



An Off-Runway Emergency Landing Aid for a Small Aircraft Experiencing Loss of Thrust

Pedro F. A. Di Donato*

National Civil Aviation Agency - ANAC, São José dos Campos, SP, 12242, Brazil

University of Michigan, Ann Arbor, MI, 48109, USA

Ella M. Atkins †

University of Michigan, Ann Arbor, MI, 48109, USA

Loss of thrust is an ongoing concern for single engine general aviation operations. If no runway is reachable, the pilot must rapidly select an off-field landing site. Future avionics upgrades could assist in this task. This paper proposes the use of a publicly available road database to assist a pilot with off-field landing site selection. Roads are divided into segments that exclude curves and obstacles such as overpasses. A safety-driven utility function is proposed to sort identified landing sites. Paths from the current position to reachable emergency landing sites are calculated as sequences of arcs and straight segments based on the gliding performance of the airplane. This method can also be applied to Unmanned Aircraft Systems or used in conjunction with a vision system.

Nomenclature

(x, y, z)	Aircraft position; Earth-fixed frame
γ	Flight path angle
$\angle B1A1B2$	Angle formed by edges $\overline{B1A1}$ and $\overline{A1B2}$ measured from the first to the second counterclockwise
ϕ	Bank angle
ψ_{Aij}	Heading of edge $AiAj$
ρ	Air density
Ai	Node i from segment A
C_D	Drag coefficient
C_i	Parameter weight for runway utility function
C_L	Lift coefficient
C_{D0}	Parasite drag coefficient
D	Drag force
d	Distance to footprint boundary
d_{Aij}	Length of edge $AiAj$
g	Acceleration due to gravity
k	Induced drag coefficient
L	Lift force
m	Mass
p_i	Parameter of utility function
q_f	Facility availability measure
q_i	Instrument approach quality
q_s	Surface quality
R	Turn radius in the horizontal plane

*Regulation Specialist (on leave), PhD Student, Aerospace Engineering Department, Member.

†Associate Professor, Aerospace Engineering Department, Associate Fellow.

r_l	Landing site length
r_w	Landing site width
U_{hwy}	Highway landing site utility function
U_{rwy}	Runway utility function
V	True airspeed
v_t	Road speed limit
w_c	Crosswind velocity
w_h	Headwind velocity
W_i	Parameter weights for road landing site utility function
<i>Subscripts</i>	
i	Landing site index
max	Maximum over the set of reachable landing sites

I. Introduction

On January 4th, 2014 a Piper PA-28-180 airplane lost thrust and performed an emergency landing on the Major Deegan Expressway in the Bronx borough of New York City after the pilot reported an engine malfunction.¹ Despite substantial damage to the aircraft, the pilot and two passengers were not injured.

Powerplant malfunctions such as this were responsible for 74 accidents involving non-commercial fixed wing aircraft in 2010.² This represents 43% of all accidents due to mechanical or maintenance factors, and 6% of all accidents of this aviation class in that year. For single-engine airplanes, an engine failure means that the pilot has a hard constraint on remaining range and time aloft. For general aviation or unmanned aircraft flying at low altitude, these limitations often preclude the possibility of landing on an airport runway, and pilots are faced with the difficult task of finding and actually landing at an off-airport site.

Roads are often selected by pilots in emergency landing situations. For example, in April 2013 a Cessna 172 pilot decided to land on a street in Henderson, NV after realizing that it would not be possible to return to the runway from which he departed.³ In May 2013, after a powerplant malfunction during cruise flight a Cessna 182 pilot chose Highway 50 near Delta, CO for landing.⁴

The 172 pilot said that he tried to turn back to the runway but was not sure if sufficient altitude was available. The 182 pilot reported that he used available maps and avionics system to look for options. He chose one airport to divert based on the runway length and heading change needed to align at the final. Deteriorating engine performance forced him to choose the highway rather than “fields and dirty roads”, “rocky terrain”, “ravines” or a river. Current avionics systems provide a list of nearest airports and a map of the region but leave further calculations to the pilot. This paper builds on previous work in emergency landing planning^{5,6} to offer pilots and/or automation off-runway landing options with focus on the loss-of-thrust emergency case.

Emergency landing planning tools have been proposed previously. In 1995 Rogers proposed an optimum maneuver, i.e. bank angle and velocity, to return to a runway after an engine failure just after takeoff.⁷ He also calculated the required runway length and the impact of different parameters in its value. His study defines the maneuver, but the departure runway is assumed the landing site and no specific automation aid for the pilot is proposed.

An emergency autoland system for general aviation was proposed by Siegel in 2011.⁸ Its design includes automatic landing site selection, guidance to this site and guidance along the final approach. Her landing site selection uses four criteria levels which go from runways with more than 5000ft to any unpopulated area. The levels are used according to the status of the aircraft engine and landing site availability inside the aircraft reachable footprint. The emergency landing sites, however, are pre-loaded into the FMS. In case none are found inside the calculated range, the author proposes the use of landing charts and color detection algorithms to find suitable unpopulated areas without consideration of other databases to assist.

The second author has been working with an emergency flight planning aid since 2001.⁹ An architecture for landing site selection and flight planning for loss of thrust is presented in Ref. 5. Landing site search provides a priority queue of runways that are feasible for the flight planner. This queue is ordered by a cost function that takes into account not only the distance to the runway, but also parameters such as runway length and width, wind components and facilities available at the airport. An application of this method to US Airways flight 1549 that experienced loss-of-thrust due to bird strike is presented in Ref. 10.

Frantis¹¹ proposed an emergency and precautionary landing assistant for small airplanes with engine failure based on stored terrain data. His assistant provides information including best glide speed, time to landing and power line data. Additionally, it computes and identifies emergency landing fields, areas where there are no buildings, power lines and trees. A terrain database can provide the required data for this task, although the article does not discuss details of how this is implemented. The author did test his system with pilots and their feedback is documented. Information regarding time to land was considered easier to interpret than range by pilots. Additionally, pilots reported that the emergency field locator would be more useful only in darkness or low visibility conditions due to the lack of time to carefully compare the images of the synthetic vision system with the field they observe through the window. Power line information was considered valuable.

The identification of landing sites based on information acquired by cameras was proposed by Sehn et al. in 2013.¹² The authors present a survey of similar work done with the same objective and compare their results with previous studies. The authors discuss that ideally when an emergency occurs, the airplane should be guided to an approximately smooth area and then the vision system would perform the final assessment to lead the aircraft to a safe landing site. They propose that this initial approximate area guidance decision should be made based on an elevation database.

We present a method to identify possible emergency landing sites from publicly available databases. The search starts with conventional runways. If no feasible solution is found, or the risk is considered too high, the system searches for off-runway landing sites. The first option is major roads. As a last resource, other suitable areas would be identified based on lack of obstructions and low population density. Identification and sorting of preferred emergency landing sites are performed by extending the landing site search proposed in Ref. 5 to consider both runway and off-runway landing sites.

As discussed in Ref. 12 this capability would complement a landing site selection protocol based on cameras, relying on cameras only for final identification of transient obstacles and local area detail not modeled in a database. Instead of using only elevation databases, considering that roads are the usual choice for pilots in the studied scenario, we propose to search for road landing sites using road databases. If this search proves unsuccessful, we would use terrain elevation and population databases. These databases are significantly smaller, so tractable for use in real-time search, than is a detailed terrain database such as that used by Frantis.¹¹

This same idea of using of public road databases to supplement information given by visual sensors is already being explored for ground vehicles. Ref. 13 proposed the use of the same database used in this study combined with visual odometry to provide estimates for vehicle's position. This paper documents the first application of this database to a flight planning application.

Below a general overview of an emergency flight planner is overviewed in Section II. Section III presents the databases used for airport runways and roads including the proposed preprocessing method. Reachable landing site identification and prioritization, as well as the airplane model and assumptions used for footprint and path calculation are the focus of Section IV. Results are presented in Section V. Finally conclusions and future work are presented in Section VI.

II. Emergency Landing Planner - Overview

An emergency flight planner has been under development by the second author since 2001.^{5,9,10} A proposed upgrade to the approach presented in Ref. 10 is shown in Figure 1. Figure 1 presents improvements in several areas such as:

- After the trajectory to a runway is planned, risk is evaluated based on the terrain and population of the overflowed area. This improvement is out of the scope of this paper but will be critical information to incorporate in future work;
- If no feasible landing runways are found, or if risk for all runways is high, the system identifies off-runway landing sites. Roads are utilized first for reasons summarized above;
- We assume the aircraft is equipped with a data link (e.g. cell phone or tablet) such that traffic data is accessible. If traffic density is sufficiently low a trajectory to the most suitable road is found that aligns aircraft heading with the road to the extent possible.

- If no suitable road landing site is found, or if the landing risk is too high, the program looks for other possible landing sites using terrain elevation and population database information.

The focus of this paper is to develop the second capability listed above using databases that are public and currently available online.

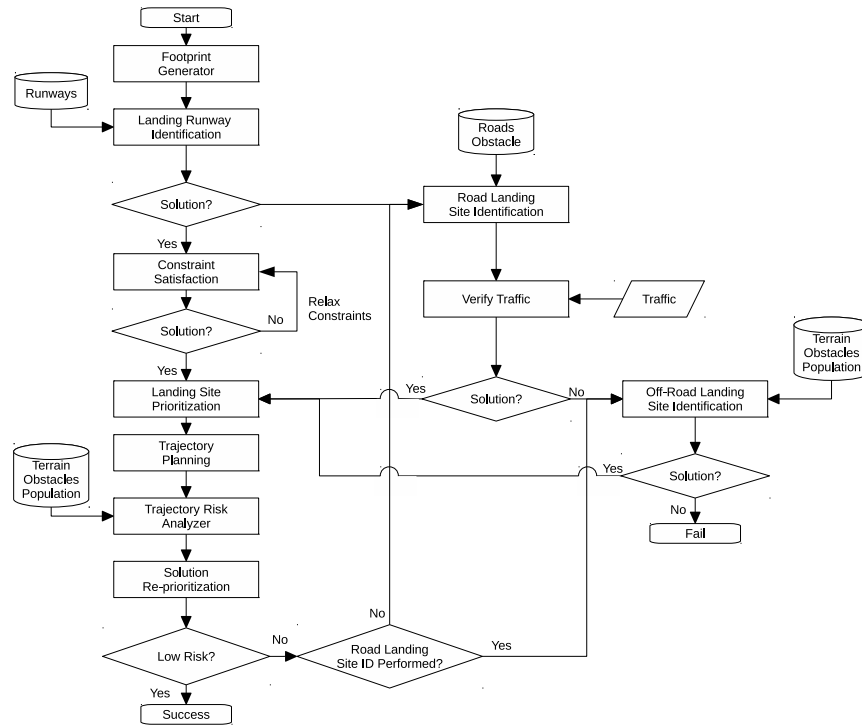


Figure 1: Flowchart of the proposed landing site identification procedure

III. Databases Description and Preprocessing

III.A. Airports

The Federal Aviation Administration makes available in its website databases of airport facilities and runways as xls files (http://www.faa.gov/airports/airport_safety/airportdata_5010/). The landing site search algorithm as presented in Ref. 5 ranks the runways based on both runway characteristics and airport facilities such as maintenance availability. Therefore, it is necessary to preprocess the data in order to link the two datasets.

This preprocessing requires extraction of the relevant information from both databases, linking both of them, i.e., runways to the corresponding airport, and exporting the result to a file that will be used by the real-time emergency landing planner. Note that each runway is transformed into two corresponding to both landing directions. Moreover, the database also includes helipads that were not considered in this work given our focus on fixed-wing aircraft.

III.B. Roads

In this work we used road data that is available under the Open Database License from OpenStreetMap[®].¹⁴ This collaborative database include not only roads, but also other information such as land use, rail and power lines. Its level of detail is dependent on the region, but studies demonstrate that the data can be considered reliable.¹⁵ Moreover, with regard to U.S roads, the Topologically Integrated Geographic Encoding and Referencing System (TIGER) data from 2005, produced by the US Census Bureau, was incorporated to the database in 2007.

Once the data from a particular region of interest is downloaded, the first preprocessing step is to filter the data for major roads information. The database contains tags to allow this filtering; only “motorway” and “trunks” road types were considered. Those correspond with the two highest levels of a seven level roads classification structure. Additionally, any road segment that is a tunnel is excluded. Output from OpenStreetMap is a list of viable motorway and trunk nodes described by GPS positions, and “ways” which are collections of nodes that forms roads. Ways are tagged with information. Speed limit and number of lanes were used to build the safety-based utility function to rank the landing sites as will be discussed in the next section. The number of lanes is always presented, while the maximum speed was conservatively estimated as 30mph for the segments without such information.

Each road can be divided in several segments (“ways”), therefore the next step is to connect different road segments. This is needed to identify feasible landing strips that extend along more than one segment. Figure 2 illustrates this process. Two road segments A and B with three nodes each are presented in the database. Since their extrema are the same they are potential candidates to be joined.

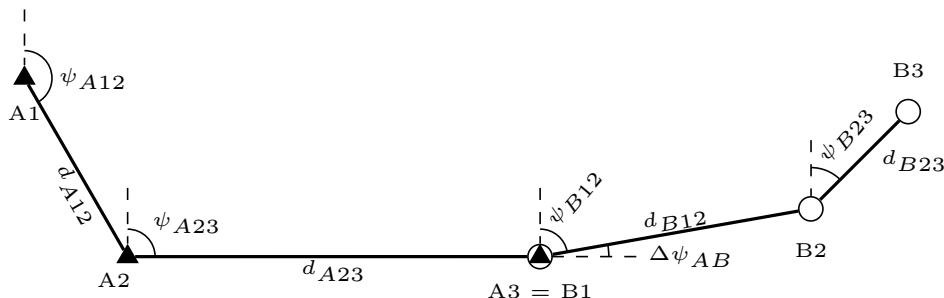


Figure 2: Road segment linkage and straight line identification

Processed data describes major roads with one or two independent segments, one for each traffic direction. Thus, the last point of one segment is only linked to a start point of another. Moreover, since nearly straight road segments are required, connections are only made if the heading change across the connection, $\Delta\psi_{AB}$ as illustrated in Figure 2, is smaller than a threshold of 10 degrees for this study. For any pair of nodes, such as $A1$ and $A2$, relative heading ψ_{A12} and distance between them d_{12} is calculated using Vincenty’s method¹⁶ applied to World Geodetic Ellipsoid WGS84. A final step in connection of roads is making sure the “tags” associated with each merged segment are not lost. This was done by transferring information from the whole segment to each associated edge.

Once the roads are connected, the next step is the proper identification of emergency landing sites. Road-based landing sites are defined by straight paths without known bridges crossing over them. Straight segments were mapped connecting nodes along each road and calculating the mean heading of each collection of edges. Using Figure 2 as example, the mean heading from the nodes $A1$ to $B2$ (supposing the two ways were connected previously) is given by:

$$\bar{\psi}_{A1-B2} = \frac{\psi_{A12}d_{A12} + \psi_{A13}d_{A13} + \psi_{B12}d_{B12}}{d_{A12} + d_{A13} + d_{B12}} \quad (1)$$

Edges are added to the group until a new edge’s heading differs from the collective average heading by a threshold set to 3 degrees in this study. This method allows large radius of curvature roads to also be considered as possible landing sites. If the total distance of the group of edges satisfies a minimum distance criteria ($0.5NM \approx 3040ft$ in this case) they are marked as a landing site candidate. Those searches are performed starting in each node of each road. If landing site candidates with overlap are identified, the one with smaller length is discarded.

The second step of preprocessing is to eliminate candidate landing strips that have overpass bridges. OpenStreetMap tags road bridges and other obstructions such as rail bridges. All are marked with the “bridge” tag and can be filtered from the database using the same tool used to select major roads. The filtered result is again a collection of nodes and ways (sets of edges).

Due to its broad application, the number of nodes and ways marked with a bridge tag can be substantial. A method for rapidly discarding ways that do not cross a landing site candidate is valuable even if this method would not eliminate all of them. This first layer of elimination is performed by simply checking

latitude and longitude maxima and minima of each segment of bridge way and comparing with values for the considered segment of the landing site candidate. Two segments cannot cross if one's minimum (maximum) latitude is greater (smaller) than the maximum (minimum) latitude of the other; the same applies to the longitude.

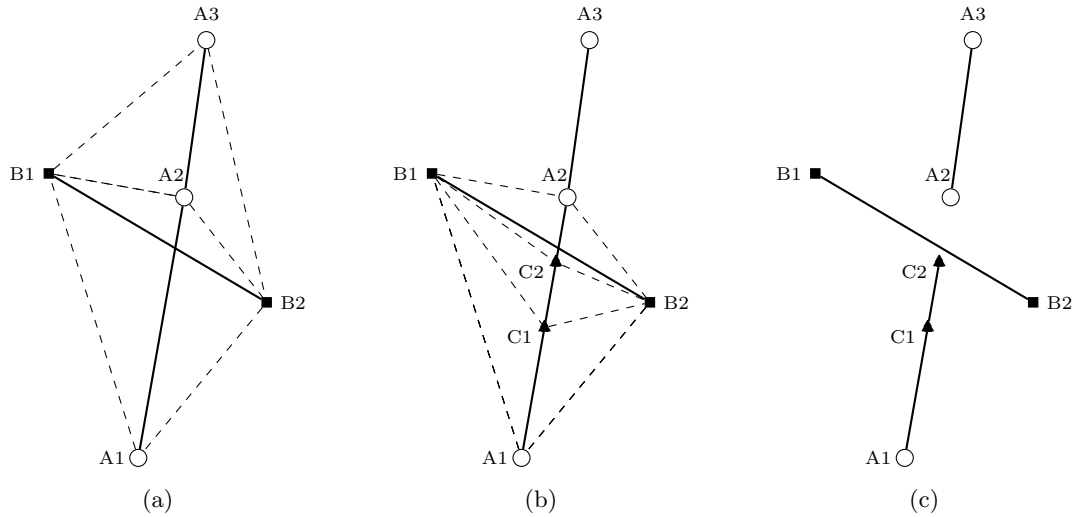


Figure 3: Bridge crossing geometries. (a) Crossing detection, (b) Crossing point identification, (c) Landing site division

Figure 3a illustrates how intersections between landing strip candidates and bridges are detected if they are not discarded in the first layer. Let A_1 , A_2 and A_3 represent road nodes previously marked as candidates and let B_1 and B_2 represent nodes marked with the bridge tag. For each two nodes A_i and A_j the angles of the quadrangle formed by vertices A_i , B_1 , A_j and B_2 are calculated from the heading of the segments linking each node pair of the set. Each node triple such as (B_1, A_1, B_2) define two angles $\angle B_1A_1B_2$ and $\angle B_2A_1B_1$. In this paper we use the definition that $\angle B_1A_1B_2$ is the angle formed by edges $\overline{B_1A_1}$ and $\overline{A_1B_2}$ measured from the first to the second counterclockwise. Thus, $\angle B_2A_1B_1$ is its complementary angle. A crossing would be characterized by the quadrangle $A_iB_1A_jB_2$ having all internal angles smaller than 180° .

Therefore, for each triple of nodes that form such a quadrangle $(A_iB_kA_j)$ or (B_k, A_i, B_l) , the algorithm finds the smaller angle between $\angle A_iB_kA_j$ and its complementary angle. The sum of those angles is equal to 360° when a crossing is present. Otherwise the sum will be smaller than 360° because it does not represent the sum of the internal angles of a quadrangle. In the case of Figure 3, the following angles would be computed for each of the quadrangles:

$$\begin{aligned} & \text{Quadrangle } A_1B_1A_2B_2 \\ & \angle A_1B_1A_2 + \angle B_1A_2B_2 + \angle A_2B_2A_1 + \angle B_2A_1B_1 = (\angle B_2B_1A_2 + \angle B_1A_2B_2 + \angle A_2B_2B_1) + \\ & \quad (\angle B_2A_1B_1 + \angle A_1B_1B_2 + \angle B_1B_2A_1) \\ & \quad = 180^\circ + 180^\circ = 360^\circ \end{aligned} \quad (2)$$

$$\begin{aligned} & \text{Quadrangle } A_2B_1A_3B_2 \\ & \angle A_2B_1A_3 + \angle B_1A_3B_2 + \angle A_3B_2A_2 + \angle B_1A_2B_2 = 2\angle B_1A_2B_2 \\ & \quad = 360^\circ - 2(\angle B_2B_1A_2 + \angle A_2B_2B_1) < 360^\circ \end{aligned} \quad (3)$$

The crossing test will indicate a crossing for the first set of nodes and not for the second. Note that this test assumes planar geometry, as distances considered for aircraft landing are sufficiently small to support this assumption.

The final step is splitting the landing segment candidate where a crossing was identified. The intersection point is obtained based on the bisection method for root finding. Starting with the landing edge where the crossing was identified, a new midpoint node is created. For example, C_1 in Figure 3b is created and the above algorithm identifies if the crossing is on edge $\overline{A_1C_1}$ or $\overline{C_1A_2}$ and the other is discarded. In this case,

the new edge would be $\overline{C1A2}$. The procedure is repeated; a new node $C2$ is generated in the midpoint and the position of the crossing is identified, until the edge becomes smaller than a selected threshold. Finally, the candidate landing strip is split in two parts, in our example, from $A1$ to $C2$ and from $A2$ to $A3$ for which no crossing is presented as in Figure 3c. Both final candidates are tested to check if they satisfy the criteria of minimal length and are discarded otherwise. Finally, after all bridge segments are processed, a list of the emergency landing road segments is produced and stored in a text file similar in format to that used for airport runways.

Note that the database used to create the road landing sites does not contain any information regarding elevation. In this paper, the Open Elevation database from MapQuest was used to provide this data. Using its web services,¹⁷ the elevation of each segment endpoint can be accessed. The elevation of the threshold of each identified landing site was obtained and included in the text file with the other information obtained from the OpenStreetMap database.

IV. On-Board Processing and Utility-Based Prioritization

Both preprocessed databases of runway and road landing sites can be pre-loaded into the airplane. After a failure, however, reachable landing sites must be identified and sorted based on the location where the failure occurs to assist the pilot in choosing the best alternative. The sort criteria is based not only in the landing site characteristics already presented in the database and discussed in the preprocessing phase, but also in parameters calculated on-board.

IV.A. Airplane Model and Footprint Generation

To initially filter landing sites to extract the reachable subset given aircraft altitude and glide performance, a footprint is calculated. The method is based on Ref 5 using point-mass model equations of the airplane performance and considering only trim states.¹⁸

$$-D - mg \sin \gamma = 0 \quad (4)$$

$$L \cos \phi - mg \cos \gamma = 0 \quad (5)$$

$$L \sin \phi - \left(\frac{mV^2 \cos^2 \gamma}{R} \right) = 0 \quad (6)$$

where the aerodynamic forces are computed using a quadratic drag polar, given by:

$$D = \frac{1}{2} \rho(z) S C_D V^2 \quad L = \frac{1}{2} \rho(z) S C_L V^2 \quad C_D = C_{D0} + k (C_L)^2 \quad (7)$$

Above, aircraft parameters are specified as: V true airspeed, R turn radius in the horizontal plane, m weight, S wing reference area, C_{D0} zero lift coefficient and k induced lift coefficient. Gravitational acceleration is considered constant $g = 9.80665 m/s^2$. Air density ρ is a function of altitude and is modeled using US Standard Atmosphere 1976.¹⁹

The aircraft position is then calculated by:

$$\dot{x} = V \cos \psi \cos \gamma \quad \dot{y} = V \sin \psi \cos \gamma \quad \dot{z} = V \sin \gamma \quad (8)$$

where ψ is the aircraft heading subject to the circular motion constraint:¹⁸

$$\dot{\psi} = \frac{V \cos \gamma}{R} \quad (9)$$

The footprint is then calculated assuming a turn with a specified change in heading, followed by a straight path segment with best glide flight path angle up to zero altitude. Small improvements from Ref. 5 considered in this paper are: the turn is assumed to be done with minimum flight path angle instead of best glide which we show in complementary work²⁰ to be the best choice for appreciable turns. The footprint is estimated to be a circle and the best fit obtained via the Modified Least Squares method from Ref. 21. The landing sites are filtered to consider only those contained inside the footprint circle.

IV.B. Safety-based utility function

Ref 5 proposed the use of the following safety-based utility function U to prioritize reachable landing runways:

$$U_{rwy} = \sum_i C_i p_i = C_1 \frac{r_l}{r_{l,max}} + C_2 \frac{r_w}{r_{w,max}} + C_3 q_I + C_4 \frac{d}{d_{max}} + C_5 \frac{w_h}{w_{h,max}} + C_6 \frac{w_{c,max} - w_c}{w_{c,max} - w_{c,min}} + C_7 q_s + C_8 q_f \quad (10)$$

The utility is a weighted sum of the following parameters: runway length r_l , runway width r_w , instrument approach quality q_I , distance to footprint boundary d , headwind velocity w_h , crosswind velocity w_c , surface quality q_s and facility availability measurement q_f .

Following the same approach, for landing on roads, we propose the following safety-based utility function:

$$U_{hwy} = \sum_i W_i p_i = W_1 \frac{r_l}{r_{l,max}} + W_2 \frac{r_w}{r_{w,max}} + W_3 \frac{v_{t_i}}{v_{t_{max}}} + W_4 \frac{d}{d_{max}} + W_5 \frac{w_h}{w_{h,max}} + W_6 \frac{w_{c,max} - w_c}{w_{c,max} - w_{c,min}} \quad (11)$$

Differences between (10) and (11) are:

- Exclusion of the parameters associated with instrument approach, facility availability and surface quality as these are not available or not applicable for roads;
- The landing site width is calculated based on the number of lanes of the highway landing strip;
- Inclusion of a parameter, weighted by W_3 , associated with the speed limit. Currently, we assume that the higher the road speed limit, the closer it is from a small manned airplane landing speed which facilitates safe landing.

V. Results and Discussion

Inspired by the major Deegan Expressway incident in early 2014,¹ this paper presents a similar case study, even though specific data of the accident is not yet available. The aircraft model used is of a Cessna 172. The initial position is assumed N 40.865°, W 73.88°. The altitude and true heading are chosen to be 1500ft and 360° (North). For simplicity, no wind was considered in this case study.

V.A. Road Database Preprocessing

Figure 4a presents the area for which information from OpenStreetMap was obtained for this example. Figure 4b presents the processed data of the major highways without tunnel segments. An overlap of Figure 4b with identified landing site candidates is presented in Figure 5. For better visualization, the candidate landing sites are marked with a thicker line. Figure 6 presents a detailed view of one of the candidates marked by a dashed rectangle in Figure 5. This corresponds to a segment of US-87, Major Deegan Expressway, close to the Northern boundary of Manhattan. This enlarged figure presents the nodes of the database represented by the circle symbols, while the two lines correspond to the two travel directions of the road.

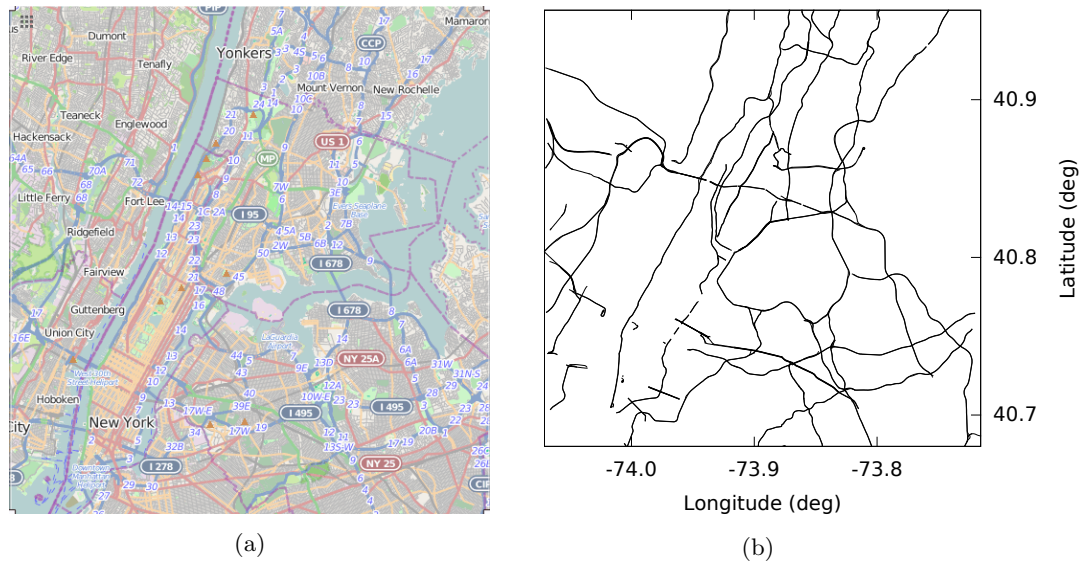


Figure 4: Road database preprocessing. (a) Original Map from OpenStreetMap, (b) Processed major road network

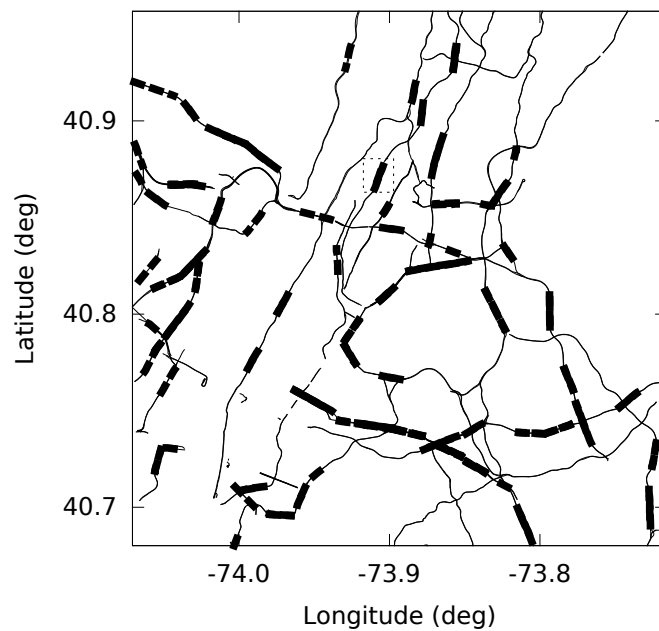


Figure 5: Landing strip identified over the road network

In Figure 6, each straight path represented by a wider line and filled circle nodes was selected as a candidate landing site by the algorithm. The lack of connection on the upper side of the figure corresponds to a tunnel that is filtered in the early stages of preprocessing. Moreover, the individual heading change of each road direction can result in different landing site identifications as this case illustrates with the North direction site showing a considerably longer distance than the South direction counterpart.

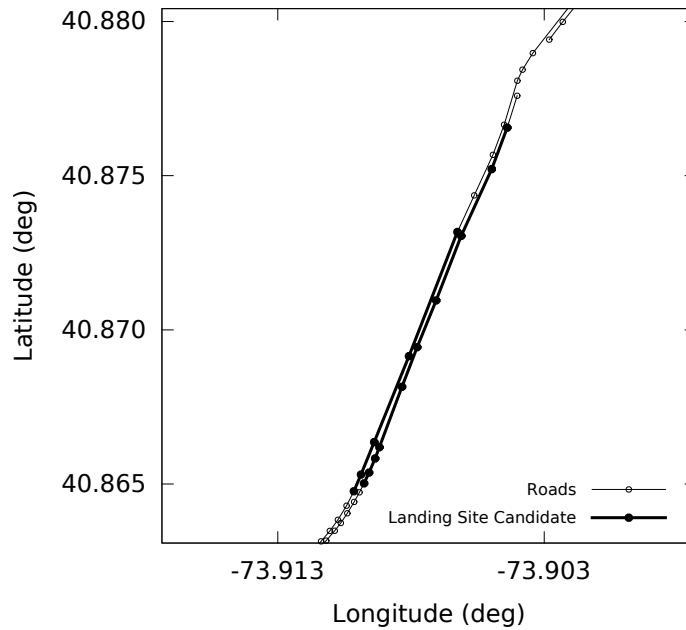


Figure 6: Landing strip identification - detailed view

The elimination of segments due to bridges is illustrated in Figure 7 where the same area of Figure 6 is presented with an overlap of bridge nodes and ways of the database represented by squares. The long line of the figure corresponds to an elevated railway, and the several parallel lines at one site correspond to the Broadway bridge. This illustrates the detailed coverage of the bridge database. This step of the preprocessing is considerably demanding for the entire road database. Figure 7 presents the same area after the elimination of conflicts with bridges. The landing site relative to the northbound lanes is correctly shortened to avoid bridges. On the other hand, since southbound lanes had no conflict with any of the bridges, they are still maintained.

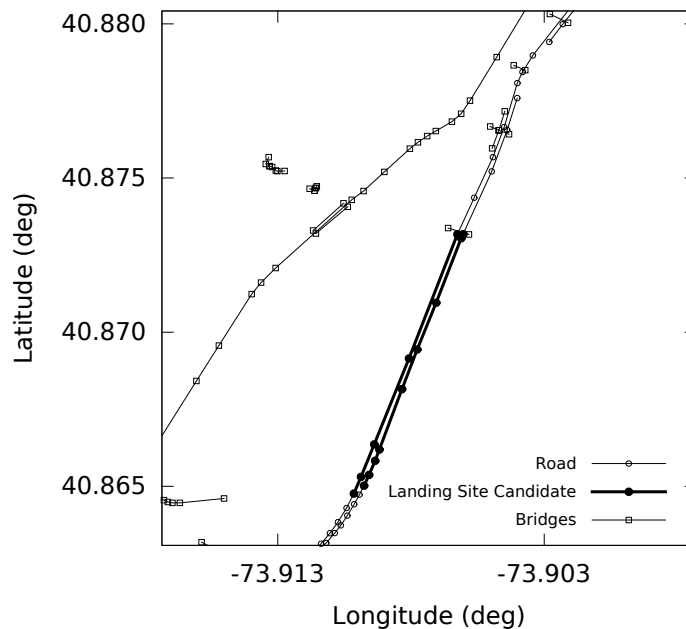


Figure 7: Bridge conflict identification and solution.

The present example area corresponds to 977Mb of data in OpenStreetMaps, while the bridges and major

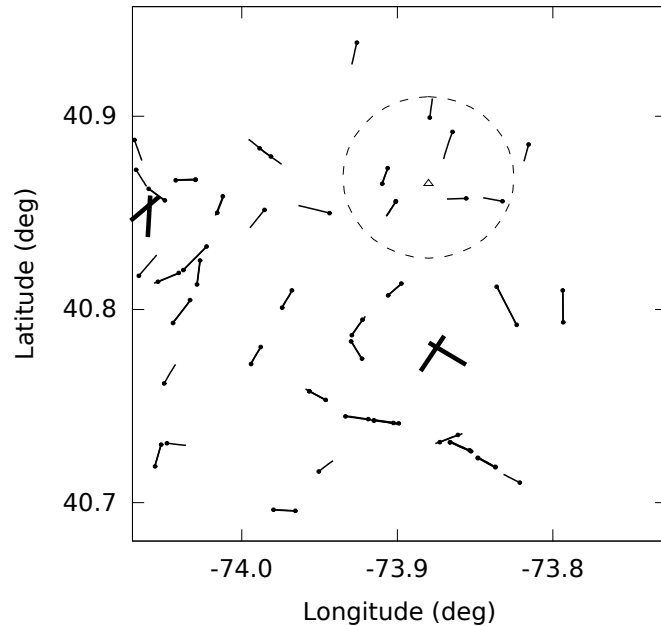


Figure 8: Footprint for the airplane at the triangle mark at an altitude of 1500ft

roads (motorway and trunk tags) alone represented approximately 4Mb each (8Mb for both). Those numbers justify preprocessing to provide aircraft with possible landing site segments.

V.B. Footprint and Road Landing Sites Priorization

Figure 8 presents the calculated footprint as a dashed line circle for the New York loss-of-thrust case study. In this case, the closest airports are La Guardia and Teterboro, both with runways marked with thick lines in the figure. It is clear that all of those runways are outside of the footprint boundary. The emergency flight planner, therefore, would advance to search for road landing sites. Note that the footprint is not centered on the initial position, which is explained by the fact that its derivation takes into account the turn needed to change aircraft heading.

All road landing site candidates derived from the preprocessed database are presented in Figure 8. Circles mark the identified thresholds according to the traffic direction of each lane. The reachable landing sites are presented in Table 1. Two sites correspond to one road with two same-direction parallel lanes explaining why only 7 thresholds of the 8 listed in Table 1 can be observed in Figure 8. Those landing sites were sorted using equal weights in cost function: $W_1 = W_2 = W_3 = W_4 = 0.25$ and $W_5 = W_6 = 0$ since wind was not considered.

Table 1: Reachable road landing sites sorted according to the utility function

Number	Latitude (deg)	Longitude (deg)	Altitude (ft)	True Heading (deg)	Length (ft)	Footprint Boundary Distance (NM)	Utility
1	40.8919	-73.8646	75	197	5160	0.93	0.82
2	40.8732	-73.9063	20	199	3246	1.27	0.80
3	40.8650	-73.9098	26	019	3146	1.13	0.77
4	40.8992	-73.8792	200	007	3488	0.65	0.71
5	40.8575	-73.8558	112	268	3266	1.22	0.70
6	40.8561	-73.9010	161	211	3096	1.30	0.70
7	40.8557	-73.9010	144	211	3059	1.28	0.69
8	40.8560	-73.8325	23	281	3343	0.22	0.51

V.C. Path Calculation

Once the reachable landing sites are sorted based as shown in Table 1, a Dubins-based flight planner based on the one described on Ref 5 connects the initial aircraft position with each identified threshold displaced by 0.2 NM if feasible. In the example presented here paths to landing sites 2, 4, 6 and 7 can be calculated using this simple flight planner. Landing sites 3 and 5 require an intermediate turn to allow the airplane to descend as needed. Required turns make it impossible to reach sites 1 and 8.

Figure 9 presents the calculated no-thrust landing paths. The initial position of the airplane is made to be equal to the upper tip of the triangle to make the initial turn visible. According to our prioritization presented in Table 1 the chosen path would be to landing site 2, but all possible paths could be presented to the pilot for an ultimate decision.

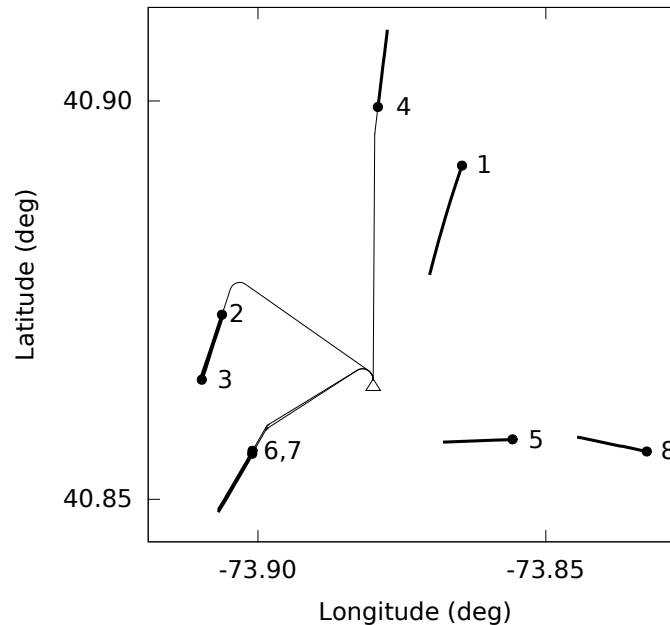


Figure 9: Trajectories calculated to road landing sites. Number correspond to entries in Table 1.

VI. Conclusions and Future Works

This paper introduces a method to make use of a public road database to identify small manned aircraft emergency landing sites. Currently, pilots do not have automation aid decision support when an emergency landing is necessary and no airport runway is reachable. Roads have historically been selected as emergency landing sites by pilots; automation can facilitate this time-critical decision. The landing site selection automation aid proposed in this work can provide a road-based landing site alternative when no runway is reachable.

Several upgrades to the proposed landing site automation aid are envisioned. First, the plan must account for other types of obstacles on the path generated or close enough to the landing site to make its use not viable. The official FAA database of obstacles can be used as a first approximation, but more details of the objects close to each road would ultimately be necessary. OpenStreetMap database and other emerging resources may provide such data.

The safety-based utility function can have a term associated with speed limit upgraded to consider the actual difference between this speed and the aircraft touchdown speed. Moreover, since the database lacks information about new obstacles, for example, ideally the pilot should maintain the largest number of landing sites at a feasible distance during the landing. The utility function could incorporate a term that would penalize isolated landing sites, i.e., landing sites that requires a path that results in all other landing sites becoming unfeasible in a short time.

The system could also be integrated to a cellular or other network that would allow two promising

upgrades to the system. First, real-time traffic could be fed into the road database in the utility function or used to prioritize or discard roads with too much traffic. Second, the system could inform authorities about the emergency and even coordinate not only rescue effort, but even short-term closure of the road for landing.

VII. Acknowledgments

This work was supported in part by CAPES-Brazil through the Science without Borders program, process number BEX 19130/12-0 and by NASA contract NNX11A078A.

References

- ¹National Transportation Safety Board, “Preliminary Report - NTSB ID ERA14LA085,” 2014.
- ²Aircraft Owners and Pilots Association, “22nd Joseph T. Nall Report - General Aviation Accidents in 2010,” 2011.
- ³Toninato, M., “NTSB Pilot Accident Report - WPR13LA209,” 2013.
- ⁴O’Connel III, M. J., “NTSB Pilot Accident Report - CEN13LA258,” 2013.
- ⁵Atkins, E. M., Portillo, I. A., and Strube, M. J., “Emergency flight planning applied to total loss of thrust,” *Journal of Aircraft*, Vol. 43, No. 4, 2006, pp. 1205–1216.
- ⁶Meuleau, N., Plaunt, C., Smith, D. E., and Smith, T. B., “An Emergency Landing Planner for Damaged Aircraft.” *Proceedings of the Twenty First Innovative Applications of Artificial Intelligence Conference*, edited by IAAI, 2009.
- ⁷Rogers, D. F., “Possible “impossible” turn,” *Journal of aircraft*, Vol. 32, No. 2, 1995, pp. 392–397.
- ⁸Siegel, D., “Development of an autoland system for general aviation aircraft,” Master Thesis, Massachusetts Institute of Technology, 2011.
- ⁹Alonso-Portillo, I. and Atkins, E. M., “Adaptive Trajectory Planning for Flight Management Systems,” *40th AIAA Aerospace Sciences Meeting and Exhibit*, AIAA, 2002, AIAA-2002-1073.
- ¹⁰Atkins, E. M., “Emergency Landing Automation Aids: An Evaluation Inspired by US Airways Flight 1549,” *AIAA Infotech@ Aerospace Conference, Atlanta, Georgia*, 2010.
- ¹¹Frantis, P., “Emergency and precautionary landing assistant,” *Digital Avionics Systems Conference (DASC), 2011 IEEE/AIAA 30th*, IEEE, 2011, pp. 6E2–1.
- ¹²Shen, Y.-F., Rahman, Z.-U., Krusienski, D., and Li, J., “A vision-based automatic safe landing-site detection system,” *Aerospace and Electronic Systems, IEEE Transactions on*, Vol. 49, No. 1, 2013, pp. 294–311.
- ¹³Floros, G., van der Zander, B., and Leibe, B., “OpenStreetSLAM: Global vehicle localization using OpenStreetMaps,” *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, IEEE, 2013, pp. 1054–1059.
- ¹⁴“OpenStreepMap,” <http://www.openstreetmap.org>.
- ¹⁵Haklay, M., “How good is volunteered geographical information? A comparative study of OpenStreetMap and Ordnance Survey datasets,” *Environment and planning. B, Planning & design*, Vol. 37, No. 4, 2010, pp. 682.
- ¹⁶Vincenty, T., “Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations,” *Survey review*, Vol. 23, No. 176, 1975, pp. 88–93.
- ¹⁷“MapQuestOpen - Open Elevation Service Developer’s Guide,” <http://open.mapquestapi.com/elevation/>.
- ¹⁸Pamadi, B. N., “General Turning Flight,” *Performance, stability, dynamics, and control of airplanes*, AIAA Education Series, American Institute of Aeronautics and Astronautics, 2004, pp. 134–146.
- ¹⁹United States of America. National Aviation and Space Administration (NASA) , *U.S. Standard Atmosphere 1976*, October 1976, pp. 1–20.
- ²⁰Donato, P. D. and Atkins, E., “Dubins Path Extension to Three-Dimensional Optimal Gliding Flight,” *Journal of Guidance, Control, and Dynamics - In preparation*.
- ²¹Umbach, D. and Jones, K. N., “A few methods for fitting circles to data,” *Instrumentation and Measurement, IEEE Transactions on*, Vol. 52, No. 6, 2003, pp. 1881–1885.