Optimal Flight Paths for Engine-Out Emergency Landing

Avishai Adler[1], Aharon Bar-Gill[2], Nahum Shimkin[3]

1. Department of Electrical Engineering, Technion, Haifa 32000, Israel

2. Department of Aerospace Engineering, Technion, Haifa 32000, Israel
E-mail: aerabg@techunix.technion.ac.il

3. Department of Electrical Engineering, Technion, Haifa 32000, Israel
E-mail: shimkin@ee.technion.ac.il

Abstract: Loss of engine power constitutes a major emergency situation in General Aviation (GA) aircraft, requiring the location of a safe-to-land strip within reach, and thereupon planning and executing an effective gliding path towards it. These critical tasks are currently entrusted with the pilot. In recent years, technological advances in avionics (GPS, GIS and computing capabilities) have reached the GA cockpit – clearing the way for safety enhancements that utilize these resources. In this paper we consider the problem of 3D trajectory planning for an engine-cut GA aircraft towards a specified airstrip, while avoiding natural or man-made obstacles. We emphasize energy efficiency, which allows the aircraft to maximally extend its reach. To that end we employ a dynamic model of the aircraft, which leads to a six-dimensional optimal control problem. We propose a computation approach that is aimed at approximating the globally optimal solution in real-time. Our approach is based on *motion primitives*, or basic maneuvers, which are parameterized flight segments of specified shapes which are locally optimized for energy efficiency. These basic maneuvers enable a coarse discretization of the search space, and the planning problem is reduced to a graph-search problem of tractable size which may be efficiently solved using an optimal graph search algorithm. Important computational enhancements include the use of pre-compiled basic maneuver dictionaries. The effectiveness of the proposed solution is demonstrated via simulation results.

Key Words: Aircraft trajectory planning, dynamic constraints, motion primitives, engine-off emergency

1 Introduction

General Aviation (GA) aircraft constitute a major part of todays air traffic. These aircraft are often prone to emergency situations; there are more than 1400 reported GA accidents per year in the US alone [11]. Effective response in such emergencies is of major importance in enhancing flight safety. In recent years, advances in avionics (including GPS, GIS/DTM, on-board computing power) are increasingly found in the GA cockpit, clearing the way for safety enhancements and aid pilot decisions and planning

In this work we address a common emergency in GA - midair engine cut. This is a particularly hazardous situation for GA since most aircraft are equipped with a single engine1 . In this situation a flight path to an available safe-landing strip need be planned. Often such emergencies occur under SPIFR (Single Pilot Instrument Flight Rule) conditions, yielding the situation even more stressful. We propose an automated path planning algorithm that generates in a matter of seconds an optimized trajectory to be followed by the pilot to safe landing. Such a trajectory should utilize to the utmost the aircrafts available energy (kinetic and potential) to increase the reachable flight range under power-out conditions.

Three-dimensional (3D) flight path generation is a computationally demanding task. A body of methods use simplified kinematic modeling of the aircraft (see, e.g., [13]). In an engine-out scenario, such simplified models cannot adequately account for the maneuverability constraints imposed by lack of engine thrust, nor do they provide an accurate assessment of the aircraft energy loss associated with different flight maneuvers. Therefore, in this work our starting point is a more elaborate dynamic model that accounts for these effects. This model has six state variables, with the pair α and ϕ (the aircraft angle of attack and roll angle) serving as the control variables.

Given the aircraft dynamic model, we may formulate our planning problem as an optimal control problem. We consider the objective of reaching given map point (latitude and longitude coordinates) with given velocity direction, while minimizing the energy loss along the trajectory. Any known obstacles should evidently be avoided, which yields the problem highly non-convex. Our goal is to find the globally optimal solution within seconds, which is out of reach of standard numerical methods.

Our approach is based on the notion of trajectory primitives, which are combined to form basic maneuvers. The trajectory primitives are flight segments of specified shape, locally optimized for energy efficiency. The three trajectory primitives we employ are straight flight, gliding and turning. Basic maneuvers are composed of several trajectory primitives, in a specified order, so as to bring the airplane from one state (comprised of position, orientation and velocity) to the other. The remaining challenge is then to find an optimal combination of these basic maneuvers, leading to landing destination. To that end, we discretize

the state space, and construct a graph in which the nodes are the discrete states, and the edges that connect them are suitable basic maneuvers. The shortest path on this graph corresponds to the flight path with minimal energy loss, and may be found using the optimality-proven and time-efficient graph-search algorithm by Dijkstra, along with some problem-specific enhancements. The use of basic maneuvers greatly reduces the search space, inherently resulting in flyable paths. Furthermore, these resulting flight paths are composed of a few path primitives that have a clear interpretation (such as turn at a certain radius, optimal glide for a certain distance, etc.), assuring ease of following of trajectory, generated via our novel algorithms.

2 Related Work

2.1 Trajectory Optimization

Trajectory planning can be formulated as an optimal control problem, with the vehicle kinematics and dynamics expressed by the state equations, and a cost functional that may capture the time to goal, path length, control effort, etc. Obstacles and exclusion zones may be represented as constraints on the positional state variables. A range of computational methods is available for the numeric solution of optimal control problems; see, e.g., [4, 3]. In the so-called indirect methods, the TPBVP induced by the minimum principle, is solved numerically. Direct methods constitute an alternative, where the continuous time problem is first approximated by a finite dimensional optimization problem, which may be solved using existing non-linear programming (NLP) algorithms. However, their convergence to global optimum is not guaranteed for highly nonconvex problems (as is the problem treated here, especially in the presence of obstacles).

Dynamic Programming (DP) provides an alternative solution, assuring globally optimality via an exhaustive search in the state space. Consequently, DP methodology [8] suffers from the notorious curse of dimensionality, so that a high precision numerical solution becomes impractical for problems with more than a few state dimensions.

The approach we pursue here uses a single-stage graph search formulation, which relies on the notion of trajectory primitives for coarse-scale problem discretization. The obtained graph-search problem is, solved using the optimal Dijkstra graph search algorithm. We then employ the Dubins motion primitives (minimum radius turns and straight line segments), which are commonly used in path planning problems (as building blocks in shortest path ones: for related aeronautical applications see for example [12]. The trajectory primitives used in the present work are naturally optimized for energy efficiency.

Our previous work in [13] also relied on a finite graph formulation, in order to find an optimal three-dimensional (3D) flight path that minimizes a combinations of additive costs, such as flight duration, piloting workload and riding qualities. It employed a simple kinematic model, since thrust availability allows to maintain constant- airspeed flight.

2.2 Flight Trajectory Planning for Gliding Aircraft

Rogers [14] deals with engine-cut emergency, which occurs shortly after lift-off. His solution is based on a simplified analytical model, while the velocity and turn rate are constant. Rademacher et al. [12] confine themselves to 2D and employ the NLP approach. Finding the maximum range is addressed in Shapira and Ben-Asher [17]. Their solution involves time-scale separations, to decouple the dynamic equations, and allow for an analytical solution. However, this solution involves extensive approximation, and would not handle obstacles. Similarly, [1] proposes an analytical open-loop solution, using the Maximum Principle. Yong et al. [19] constitutes an additional example of this type of formulation. Ben-Asher - with Dekel [6] - revisit the maximum range optimal glide problem, looking at the engine cut-off emergency, same as we do hereafter. They study the Pseudo Spectral numeric method. However, the actual engine cut-off emergency involves real-life non-convexities such as natural relief or man-made obstacles, which hamper the applicability of this approach as a global solution for the full problem.

3 Problem Formulation

The problem we consider is that of generating a flyable trajectory between an origin point (that includes the aircraft position, orientation and velocity) and a destination point, while minimizing the en-route energy loss.

3.1 Nonlinear Aircraft Model

Consider an Inertial Reference System, I, which is a right-handed coordinate system that coincides with the local east-north-up (ENU) coordinates, with axis X_I pointing to the east, Y_I to the north and Z_I pointing upwards. Let X, Y, Z denote the respective aircraft position in these coordinates. The following set of equations may then be used to describe the three-dimensional motion for a gliding aircraft over a flat Earth [9]:

$$
\begin{align}
\dot{X} &= V \cos\gamma \cos\xi \tag{1} \\
\dot{Y} &= V \cos\gamma \sin\xi \tag{2} \\
\dot{Z} &= V \sin\gamma \tag{3} \\
\dot{V} &= -\frac{D(\alpha, V)}{m} - g \sin\gamma \tag{4} \\
\dot{\gamma} &= \frac{L(\alpha, V) \cos\phi}{mV} - \frac{g}{V} \cos\gamma \tag{5} \\
\dot{\xi} &= \frac{L(\alpha, V) \sin\phi}{mV \cos\gamma} \tag{6}
\end{align}
$$

Here V is the airspeed of the aircraft, γ the vertical flight path angle (positive values are above the horizon), ξ is the heading angle (azimuth), α is the angle of attack (AOA), and ϕ is the roll angle. L and D are the lift and drag forces, which may be expressed in the standard form

$$
L(\alpha, V) = \frac{1}{2} SV^2 C_L(\alpha), \ D(\alpha, V) = \frac{1}{2} \rho SV^2 C_D(\alpha) \tag{7}
$$

where $C_L(\alpha)$ and $C_D(\alpha)$ are the lift and drag coefficients, ρ the air density and S the wing area. This model assumes coordinated turns, with no sideslip.

These equations describe the aircraft's dynamics, using the six-dimensional state vector $x = [X, Y, Z, V, \gamma, \xi]^T$, and control vector $u = [\alpha, \phi]^T$. We note that α and ϕ may are easily governed by the pilot on a relatively short time scale, and therefore qualify as the controls in our model.

3.2 Flight Envelope Constraints

The aircraft motion variables are subject to physical and safety constraints that affect its maneuverability, and should be taken into account in path planning. The main constraints to be considered are the following:

$$
\begin{array}{ll}
a_{vert}^{min} \leq \dot{\gamma}V \leq a_{vert}^{max}, & a_{horiz}^{min} \leq \dot{\xi}V \leq a_{horiz}^{max} \\
V_{min} \leq V \leq V_{max}, & \gamma_{min} \leq \gamma \leq \gamma_{max} \\
\phi_{min} \leq \phi \leq \phi_{max}, & \alpha_{min} \leq \alpha \leq \alpha_{max}
\end{array} \tag{8}
$$

The structural load limitation constrains the acceleration components $\dot{\gamma}V$ and $\dot{\xi}V$, and maximum-allowable velocity V [5]. Stall avoidance yields bounds on the minimal value of velocity V and the maximum absolute value of the AOA. Constraining γ and ϕ keeps the aircraft within its performance envelope.
Additional constraints apply to the position coordinated (X, Y, Z). Typically these constraints will be in the form

$$
Z \geq h_{min}(X, Y) \tag{9}
$$

where $h_{min}(X, Y)$ represents safe distance above ground (extracted from a Digital Terrain Map). No-fly zones (involving the X-Y coordinates only) can be represented in the same way, with h_{min} set to a high enough value there.

3.3 Objectives

Let $x(t_0) = x_0$ denote the initial (current) state vector. We wish to find a feasible flight-path $x(t)$, $t \in [t_0, t_f]$ that ends up in a required end-set $\mathcal{X}_f$, namely $x(t_f) \in \mathcal{X}_f$, with the final time t_f a free parameter. The set $\mathcal{X}_f$ should allow an approach to the landing strip in a direction and velocity fit for landing.
Let $E(t)$ denote the aircraft total energy (kinetic and potential), and $\dot{E}$ its time derivative. Thus, essentially, $E = \frac{1}{2}mV_g^2 + mgZ$. Here V_g is the aircraft ground speed, which equals V if the wind effect is small, and otherwise can be computed as $(\dot{X}^2 + \dot{Y}^2 + \dot{Z}^2)^{1/2}$ using (1)–(3). Evidently, the total energy is a function of the current state. Our goal can now be stated as follows. Maximize the terminal energy $E(t_f) = E(x(t_f))$, or equivalently minimize the exhausted energy

$$
J = -\int_{t_0}^{t_f} \dot{E}(x(t), u(t))dt \tag{10}
$$

subject to the dynamic model (1)–(6), static constraints (8)–(9), and terminal conditions $x(t_f) \in \mathcal{X}_f$.
Observe that if $\mathcal{X}_f$ is a singleton, i.e., the terminal state is fully specified, then the terminal energy $E(t_f)$ is specified as well, and the optimization problem reduces to determining the feasibility of the required path. Furthermore, in case the problem is feasible there will generally be multiple solutions, and the recommended path will need to be selected based on secondary criteria (shortest path, riding qualities, etc.).

To keep the focus on the critical aspect of energy conservation, we choose here to leave the terminal altitude $Z(t_f)$ as a free parameter. Thus, $\mathcal{X}_f$ is defined in terms of the other state variables $(X, Y, Z, V, \gamma, \xi)$, specifying an approach to the landing strip at given orientation and velocity, but with unspecified altitude. Clearly, then, the problem of minimizing the energy loss becomes equivalent to maximizing the terminal altitude. It is assumed that the pilot will be able to get rid of the excess energy once in the vicinity of the landing strip. Conserving the excess energy till the last phases has the important advantage that it allows to correct for miscalculations, pilot errors and unforeseen wind conditions that might occur en-route and exhaust the aircraft energy reserves.

4 Trajectory Primitives and Basic Maneuvers

The basic maneuvers constitute the building blocks for the trajectory construction and optimization. These maneuvers are designed to bring the aircraft from one state (position, orientation and velocity) to the other. This enables a coarse discretization of the state space and a resulting sparse graph for optimal search purposes. These basic maneuvers are constructed using several segments of trajectory primitives, in a pre-specified order. They are pre-defined maneuvers that include straight flight (with possible velocity adjustment), optimal glide, and constant-rate turns - fliable and common, yielding both the flight trajectory and admissible control sequences to follow it.

4.1 Gliding Primitive

Straight-line gliding at a constant speed is a basic aero maneuver that is widely discussed in the aeronautics literature [15]. In addition to speed, both the heading angle ξ and the flight path angle γ are kept constant. The gliding constraints are therefore $\dot{\gamma} = 0$, $\dot{\xi} = 0$, $\dot{V} = 0$.
Applying these equalities to the flight dynamics equations, we first obtain from equation (6) that $\phi = 0$. Equations (4) and (5) now imply $D = -mg \sin \gamma$, $L = mg \cos \gamma$, and together with (7) we obtain

$$
\tan \gamma = -C_D(\alpha)/C_L(\alpha) \tag{11}
$$

$$
C_D(\alpha)^2 + C_L(\alpha)^2 = \left(\frac{mg}{\frac{1}{2}\rho S V^2}\right)^2 \tag{12}
$$

For a given glide velocity V, the corresponding AOA α can be obtained from the last equation (or vice-versa), and γ from the previous one. Note that α, which serves as control, is also fixed here. It is evident from (11) that γ is negative, so that the aircraft is loosing height (or potential energy) due to drag, while maintaining its velocity (or kinetic energy).
Maximal-range gliding: Revisiting equation (12), we can see that

$$
\frac{\text{Distance}}{\text{Altitude Loss}} = -\frac{1}{\tan \gamma} = \frac{L}{D} = \frac{C_L(\alpha)}{C_D(\alpha)} \tag{13}
$$

That is, the horizontal distance traveled per unit loss in altitude is dictated by the L/D-ratio (lift-to-drag ratio). Figure 1 depicts the typical behavior of the L/D-ratio for

2012 24th Chinese Control and Decision Conference (CCDC)

any aircraft – a curve with a distinct maximum, known as $(L/D)_{max}$. This optimal ratio is obtained at a certain velocity $V_{(L/D)max}$, the optimal-glide speed which can be computed from the above equations, and is further usually quoted in the Pilots Operating Handbook (POH), along with $(L/D)_{max}$.
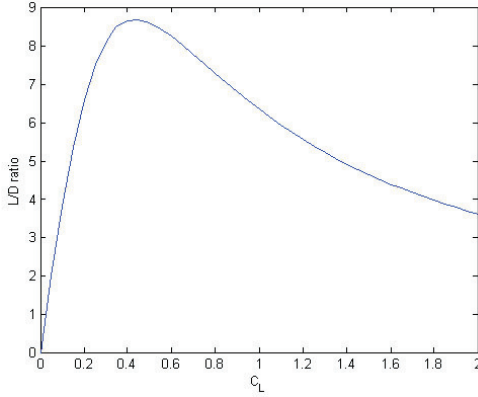

Figure 1: Typical relation between C_L and L/D.

Gliding at $(L/D)_{max}$ is the optimal maneuver for maximum range flight [1]. Equivalently, it entails the minimum energy loss per distance traveled. We therefore adopt it in the following as the basic trajectory primitive for fixed-heading flight.

4.2 Velocity-Adjustment Primitive

Adjustment of the aircraft speed will be required before and after turns, and possibly in the initial and final phases of the power-off flight. We will use a straight-flight trajectory primitive for that purpose, again with a constant flight path angle γ, tracing a straight path in the vertical plane. The following constraints therefore define the straight-flight primitive: $\dot{\gamma} = 0, \dot{\xi} = 0$.

As before, $\dot{\xi} = 0$ implies $\phi = 0$ by equation (6). Further, equation (5), together with (7), implies

$$\dot{\gamma} = \frac{\frac{1}{2}\rho SV^2 C_L(\alpha)cos\phi}{mV} - \frac{g}{V}\cos\gamma \qquad (14)$$

Therefore, by the straight-flight constraints we get:

$$C_L(\alpha) = \frac{mg\cos\gamma}{\frac{1}{2}\rho SV^2} \qquad (15)$$

The control α can be derived from this equality as a function of V, using the airplane-specific model for $C_L(\alpha)$. The time-varying velocity profile $V(t)$ can now be found by integrating equation (4).

According to the choice of γ, the straight flight primitive may correspond to fixed altitude flight ($\gamma = 0$), descending ($\gamma > 0$) or climbing ($\gamma < 0$). The velocity will change accordingly. For example, during climb the aircraft will reduce its velocity while gaining potential energy.

Optimal Velocity Adjustment: Suppose we wish to increase or decrease the aircraft velocity from an initial velocity V_1 to a final velocity V_2. For this purpose we will use the straight-flight primitive that exhausts the least energy. Since the initial and final velocities are given, this optimal maneuver is the one with minimal descent. The solution can be found numerically by iterating over the vertical flight path angle γ. This calculation can be done off-line, resulting in a short table of optimal γ values for velocity pairs (V_1, V_2) in the relevant range.

4.3 Constant-Rate Turning Primitive

Turning is of course required for changing the initial and final aircraft heading, and possibly for en-route obstacle avoidance. Our third trajectory primitive enables change in heading (ξ), while yielding a helix-like trajectory (circles in the horizontal plane, with constant rate of descent). The basic formulation is based on Rogers [14], in which an optimal turn was derived based on basic approximation to the flight dynamics and energy considerations.

We assume a coordinated turn (with no sideslip), in which the velocity V, the flight path angle γ and the turning rate $\dot{\xi}$ are constants (yielding a constant-radius turn). The corresponding constraints are: $\dot{V} = 0, \dot{\gamma} = 0, \dot{\xi} =$ constant.

Applying these constraints to the flight dynamics equations (4)–(6) yields the following set of equations:

$$0 = -D - mg\sin\gamma \qquad (16)$$
$$0 = L\cos\phi - mg\cos\gamma \qquad (17)$$

Suppose we parameterize the set of turns by the pair (V, γ). The required controls α, ξ can be calculated as follows. By equations (16) and equation (7),

$$C_D(\alpha) = \frac{mg\sin\gamma}{\frac{1}{2}\rho SV^2} \qquad (18)$$

from which α can be extracted. Now, from equation (17),

$$\cos\phi = \frac{mg\cos\gamma}{L(\alpha)} = \frac{mg\cos\gamma}{\frac{1}{2}\rho SC_L(\alpha)V^2} \qquad (19)$$

from which the absolute value of the ϕ can be found. The sign of ϕ determines whether the maneuver is a right or a left turn.

The turn rate $\dot{\xi}$ can now be computed using equation (6). The horizontal *turn radius* R can now be computed as

$$R_{turn} = \frac{V\cos\gamma}{\dot{\xi}} = \frac{m(V\cos\gamma)^2}{L(\alpha)\sin\phi} = \frac{m\cos^2\gamma}{\frac{1}{2}\rho SC_L(\alpha)\sin\phi} \qquad (20)$$

Optimal turn: We will choose the turn parameters so as to minimize the total energy loss during the turn. Assuming that the required change $\Delta\xi$ in azimuth is given, we wish to minimize $dE/d\xi$. As the kinetic energy does not change during this constant-velocity maneuver, this is equivalent to minimizing $|dZ/d\xi|$. The latter derivative can be obtained by dividing equations (3) and (6), and is constant during this maneuver. We can now numerically minimize the energy loss rate over the relevant set of turn parameters (V, γ), to obtain the optimal turn velocity V^*_{turn} and vertical flight path angle γ^*_{turn}. From these we can calculate the required controls and the optimal turning radius R^*.

4.4 Basic Maneuvers

We now connect the above-defined trajectory primitives to form the basic maneuvers, that will be used to connect consecutive states (or nodes) in our search graph.

Recall that a state is defined by $x = [X, Y, Z, V, \gamma, \xi]^T$. Suppose we are given two states, x_i and x_{i+1}, which we wish to connect by a basic maneuver. This will be done under the following terms:

1. A basic maneuver is composed of a sequence of the three trajectory primitives introduced above.

2. The spatial distance between neighboring states is sufficient to complete the maneuver. Allowing for sufficient distance between connected states is a key feature in reducing the complexity of the search graph.

3. The transitions between trajectory primitives are continuous in the state variables (X, Y, Z, ξ, V). However, we do allow instantaneous changes in the vertical flight-path angle γ. This assumption is also used in Rogers [14]. This variable can be modified relatively quickly by the pilot within its range, with negligible effect on the aircraft energy. Naturally, in the actual trajectory flown, this variable is smoothed by the pilot.

4. The altitude coordinate Z_{i+1} of the goal state will be ignored. Instead, we will match the (X, Y, ξ, V) goal coordinates, while maximizing the terminal altitude Z_{i+1}. Note that this is equivalent to minimizing the energy loss during the transition, as both velocities are specified. This choice follows the rationale discussed in Section 3.3.

5. Each trajectory primitive will use the optimal parameters that were selected for energy efficiency, as specified above for each primitive.

Given the last assumption, the geometry of the path in the horizontal plane is composed of turns at a given radius R^*, and straight line segments (the gliding and velocity adjustments primitives). We will essentially construct the required basic maneuver by choosing the *shortest path* in the vertical plane, using these path primitives. As is well known, the solution to this shortest-path problem (under constant velocity and altitude conditions) is obtained by the Dubins curve arc-line-arc construction [7]. Once the initial arc-line-arc curve for the horizonal path has been computed, it needs to be complemented and possibly modified to account for the velocity profile along the path. The straight line segments constitute optimal glide. The arcs consist of the turn primitive, with optimal turn speed. Evidently, some speed adjustment between these different velocities is needed (see Figure 2). For that purpose we will employ the Velocity-Adjustment Straight Flight Primitive. There are two kinds of velocity adjustments:

1. Internal velocity adjustments, from an arc to straight glide and vice versa. Here the required velocity adjustment maneuver are simply inserted at the initial and final parts of the straight-line segment of the path.

2. Boundary velocity adjustments: These are required to match the speed of the initial state V_i to the first turn or gliding primitive, and conversely to match the speed of the last turn or gliding primitive to that of the terminal state,

V_{i+1}. If such speed adjustments are required, the velocity adjustment primitives are first inserted, adding an initial and terminal straight-line segments to the horizontal path, and the optimal arc-line-arc Dubins curve is recalculated with the new start and end points induced by these segments.
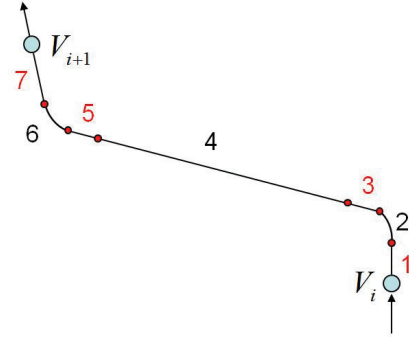

Figure 2: Top view of the primitive-connection scheme – transition $x_i \rightarrow x_{i+1}$.

The complete primitive-connection scheme is presented in Figure 2. The transition is divided into seven segments (at most):

1. Velocity adjustment $V_i \rightarrow V_{turn}^*$.

2. Turning at V_{turn}^*.

3. Velocity adjustment $V_{turn}^* \rightarrow V_{(L/D)max}$.

4. Gliding at $V_{(L/D)max}$ (using the $(L/D)_{max}$ ratio).

5. Velocity adjustment $V_{(L/D)max} \rightarrow V_{turn}^*$.

6. Turning at V_{turn}^*.

7. Velocity adjustment $V_{turn}^* \rightarrow V_{i+1}$.

Once fixed, the cost (i.e., altitude loss) of the complete maneuver may easily be calculated by adding up the costs of all segments.

Clearly, some of these seven segments may not be needed, depending on the initial and final states - e.g., if there is no need for turning, only segments 1,4 and 7 are required.

4.5 Wind Effects

Inclusion of the effect of wind in our dynamic model (equations 1)–(3) is standard. The magnitude and azimuth of the wind velocity w, assuming a steady wind front, may be estimated on board – e.g., using TAS and GPS data to obtain ground speed, or from available meteorological data. In the presence of wind, we retain the same trajectory primitives, developed above for the no-wind case. However, these trajectories are now computed relative to the wind motion, so that a drift of size $w\Delta t$ must be added to obtain the actual position, where Δt is the flight duration. The basic maneuver connection scheme is modified by an iterative algorithm, so that the destination point is reached despite this positional drift.

5 Flight Path Generation

Having defined the basic maneuvers, we may compose a flight path from the source to the destination point by concatenating a number of these maneuvers in appropriate sequence. It remains to find a sequence of such basic maneuvers which is energy-optimized, accordin3 - resorting to Optimal Graph Search.

5.1 State Discretization

Recall that the aircraft state vector consists of the six-tuple $(X, Y, Z, V, \gamma, \xi)$. Essentially, this space needs to be discretized, so as to form the nodes of the search graph. However, noting that the altitude Z is the quantity to be optimized, we will not include this coordinate in our discrete grid, but rather evaluate it as part of the search procedure. This indeed entails a reduction of one dimension in the formed grid, hence to a smaller search graph.

The five space variables (X, Y, V, γ, ξ) are sampled as follows, to form a discrete state grid:

• The horizontal plane variables (X, Y) are sampled uniformly within their range. We denote the corresponding sampling resolutions as Δ_X, Δ_Y, and reasonably set $\Delta_X = \Delta_Y$.
• The velocity V is sampled via squared-linear differences. That is, equal differences in the values of V^2. This allows linear changes in the energy between states.
• The vertical flight path angle γ is sampled uniformly.
• The horizontal flight path angle ξ obtains discrete values according to the structure of the spatial neighborhood group employed (as explained below).

The grid point corresponding to the initial state x_0 is denoted p_0, and the initial altitude by Z_0. Similarly, $\mathcal{P}_f = \{p_f\}$ denotes the set of grid points that correspond to the target set $\mathcal{X}_f$.

For each grid point $p_i = [X, Y, Z, V, \gamma, \xi]_i$ we define a neighborhood group $\mathfrak{N}(x_i)$. This group contains the grid points with horizontal coordinates that may be reachable from the original point p_i in a single transition. The neighborhood group is defined using a 2-ball in the horizontal plane:

$$\mathfrak{N}(p_i) = \{p_j \neq p_i : \sqrt{(X_j - X_i)^2 + (Y_j - Y_i)^2} \leq N\Delta_{XY}\}$$

This neighborhood group is referred as a *star formation* with radius N (Figure 3).

The horizontal angle ξ is assigned discrete values according to the neighborhood group. These values coincide with the angles between the original grid point and each of its neighbors. Using these convention (rather than, e.g., uniform sampling) allows to generate straight flight paths between consecutive grid points while keeping the horizontal flight path angle ξ fixed at one of its discrete values, thus barring the need for horizontal turns due to discretization mismatch.

Terrain avoidance is simply incorporated into the discrete model by specifying at each grid point x_i the minimal allowed flight altitude $(Z_i)_{\min}$. These values may be obtained from a Digital Terrain Model (DTM), with appr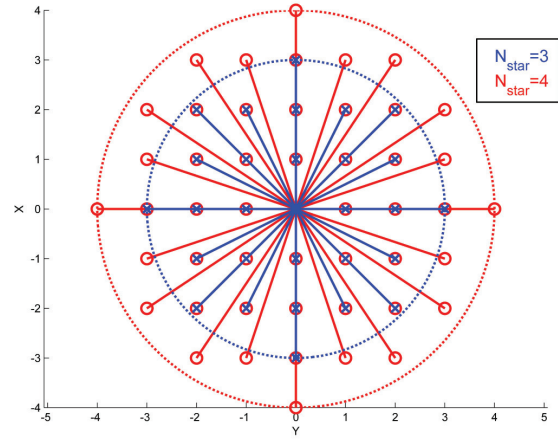opriate error margins added. Transitions to grid points will only be allowed if the achievable altitude exceeds the lower bound.

Figure 3: Star formation with radius $N = 3, 4$.

5.2 Grid Costs

Consider two neighboring grid points, p_i and $p_j \in \mathfrak{N}(p_i)$. Suppose p_i is tagged with an altitude coordinate Z_i (such tags are generated during the search process for each encountered grid point, starting with the origin whose altitude is given). We can then compute the basic maneuver from point p_i to p_j, and the associated altitude loss $-\Delta Z_{ij}$. We then obtain that p_j is reachable from (p_i, Z_i) at altitude $\hat{Z}_j = Z_i - \Delta Z_{ij}$.

The altitude loss $-\Delta Z_{ij}$ can be considered the cost of the edge connecting the tagged point (p_i, Z_i) and p_j. A few comments are in order regarding this cost function:

1. The basic maneuver that connects $(p_i, Z_)$ and p_j is energy-optimized, as described above, and generally depends on the altitude Z_i through the air density ρ. Therefore, the cost ΔZ_{ij} will generally depend on Z_i. If changes in air density can be ignored (e.g., when the overall change in altitude during the emergency landing maneuver is limited), then ΔZ_{ij} can be computed irrespectively of Z_i.

2. Recall that each grid point p_i includes a velocity coordinate V_i. Thus, maximizing the aircraft altitude Z_i at a give grid point is equivalent to maximizing its overall energy there. Our goal is to reach a target grid point $p_f \in \mathcal{P}_f$ with maximal altitude Z_f^*.

3. The maximum altitude Z_i^* at which a given grid point p_i can be reached can be considered its optimal *value*. Z_i^* is obtained as the initial altitude Z_0 minus the costs $-\Delta Z_{ij}$ along an optimal path on the grid.

5.3 Optimal Graph Search

Once the graph and cost structure have been defined, we focus on graph-search algorithms [16] with proven optimality.

The standard Dijkstra algorithm identifies sequentially the nodes that are closest to the origin, until the destination is reached. As the search propagates in forward direction, the cost computation between any pair of nodes p_i and p_j is

required only after first node has been assigned its optimal value, which is the maximal altitude Z_i^* to that node in our case. Hence, the corresponding cost $-\Delta Z_{ij}$ can be computed based on (p_i, Z_i^*) and p_j.

Dijkstra's algorithm admits well known polynomial bounds on the required computation time. However, some heuristics-driven variants can provide much faster performance in practice, provided that a good heuristics is available to guide the search. We employ here the standard A^* algorithm, which employs a heuristic function that provides an estimate for the disitance to the goal. To ensure convergence to the optimal solution, this function must not overestimate the distance to the goal; in that case it is called an admissible heuristic. We employ here the single-maneuver heuristic, that takes into account the aircraft initial and final orientation. It may be obtained by computing the cost associated with a single basic maneuver (as per section 4.4), that takes the aircraft form the current node to the goal node. This computation ignores any possible obstacles along the way.

5.4 Pre-compiled Dictionaries

An important means of accelerating the search procedure is by pre-computing the basic maneuvers between adjacent nodes, and storing the associated transition costs. This usefully transfers a major part of the computational burden to an off-line process. We observe that the transition cost from p_i to a neighboring node p_j does not depend on the absolute horizontal position (X_i, Y_i), but only on the difference $(\Delta X, \Delta Y)$ in that position. Therefore, the pre-compiled dictionary need be computed only for nodes p_i with vertical coordinates $X_i = Y_i = 0$. The vertical coordinate Z_i (which is not initially discretized) can be easily incorporated by computing the transition costs for several discrete altitude values, and possibly interpolating between them as needed during execution.

If wind effects are to be accounted for, such pre-compiled dictionaries need to be prepared for different wind conditions. Again, this computation may be carried out and stored for a discrete number of wind conditions, which are interpolated on-line according to the actual wind readings.

5.5 Flight-Path Smoothing

After an optimal path through the graph has been found, some post-processing may be applied to improve and smooth the obtained flight path. We can distinguish two categories of path smoothing. Path smoothing over the original grid can be readily carried out, for example, by systematically attempting to unify each two-segment path element into a single segment, using a single permissible basic maneuver. Once the resulting path segment is verified to be collision free, it can replace the original two-part segment. A second category of path smoothing is related to smoothing discontinuities due to transitions between different path segments. As already discussed, in most circumstances one can rely on the pilot to smooth out these discontinuities. In case we wish to present to the pilot a smoother version of the flight path, a number of path refinement algorithms may be applied for this purpose. An interesting

option is the use of standard missile guidance algorithms, such as Proportional Navigation, relative to a virtual target point (or "carrot") that is moved along the original path at a fixed distance before the aircraft. A recent application of this approach for aircraft navigation may be found in [10]. Specific details concerning the application of this approach to the present problem can be found in [2], but will not be pursued here.

6 Experimental Results

We present next some simulation experiments that illustrate the performance of the proposed algorithms. More complete details about the conditions of these experiments may be found in [2], along with some additional scenarios.

The aircraft model used in the simulation represents the Cessna 172 Skyhawk, the relevant parameters were taken from the POH (Performance Operational Manual) [5], as well as [15, 18]. The tested aircraft is assumed to have a mass of 2000 lbs. Configuration constraints include $-10° \le \gamma \le 30°$, $-60° \le \phi \le 60°$, $-44\,\text{knots} \le V \le 92\,\text{knots}$. For the discretization scheme we use $\Delta_X = \Delta_Y = 500m$, the range of V is divided to 10 discrete values, and a star formation with $N = 3$ yields 16 neighbor nodes in the horizontal plane. Accordingly, the heading angle ξ can assume 16 discrete values, with irregular spacing between 18 and 26 degrees (see Figure 3).

(a) Scenario 1: top view (b) Scenario 2 top view

(c) Scenario 1: isometric view (d) Scenario 2: isometric view

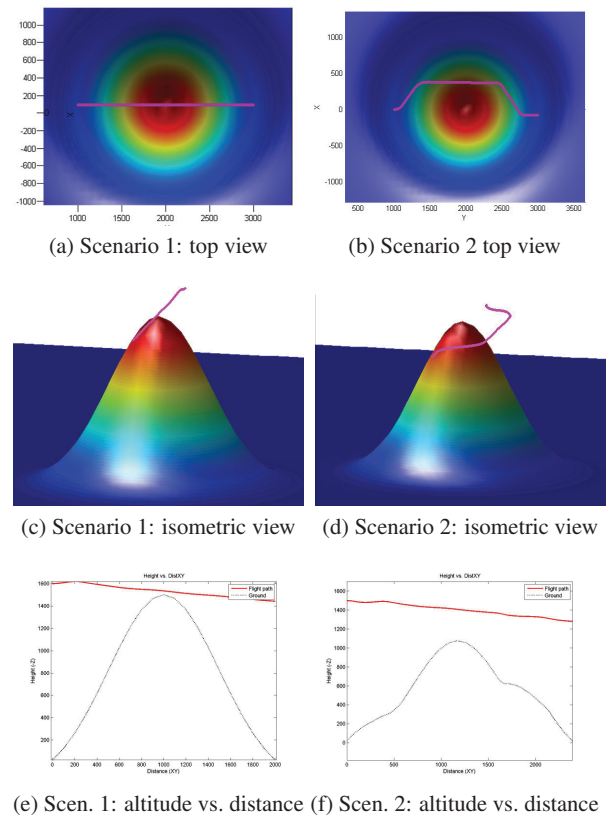(e) Scen. 1: altitude vs. distance (f) Scen. 2: altitude vs. distance

Figure 4: Computed trajectories for scenarios 1 and 2

Scenarios 1 and 2 address an identical terrain contour with a single obstacle, peaking at approximately 1,400 meters,

that stands between the initial and goal positions. These two scenarios differ only in the altitude of the initial points, which is 100 meter lower in the second case. Figure 4 presents the two obtained flight paths. In Scenario 1 there was enough energy for the aircraft to avoid the obstacle by flying above it. It required the aircraft to gain altitude at the initial phase to go over the peak, as may be seen in Figure 4(e). In Scenario 2, which starts at a lower point, there was not enough energy for such a maneuver, so the algorithm automatically chose a flight-path which flies around it. The run time for Scenario 1 was 0.2 seconds, and for Scenario 2 the run time was 1.3 seconds. Scenario 3 demonstrates a more elaborate planning task. The scenario consists of two obstacles, which the aircraft cannot fly over. As may be seen in Figure 5, the optimal path crosses in between the two obstacles.
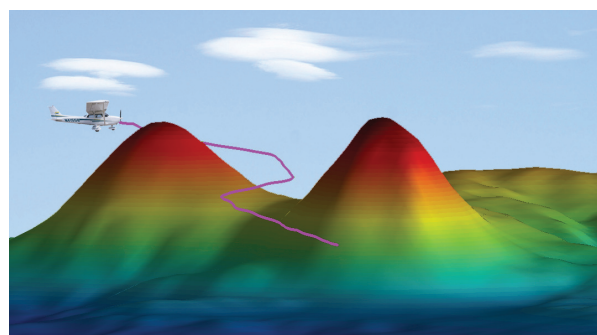
(a) Scenario 3 - isometric view
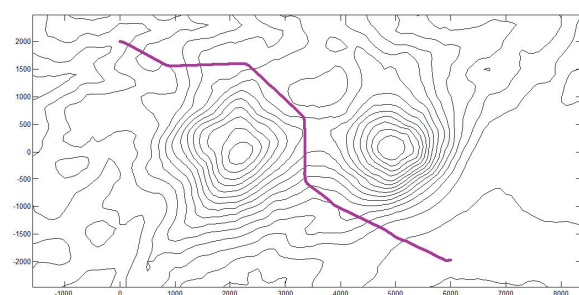
(b) Scenario 3 - top view
The height of the ground is presented by contour lines.

Figure 5: Scenario 3 - passing in-between obstacles

7   Conclusion

This paper proposes a novel method for fast on-board computation of energy-optimized flight-paths for engine-off emergencies. We have presented an efficient algorithm for generation of optimal flight paths. Our method is based on the notions of basic maneuvers, which are locally energy-optimized and extend over a range of several hundred meters or more. The algorithm relies on the construction of a search graph though discretization. It is therefore global in nature, and can readily accommodate obstacles of arbitrary shape and position.

Although the algorithm was developed for the generation of emergency flight-paths, the basic concept can be used to efficiently solve many other optimal planning and control problems. The complexity of such problems can be reduced dramatically, provided that meaningful motion-primitives can be derived.

REFERENCES

[1] Z. Adin and E. Kreindler. Control of free-flight aeromodels for maximal flight duration. Master's thesis, Technion - Israel Institudte of Technology, 1992.

[2] A. Adler. Real-time computation of flight paths for emergency landing. Master's thesis, Technion – Israel Institute of Technology, August 2009.

[3] J. Ben-Asher. *Optimal Control Theory with Aerospace Applications*. AIAA, Reston, VA, 2010.

[4] A. E. Bryson. *Dynamic Optimization*. Addison Wesley, Menlo Park, CA, 1999.

[5] The Cessna Aircraft Company. *Cessna 172N Performance Operational Manual*, 1959.

[6] M. Dekel. Pseudo-spectral-method based optimal glide in the event of engine cut-off. Master's thesis, Technion—Israel Institute of Technology, 2011.

[7] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal position. *American Journal of Mathematics*, 79:497–516, 1957.

[8] H. J. Kushner and P. G. Dupuis. *Numerical Methods for Stochastic Control Problems in Continuous Time*. Springer-Verlag, second edition, 2000.

[9] P. Lu, D. B. Doman, and J. D. Schierman. Adaptive terminal guidance for hypervelocity impact in specified direction. *Journal of Guidance, Control, And Dynamics*, 29(2):269–278, March-April 2006.

[10] E. Medagoda and P. Gibbens. Synthetic-waypoint guidance algorithm for following a desired flight trajectory. *Journal of Guidance, Control and Dynamics*, 33(2):601–606, March 2010.

[11] National Transportation Safety Board. Aviation Statistics Reports, Table 10: Accidents, Fatalities and Rates, 1991-2010, U.S. General Aviation. Available at: www.ntsb.gov/data/aviation_stats.html.

[12] B. J. Rademacher, P. Lu, A. L. Strahan, and C. J. Cerimele. In-flight trajectory planning and guidance for autonomous parafoils. *Journal Of Guidance, Control, and Dynamocs*, 32(6):1697–1712, November-December 2009.

[13] E. Rippel, A. Bar-Gill, and N. Shimkin. Fast graph-search algorithms for general-aviation flight trajectory generation. *Journal of Guidance, Control, and Dynamics*, 28(4):801–811, October 2005.

[14] D. F. Rogers. The possible 'impossible' turn. *AIAA Journal of Aircraft*, 32(2):392–397, March-April 1995.

[15] J. Roskam. *Airplane Flight Dynamics and Automatic Flight Controls, Part I*. DAR corporation, Lawrence, KS, 1995.

[16] S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education, 2003.

[17] I. Shapira and J. Z. Ben-Asher. Singular pertubation analysis of optimal glide. *Journal of Guidance, Control, and Dynamics*, 27(5):915–918, September-October 2004.

[18] K. R. Sivier. Cessna 172 data for the frasca r-see aircraft flight simulator. Technical report, University of Illinois, 1999.

[19] E. Yong, G. Tang, and L. Chen. Three-dimensional optimal trajectory for global range of cav. *1st International Symposium on Systems and Control in Aerospace and Astronautics*, 2006.