced landing," Ph.D. dissertation, Queensland University of Technology, 2011.

[22] S. Paul, F. Hole, A. Zytek, and C. A. Varela, "Wind-aware trajectory planning for fixed-wing aircraft in loss of thrust emergencies," in *IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE, 2018, pp. 1–10.

[23] B. Ayhan, C. Kwan, B. Budavari, J. Larkin, and D. Gribben, "Preflight contingency planning approach for fixed wing uavs with engine failure in the presence of winds," *Sensors*, vol. 19, no. 2, p. 227, 2019.

[24] D. Applegate, R. Bixby, V. Chvátal, W. Cook, D. Espioza, M. Goycoolea, and K. Helsgaun, "Concorde TSP Solver," 2003, [cited 2 May 2019]. [Online]. Available: https://www.tsp.gatech.edu/concorde.html