Determination of Safe Landing Zones for an Autonomous UAS using Elevation and Population Density Data

Edward Carney* , Lina Castano†, and Huan Xu‡
University of Maryland, College Park, Maryland, 20740

This work determines a safe emergency landing zone for an autonomous UAS by utilizing ground elevation, terrain ruggedness, land coverage, and population density data. The proposed system uses openly available high-fidelity Light Detection and Ranging (LIDAR) datasets and surface reflectance data from the Landsat land remote sensing project to determine potential landing zones based on current UAS position and calculated data provided by these datasets. The elevation data is represented as a set of coordinates and an elevation, with the bounding latitudes and longitudes for the dataset included in the dataset metadata. Threshold values for these data set a minimum required topographic quality for an area to be considered valid for landing. A weighting algorithm is used to determine the optimal landing zone for a given UAV situation and position. A constrained bidirectional rapidly exploring random tree algorithm is used with onetime pruning to generate a path to the chosen landing zone that obeys aircraft heading and motion constraints. We show that the optimal landing zone and corresponding path plan update based on the specified topographic quality threshold values and weighting configuration.

I. Nomenclature

LS	=	Landscan
DEM	=	Digital Elevation Model
TRI	=	Terrain Ruggedness Index
LCI	=	Land Cover Index
SSC	=	Strip Selection Criteria
WC	=	Weighting Configuration
$W(i)$	=	Calculated Score for Strip i
K_{DEM}	=	Weighting Coefficient for Average Elevation
K_{TRI}	=	Weighting Coefficient for Average Ruggedness
K_{LC}	=	Weighting Coefficient for Average Land Cover
$K_{diffDEM}$	=	Weighting Coefficient for Maximum Elevation Difference
$K_{distUAV}$	=	Weighting Coefficient for Distance from UAV
θ_{min}	=	Minimum Required Turning Angle

II. Introduction

Unmanned air vehicles (UAVs) and unmanned air systems (UAS) are of increasing relevance in civilian and military operations. Applications range from delivery and transportation of products and supplies [1], to aerial videography [2] and surveillance operations [3]. Most of these applications require the UAV to travel adjacent or close to populated areas during its mission plan. The risk to nearby populations cannot be ignored as operation of unmanned air vehicles involves potential hazards, such as vehicle component failures, loss of communication or atmospheric events, among other possible factors. Strategies to mitigate possible effects of a vehicle malfunction would increase safety of UAS operations. Finding an emergency landing zone when the vehicle is at risk of losing operability decreases hazards to nearby population and increases the possiblity of vehicle recovery. Population data can be used when choosing an

*Graduate Student, University of Maryland, Department of Aerospace Engineering
†Postdoctoral Research Associate, Aerospace Engineering, University of Maryland, College Park, and AIAA Member
‡Assistant Professor, Aerospace Engineering/Institute for Systems Research, College Park, and AIAA Member

emergency landing location, thereby decreasing the risk to the population on the vehicle's path. This safety feature contributes towards FAA requirements for inclusion of UAS into the National Airspace System [4] [5].

This work focuses on autonomously determining a safe emergency landing zone in the event of a critical vehicle failure. Geographical information about the most suitable landing locations is calculated using Light Detection and Ranging (LIDAR) datasets[6], LandScan data [7], and reflectance data from the Landsat land remote sensing project [8]. The LIDAR data is obtained in LAS datasets and is processed in order to identify potential landing strips. The final landing strip is determined based on geographical elevation, calculated terrain ruggedness, predicted land cover, and population density. The terrain ruggedness is calculated for each data entry provided by the elevation data. These data are used to remove infeasible landing strips and score every remaining potential landing strip.

Prior work in this field has made use of onboard vision-based sensors to obtain and process aerial imagery for UAV landing site determination [9] [10] [11] [12]. Methods utilizing onboard visual sensors are very accurate in that they consider the real-time position of the UAS based on imagery, but they naturally suffer from limitations imposed on any vision-based identification system. These limitations can be exacerbated by deployment on autonomous UAS platforms, as well as exposure to a variety of environmental conditions [13]. In particular, vision based landing zone determination could be greatly impaired or impossible in low-light or heavy cloud-cover situations. It could be possible to use imagery of the same region previously acquired by other manned or unmanned systems [14], but this requires that another aircraft has been to the same area, captured an image of sufficient quality, and retained that image (which may or may not be feasible depending on available on-board data storage constraints). Landing zone consideration is also limited to imaged areas; it is possible for potential landing zones to exist outside of the visual range of a UAS depending on the nature of the emergency driving the forced landing. Methods involving onboard active LIDAR sensors are limited to UAVs which have the payload capacity to support such sensors.

A similar approach for landing site determination based on previously-obtained LIDAR data has been done for single-rotor UAVs [15]. LIDAR data obtained through active sensors have been proposed for landing site determination for spacecraft [16] [17] [18]. In these approaches, LIDAR data has either been the primary factor for landing site determination, or used in conjunction with other active sensors. Previous methods which have incorporated LIDAR data, however, have not incorporated population density data in their determination of safe landing sites. This approach is acceptable for scenarios where population considerations are not applicable, but should be reconsidered for systems which will be operated on UAS in the NAS. The proposed approach considers population density in determining safe landing sites, selecting only those sites which have appropriately low population density. Additionally, other methods have used data either obtained from active onboard sensors or from private institutions, whereas all the LIDAR data used in this work is freely available to the public through the United States Geological Survey National Geospatial Program.

A significant portion of available literature involving autonomous landing site determination and landing has been focused on rotorcraft UAVs [10] [12] [19], a problem which naturally has different sets of considerations than those for fixed-wing UAVs. An additional buffer region must be considered for fixed-wing UAVs to account for the required stopping distance. The landing distance required depends on several factors, which include: the aircraft landing mass and landing speed, as well as runway characteristics such as the surface wind and temperature, the runway elevation and slope, surface conditions, and a safety factor distance. Safety factors depend on the type of aircraft, runway conditions and breaking system of the vehicle.

This work utilizes calculated ground data and population density data to determine a safe emergency landing zone for an autonomous UAS. The proposed system uses openly available high-fidelity LIDAR and solar reflectance datasets to determine potential landing zones based on current UAS position and calculated topographical quality data provided by the datasets. The topography data here consisted of elevation, ruggedness, and land cover type and was calculated from openly available datasets.

The system will incorporate these topographical quality data with population density data to select a landing location which is safe for both the UAS and the surrounding population. This work is unique in its incoporation of United States Geological Survey data and National Aeronautices and Space Adminstration (NASA) Landsat data with LandScan data for population density. Although similar work has been discussed for ensuring population safety [20] and using LIDAR and Landsat data to determine landing zones for fixed-wing UAVs based on elevation standard deviation [21], no work found in available literature that combined LandScan population density dataset with elevation, ruggedness, and cover type data to determine safe landing regions. The length of the required landing zone can be lengthened or shortened depending on the specific UAS. The system first determines which areas have a population density sufficiently low to ensure a safe landing region; a configurable safety factor is added around every potential landing zone prior to consideration for landing. The elevation, ruggedness, and land cover type datasets are then used to determine if the quality of the strip is sufficiently high to ensure a safe landing for the UAS. The resulting regions represent the potential

emergency landing zones for the UAS. All potential landing zones are then scored using configurable weightings, and the highest scoring strip determined and output from the system as the emergency landing zone. A constrained bidirectional rapidly exploring random tree (BRRT) algorithm is then used to plan a path from the current UAV position to the chosen landing strip. If there are no regions remaining after the initial consideration, the process falls back to compute a crash zone based on LandScan population data [22]. Although the crash zone will not be safe for the UAV, it will ensure safety of the surrounding population. This work integrates emergency landing zone calculation and route planning for fixed-wind UAVs with a fallback crash zone computation as found in [22], this combined implementation is unique as it could not be found in any related literature.

In a real-world application, the topographical quality data would be uploaded to the UAV prior to takeoff and be available for processing in the event an emergency landing is required. The ability to calculate a new landing zone from data already on-board the UAV allows for reaction in low-visibility situations and reduces the on-board hardware requirements for the UAV (no active visual or LIDAR system is required). Even though additional hardware may be required for the automated landing, the initial site determination can be done in visually degraded environments.

This work entails integration of ground elevation, terrain ruggedness, land coverage, and population density data together into a UAV Matlab/Simulink model to provide emergency landing areas. The selected areas are safe for both the surrounding population and the UAV. Valid routes to those landing areas are also generated. This presents a unique contribution to ongoing research of autonomous UAS operation in the National Airspace System. The remainder of this paper is structured as follows: Section III provides an overview of the emergency landing module and its integration into the UAV simulation model; Section IV reviews the landing strip identification, selection, and route planning processes; Section V presents results for several considered configurations; and Section VI provides a conclusion.

III. Emergency Landing System

An existing UAV simulation model was used in this work to supply initial conditions for UAV geographic location and altitude to the landing strip selection and route-planning systems. These systems were added to the existing simulation model through an additional emergency landing module. Section A provides an overview of the emergency landing module (Figure 1) and Section B reviews the module's integration into the existing simulation model (Figure 2).

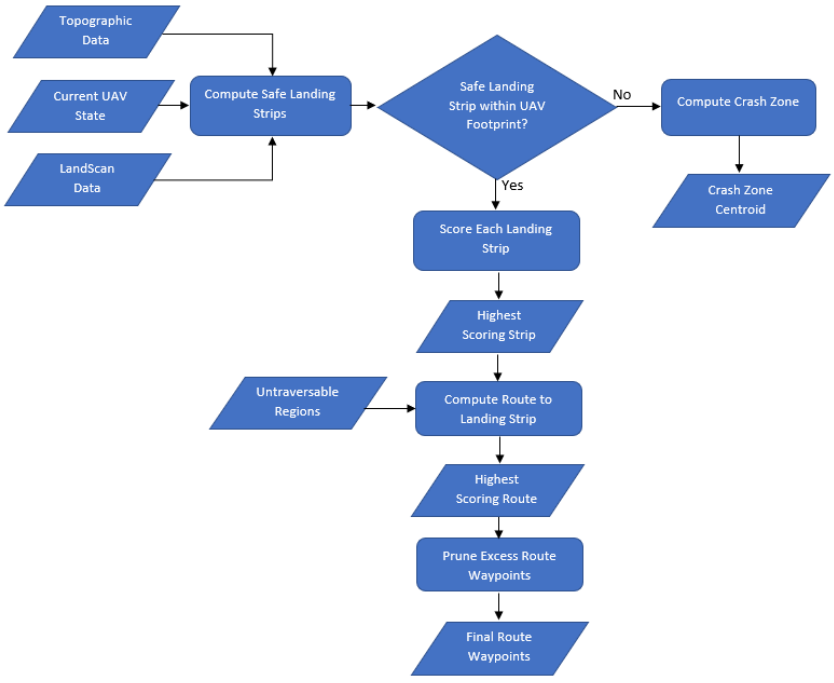


Fig. 1 Emergency Landing Module process flow: Safe landing regions are chosen using LandScan data and topographic quality data. If a suitable landing area is found the module will output a path to the optimal landing strip.

A. Emergency Landing

Aircraft can undergo a number of potential malfunctions. These include loss of a propeller, engine malfunction, loss of control over actuators and control surfaces, and structural integrity. Faults can be located at the system level, subsystem level and it can be software or hardware based. A fault is an unpermitted deviation of at least one characteristic parameter of the system from the acceptable condition. Faults can evolve into failures, which are permanent interruptions of a system's ability to perform a required function under specified operating conditions [23]. If the aircraft's ability to complete its mission has been compromised due to an imminent failure, a landing location must be chosen. Landing as soon as possible should be the first response, and therefore finding a landing location becomes the priority. The landing procedure may either be a forced landing or a preventive landing depending on the conditions of the aircraft.

This work entails development of an emergency landing module, as shown in Figure 1, that can be used on autonomous UAS, allowing the aircraft to choose a safe course of action without the intervention of a human operator. The time scale of faults and latency of communications make it difficult for an operator to be informed of the aircraft condition and take the appropriate action.

The emergency landing module will be triggered upon event detection and will compute safe regions for landing using LandScan data, after which it will use predetermined topographical quality datasets to select the best possible landing strip that meets defined quality minimums. A route is then planned from the current UAV position to the chosen landing strip using a BRRT algorithm constrained to respect heading requirements. In the event that a safe landing region cannot be found, the emergency landing module will output no landing strip and instead determine that the aircraft must select a crash location using the process documented in [22]. Figure 1 shows the process flow for the emergency landing module. A safe landing location is calculated using both geographic profiles and population density data.

B. UAV Flight Software Integration

The emergency landing module developed in this work has been interfaced with a 6-DOF flight simulation software of a small, fixed wing UAV. The simulation software was developed in Matlab/Simulink. Figure 2 shows the components of the nominal simulation and the emergency landing module. In this simulation, mission waypoints are uploaded prior to mission initiation. Then a path planning module generates a path based on mission waypoints in a queue. The desired values go into the controller block which then generates actuator commands which are sent to the aircraft dynamics block. The aircraft model uses forces and moments and incorporates environmental factors.

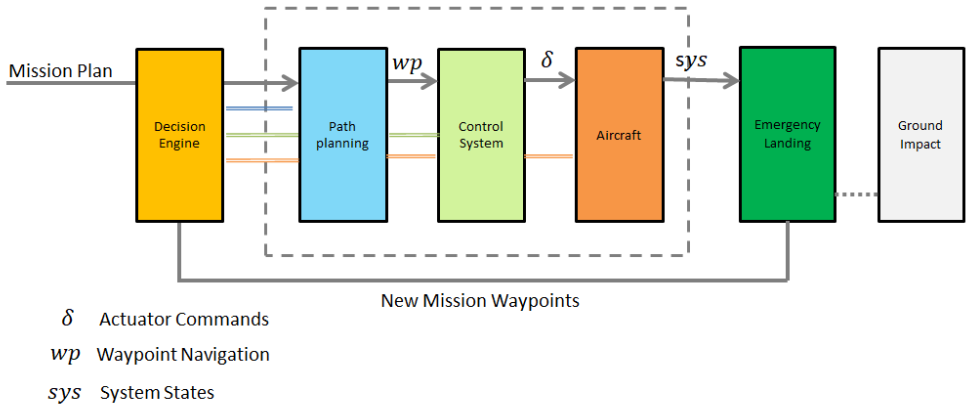


Fig. 2 UAV flight software integration with Emergency Landing module. The blocks within the dashed box correspond to the nominal UAV flight software.

All of the blocks are linked with the decision engine module, which will trigger a fault notice upon its detection. Once a fault has been identified, isolated and classified, a decision is made to land the aircraft. This will also depend on

the nature of the fault and the remaining controls of the aircraft. The emergency landing block will become active and a new set of mission waypoints will be calculated based on topographical/LandScan data.

IV. Landing Zone Determination

Landing zones were determined using a combination of LandScan, Digital Elevation Model, Terrain Ruggedness Index, and Land Cover Index data provided by Millennium Engineering and Integration Company. These data were consolidated into one dataset covering the same geographical region at the same fidelity and used to score potential landing strips to determine the optimal landing zone for the UAV. A bidirectional rapidly exploring random tree algorithm was then used to generate a route plan from the UAV's current position to the desired landing strip.

A. Available Data and Consolidation

The topographic data used to determine the most suitable landing strip for the UAV were generated from several existing datasets. It was necessary for this work to consolidate these data into one simplified structure to facilitate easy access of data values for landing strip determination and selection. Sections IV.A.1-4 provide an overview of the topographic datasets, and Section V describes how these datasets were consolidated into one simplified structure.

1. LandScan (LS) Data

The population density data used for this work was determined from LandScan USA datasets. LandScan USA incorporates several available sources of population data to provide high resolution population distribution data at approximately 90m step-sizes [24]. This data is similar to that used in [22] for determination of potential emergency crash zones for UAVs. For this work, the data was post-processed and simplified to consider only binary data values; with one corresponding to pixels that have a measured population greater than of equal to seven people, and zero corresponding to pixels that have a population less than seven people. The resulting population plot for the region considered in this work is shown in Figure 3, with yellow regions indicating areas of high population density and blue regions indicating areas of low population density.

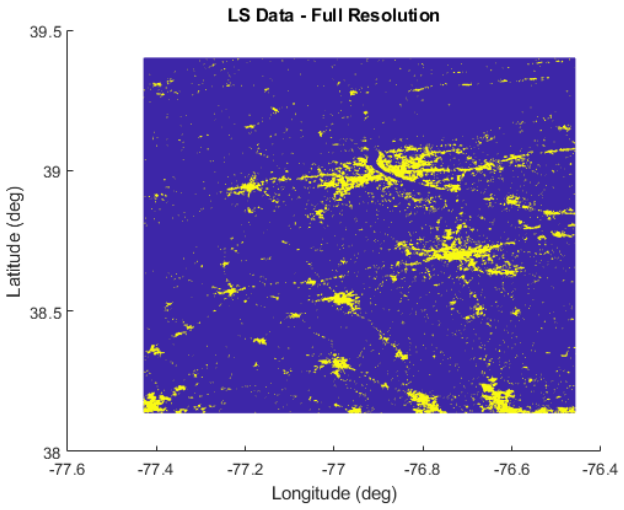


Fig. 3 Population Density Grid Plot - Full Resolution

2. Digital Elevation Model (DEM) Data

Digital elevation model data provides topographic elevation data for geographic regions; all DEM data used in this work was obtained through LIDAR source data. LIDAR is a surveying method which uses the measured return from a pulsed laser to obtain a three dimensional representation of a specified region; a laser is pulsed, the reflected light is detected by

the receiver electronics, and the time taken between the pulse and detection is used to determine distance from the laser to the target. The resultant data are an accurate set of (x,y,z) coordinates for the region, with z corresponding to the measured elevation. In general, this data can be obtained from stationary and mounted LIDAR systems. This includes systems on ground vehicles, aircraft, and spacecraft. For this work, the LIDAR data have been supplied by the 3D Elevation Program (3DEP)[6] initiative under the United States Geological Survey (USGS) National Geospatial Program (NGP), a cooperative effort between public and private agencies to enhance the available high-quality topographic data for the contiguous United States, Hawaii, and Alaska. As of this paper, the 3DEP is still actively collecting data; these data are all freely available from USGS via The National Map, an interactive map which allows the download of LIDAR data for a specified region. All LIDAR data utilized for this work was obtained from The National Map and is contained in LASER (LAS) datasets, per the minimum specification for LIDAR data for the 3DEP.

LAS datasets are a collection of one or more LAS files, which is an industry-standard method of storing LIDAR data. Included for each region are the x,y,z coordinates for every collected data point. These points can be individually plotted to obtain a three dimensional point cloud representation of the area of interest (AOI). The LAS datasets downloaded from The National Map contain metadata which includes the bounding latitude and longitude lines for the AOI, from which the x,y,z coordinates are referenced. Figure 4 shows a LIDAR point cloud with 10 meter resolution. In this graph, the elevation is color coded such that areas of higher elevation are shown in yellow and those with lower elevation are shown in blue.

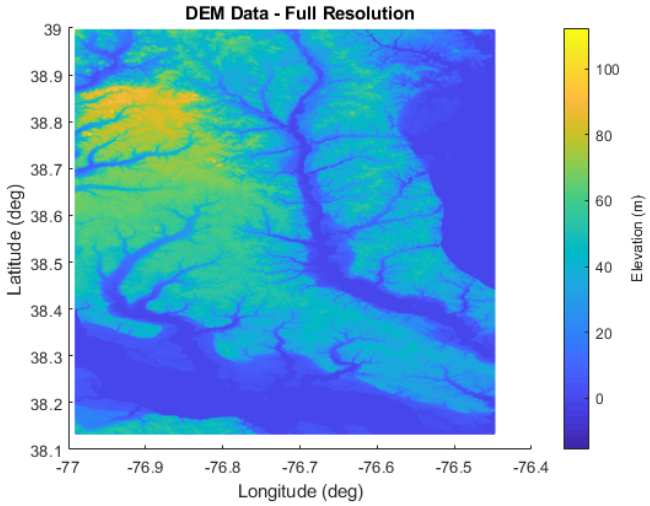


Fig. 4 Digital Elevation Model Grid Plot - Full Resolution

3. Terrain Ruggedness Index (TRI) Data

The terrain ruggedness index (or topographic ruggedness index) provides a simple quantifiable metric with which to judge terrain heterogeneity. TRI data are determined from a provided DEM dataset; a TRI value is calculated for each DEM data point as the mean difference in elevation between the current cell and the eight immediately surrounding cells. For a given data point i, j this is determined through the following equation [25],

$$TRI(i,j) = \frac{\sum_{p=i-1}^{i+1} \sum_{q=j-1}^{j+1} |DEM(p,q) - DEM(i,j)|}{8}. \tag{1}$$

As a result of basing the TRI computation on available DEM data, the TRI and DEM data will have the same resolution and step size. For the purpose of this work, as well as in the supporting literature, the DEM data were provided in meters, thus the corresponding TRI data were also measured in meters. These data are then typically grouped into the following classifications: level = 0 - 80 m; nearly level = 81 - 116 m; slightly rugged = 117 - 161 m; intermediately rugged = 162 - 239 m; moderately rugged = 240 - 497 m; highly rugged = 498 - 958 m; and extremely rugged = 959 - 4367 m [26]. For the purposes of this work in determining potential emergency landing zones for fixed-wing UAVs, lower TRI values were preferred, and even the most stringent classification of level contained TRI values that exceeded

desired levels for terrain ruggedness. As such, the classification was further restricted. A plot of calculated TRI data corresponding to the above DEM data from Figure 4, is shown in Figure 5.

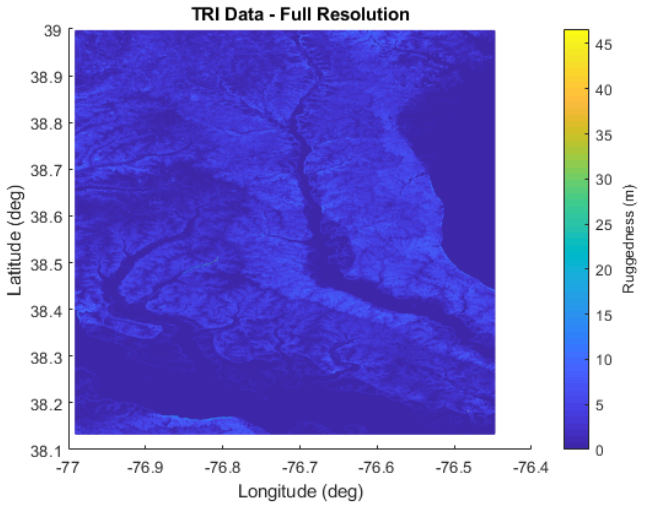


Fig. 5 Terrain Ruggedness Index Grid Plot - Full Resolution

4. Land Cover Index (LCI) Data

Land cover index data provides a classification system for geographic regions based on post-processing of publicly available Landsat surface reflectance data, which used the USGS Land Use and Land Cover (LULC) system as a bases. Table 1 shows the various classification types used for this work, the corresponding LC integer values, and which classifications were considered for potential landing zones. Figure 6 shows the corresponding plot of LCI values color-coded so that lower values appear more blue and higher values are more yellow. Due to the high number of unclassified data points in this set, the unclassified land type was accepted as a viable landing cell; only cells that were known to be of a land type that presented a higher risk to the UAV were excluded. Generally, lower viable LCI values corresponded to more preferable land classifications.

Land Classification	Integer	Viable Landing Cell
Unclassified	0	True
Water	1	False
Bare Earth	2	True
Crop Field	3	True
Vegetation	4	True
Dense Vegetation	5	False
Building	6	False
Built-Up Surface	7	True

Table 1 Land Cover Classification Types

5. Data Consolidation

The DEM, TRI, LC, and LS data were provided in the form of four three-dimensional matrix sets, one for each data type. Each three-dimensional matrix consisted of three stacked two-dimensional matrices; the first two-dimensional matrix held the pertinent DEM, TRI, LC, or LS data values, and the other two matrices held the latitude and longitude information for that data. Due to the various sources used to obtain the different data, the coverage area and data

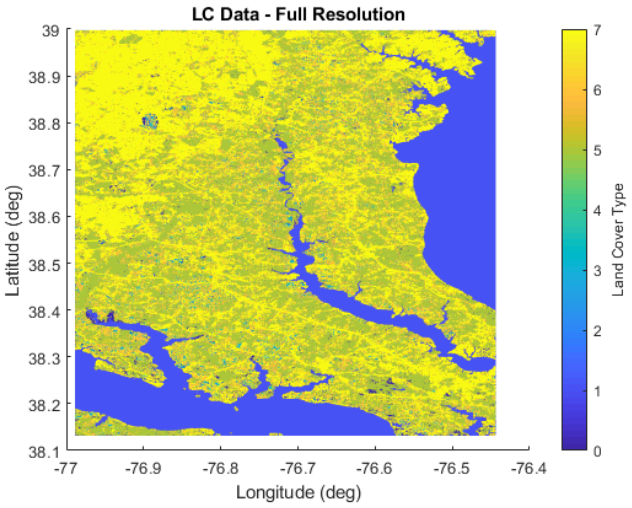
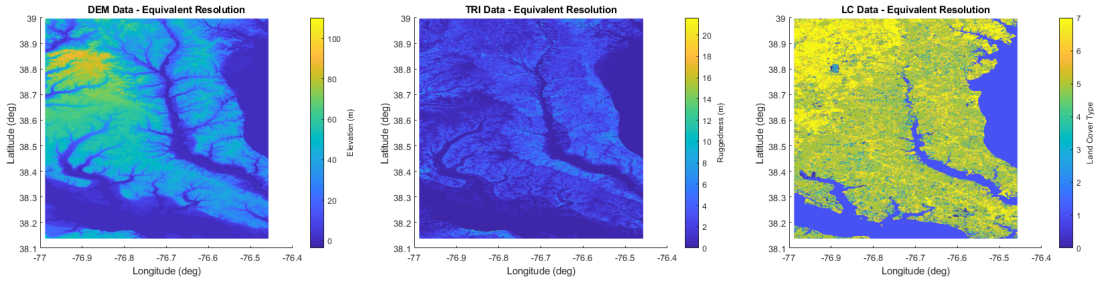


Fig. 6 Land Classification Index Grid Plot - Full Resolution

fidelity varied for each matrix set. To allow for simplified integration into the landing zone determination process, these data were pruned so that all values would fall within the same coverage area; two-dimensional linear interpolation was then used to set all matrices to have the same data fidelity. The step sized used for the linear interpolation was based on the fidelity of the LandScan data. This was done to maintain the integrity of the LandScan data and limit the possibility of any data loss. The primary objective of the system is to avoid populated zones, which are designated in the LandScan matrix as ones; using the step-size set by the LandScan matrix prevented modification of this data during the interpolation. This helps ot ensure the system still meets its primary objective of avoiding populated regions in the event of an emergency UAV landing. The resulting datasets for DEM, TRI, and LC can be seen in Figures 7a, 7b, and 7c, respectively.



(a) Digital Elevation Model **(b) Terrain Ruggedness Index** **(c) Land Classification Index**

Fig. 7 Data Grid Plots - Equivalent Resolution

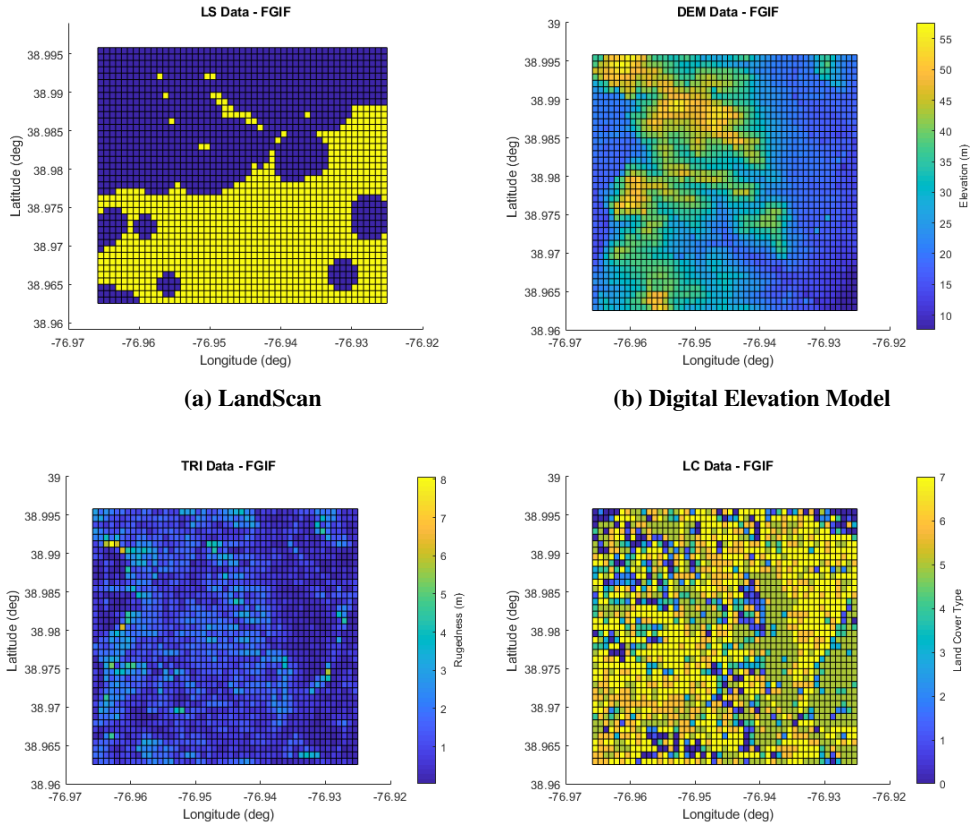
The resulting datasets covered the same geographic region and had the same resolution as the LandScan data shown in Figure 3. The matrices for this data were then concatenated into one three-dimensional principal matrix, which consisted of six stacked two-dimensional matrices. The first four matrices contained the LS, DEM, TRI, and LC data, and the other two matrices contained the latitude and longitude information for that data. This allowed for trivial access of data at any given latitude and longitude.

B. Landing Zone Calculation

The determination of a viable landing zone is comprised of the following three parts: (1) Data reduction, (2) Landing strip identification, (3) Strip weighting and selection.

1. Data Reduction

The area covered by the LS, DEM, TRI, and LC datasets was sufficiently large, and the step-size sufficiently small, to result in significant processing times without data reduction. This could result in processing times on the order of minutes for the areas and machines used in this work. Additionally, the dataset coverage area could be substantially larger than the footprint of potential landing zones reachable by the UAV in an emergency landing situation; therefore, determining landing zones for the entire dataset coverage area could result in a large number of unreachable landing zones. To mitigate this, the principal matrix, which consisted of the available LS, DEM, TRI, and LC datasets, was pruned prior to processing so that the coverage area was based on the feasible ground impact footprint calculated at the time of fault detection [22]. An example of this reduced coverage area for the LS, DEM, TRI, and LC datasets is shown in Figures 8a-d.



(a) LandScan **(b) Digital Elevation Model**

(c) Terrain Ruggedness Index **(d) Land Classification Index**

Fig. 8 Data Grid Plots - Feasible Ground Impact Footprint

2. Landing Strip Identification

The reduced data were then used to determine viable landing zones. Every cell of the reduced principal matrix was surveyed, and regions which violated defined constraints of population, elevation, ruggedness, or land cover, were dropped from consideration. The exact constraint values could be updated to allow for more stringent or relaxed strip selection criteria. Specifically, two cases were considered: (1) cells were removed from consideration if they had a boolean population value of one, a DEM value greater than 25% of the UAV's current altitude, a TRI value above one meter, or a land cover value equal to one or six, (2) cells were removed from consideration if they had a boolean population value of one, a DEM value greater than 50% of the UAV's current altitude, a TRI value above two meters, or a land cover value equal to one or six. These cases demonstrate how the number of potential landing zones increases when the strip selection criteria are relaxed to allow for higher ruggedness or elevation.

Each valid grid square was then checked for adjacency to other valid grid squares. The dimensions of the required

landing zone is a configurable input to the landing strip identification process; for the small UAVs considered in this work, a strip one grid cell wide and three grid cells long was deemed sufficient for landing, equivalent to approximately a 90 meter by 270 meter landing zone. A safety margin of one grid square was added to this strip, resulting in each potential landing zone requiring minimum dimensions of three 90 meter grid squares in width and five 90 meter grid squares in length. All possible combinations of these dimensions were checked for each grid square. If these dimensions could be satisfied in any direction along the square, that set of grid squares was retained as a potential landing zone. A final pruning process was done after landing strips were identified to eliminate any duplicate landing zones.

3. Strip Weighting and Selection

Once potential emergency landing zones were determined, an individual strip was selected and output as the result from the analysis. The criteria for strip selection was based on a weighting algorithm, which determined a net score for each strip based on the average strip DEM, TRI, and LC values, the statistical range of strip DEM values, and the distance of the strip's centroid from the current UAV position.

Lower average DEM values were considered more desirable as they provide the UAV with additional time to maneuver due to the increased amount of time required to descend to ground level. Lower average TRI values were considered more desirable as they indicated a more level landing strip and a reduction in potential damage to the UAV when landing. Lower LC values were also considered more desirable, so that the values were preferred in order 2, 3, 4, 5, 7, corresponding to bare earth, crop field, vegetation, dense vegetation, and built-up surface, respectively; similar to TRI data, lower values of LCI indicated land types that would reduce potential damage to the UAV. Preference for lower values for the statistical range of DEM values was intended to reduce to the potential that a strip may have a single high elevation feature (such as a tree or rock outcrop) that could cause potentially catastrophic damage to the UAV; such a feature may not be apparent in the average DEM value for the strip, but would result in a larger statistical range and thus a lower net score. Smaller values for the distance between the UAV and the strip centroid were considered more desirable as closer landing strips allowed for shorter landing time and reduced flight time in a potentially faulty state.

Using the above definitions, for strip i the weighting $W(i)$ is determined as:

$$\begin{aligned} W(i) = \big[&K_{DEM} * DEM_{avg}(i) + K_{TRI} * TRI_{avg}(i) + K_{LC} * LC_{avg}(i) \\ &+ K_{diffDEM} * DEM_{maxDiff}(i) + K_{distUAV} * distanceFromUAV(i)\big], \end{aligned} \tag{2}$$

where

$$DEM_{maxDiff}(i) = |max(DEM(i)) - min(DEM(i))|, \tag{3}$$

and K_{DEM}, K_{TRI}, K_{LC}, $K_{diffDEM}$, and $K_{distUAV}$ correspond to the defined weightings for the average DEM, TRI, and LCs value for the strip, the range of strip DEM values, and the distance of the strip's centroid from the current UAV position, respectively. Strip scoring was only done for strips that met the strip selection criteria discussed in the Landing Strip Identification section. In addition to the two sets of strip selection criteria considered for this work, two weighting configurations were used to score the potential landing zones: (1) weightings with higher relative values for K_{DEM}, K_{TRI}, K_{LC}, and $K_{diffDEM}$ compared to $K_{distUAV}$, and (2) weightings with a higher relative value for $K_{distUAV}$ compared to the other weightings. This is intended to correspond to two potentially different fault scenarios with distinct needs. The first weighting configuration represents a scenario which prioritizes the topographic quality of the landing strip over the distance to the strip; this is intended to align with a scenario where the fault condition does not demand immediate grounding of the aircraft, such as an engine-out scenario, where the quality of the landing strip can be considered a significant factor to reduce potential damage to the UAV. The second weighting configuration represents a scenario which prioritizes the proximity of the landing strip over the topographic quality; this is intended to align with a scenario where the fault condition demands immediate grounding of the aircraft, such as a structural failure or partial loss of control, where grounding the aircraft as soon as possible is the highest priority to reduce the potential risk for inadvertent ground impact in a populated region. In both cases, the strip with the highest score was chosen as the optimal landing zone and output as a set of waypoints containing latitude and longitude data.

C. Path-Planning

Once an optimal landing strip was determined through the weighting algorithm, path-planning was done to determine a route from the UAV current position to the landing strip. This was accomplished using a constrained bi-directional

rapidly exploring random tree algorithm (BRRT), which allowed for rapid exploration of the freespace defined by the chosen landing strip and the UAV position.

RRT and BRRT algorithms have been used for a variety of path-planning situations, including obstacle avoidance for fixed-wing UAVs [27] [28], and landing route planning for quad-rotor UAVs [29] and fixed-wing UAVs [30]. The RRT algorithm generates a probabilistic mapping of a defined freespace and provides a guaranteed collision-free path given appropriately defined global start point, global goal point, and global untraversable regions. For this purely geometric use of the algorithm, the resultant path is defined as a series of nodes with positional latitude and longitude information, and path segments of defined length connecting the nodes. This path is created by generating a random node q_{rand} within the freespace, determining the closest existing node q_{close}, extending some delta distance from q_{close} to q_{rand}, generating a new point q_{new} at the end of this extension (forming new path segment $\underline{l}_{new}$), then verifying point q_{new} and path segment $\underline{l}_{new}$ do not fall within or intersect any global untraversable regions or violate any motion constraints; this process continues iteratively until a path from the global start point to the global goal point is attained. For an RRT algorithm, this process only takes place from the global start point, so all paths originate at that point; for the BRRT algorithm used in this work, path segments also grow from the global goal point. There is an additional step to every iteration for the BRRT where the same process occurs, but from the global goal point; so every iteration requires finding the closest nodes from the global start point and global goal point and attempting to build a new path segment for each. In this way, the paths from the global start and goal points will grow towards each other naturally. Figure 9 and Algorithm 1 show a graphic and pseudo-code for the procedure to add a new point.

Algorithm 1 BRRT Algorithm

```
1:  procedure AddPoint
2:      qRand ← randomly generated point
3:      qCloseStart ← closestStartNode(qRand)
4:      lNewStart ← extend from qCloseStart to qRand
5:      qNewStart ← new point at end of lNewStart
6:      if isValid(qNewStart) AND isValid(lNewStart) then
7:          addPointStart(qNewStart)
8:      qCloseGoal ← closestGoalNode(qRand)
9:      lNewGoal ← extend from qCloseGoal to qRand
10:     qNewGoal ← new point at end of lNewGoal
11:     if isValid(qNewGoal) AND isValid(lNewGoal) then
12:         addPointStart(qNewGoal)
```

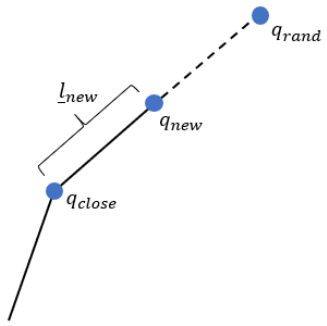
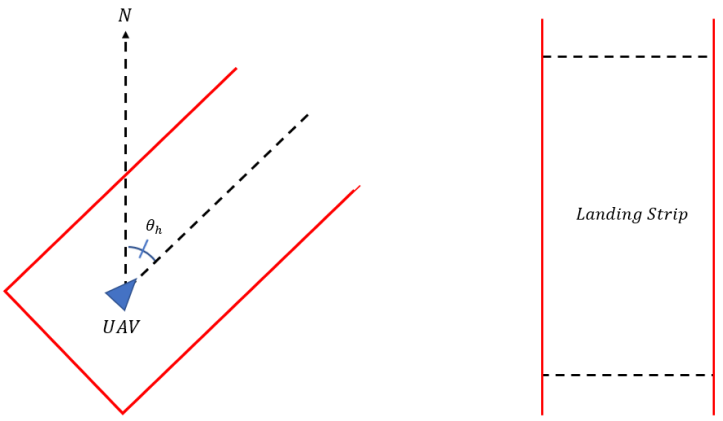


Fig. 9 RRT process for adding a new point.

Untraversable regions were defined by a series of waypoints forming the boundary for the region; the algorithm would not consider any new points that existed inside of or on on the boundary of any of these regions or resulted in a path segment intersecting any of these regions. If the waypoints for a region did not form a closed polygon, the check determining if the new point existed inside of the region would be omitted. Such open polygons were used to impose heading constraints on the path for initial motion from the start point and on approach to the landing zone. The expected behavior being that the UAV would have limited options available to it for an initial maneuver depending on its current heading, and that it must approach the landing strip so that it lands along the long dimension of the strip.

The constraint for the initial maneuver was imposed using a three-sided untraversable polygon around the UAV current position, essentially forming an open rectangle build around the UAV. The rotation of this open rectangle would be such that the open end would align with the current UAV heading. The path was not permitted to intersect any closed side of the rectangle, thus forcing the path to grow out of the open end of the rectangle corresponding to the current UAV heading. A graphic of the heading constraint can be seen in Figure 10a, where the red lines represent the untraverasble open rectangle imposed for a UAV heading of $\theta_h = 45$ deg. For this work, heading is measured as degrees clockwise from north. The constraint for landing approach was imposed using two untraversable line segments bounding the long sides of the rectangular landing strip; this left the short sides of the landing strip open for path planning. This forced the path to approach the center of the landing strip along the long dimension of the strip. A graphic of the landing strip constraint can be seen in Figure 10b, where the solid lines represent the untraversable line segments imposed along the long dimension of the landing strip. These two constraints were sufficient to require the constructed path to start at or near the heading of the UAV and end by approaching the landing strip along its long sides.

Additional imposed motion constraints were set such that the angle formed by the original path segment containing



(a) Heading constraint. **(b) Landing strip constraint.**

Fig. 10 Imposed motion constraints for a UAV heading of 45 degrees and a vertical landing strip.

q_{new} and the new path segment l_{new} was greater than a defined minimum angle θ_{min}. This prohibited waypoint plans that would require excessively sharp or sudden motion for the UAV. This constraint was intended to represent the more limited motion capabilities of fixed-wind UAVs as compared to rotorcraft. Figure 11 shows a graphic of this constraint check.

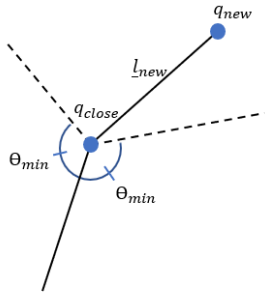
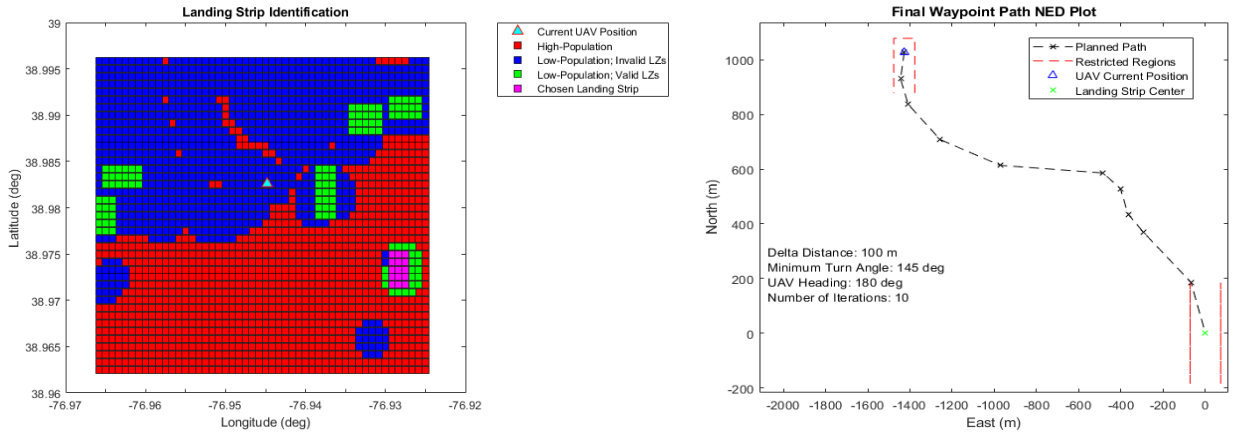


Fig. 11 Minimum angle check for BRRT algorithm.

This process of path generation was repeated for a defined number of iterations, so that the number of complete paths equaled the number of iterations; for this work, the number of iterations was fixed at 10. For each iteration the new path was scored based on its overall distance; shorter overall distance corresponded to higher scores. More iterations yielded a more optimal path (shorter overall distance) at the expense of longer runtimes. After the BRRT algorithm iterations completed, a one time pruning algorithm was run on the highest scoring path and the resulting pruned path was then output from the BRRT algorithm. This pruning removed excess points by iteratively attempting to connect two points directly and skip any intermediary points; if a valid path could be drawn between the two points, then any intermediate points were removed from the path. This process was repeated until no additional points could be removed without the path entering untraversable regions or violating motion constraints. The resultant path was represented as a series of sequential waypoints which could then be input into the path planning module for the aircraft model, which would generate the necessary flight segments between waypoints.

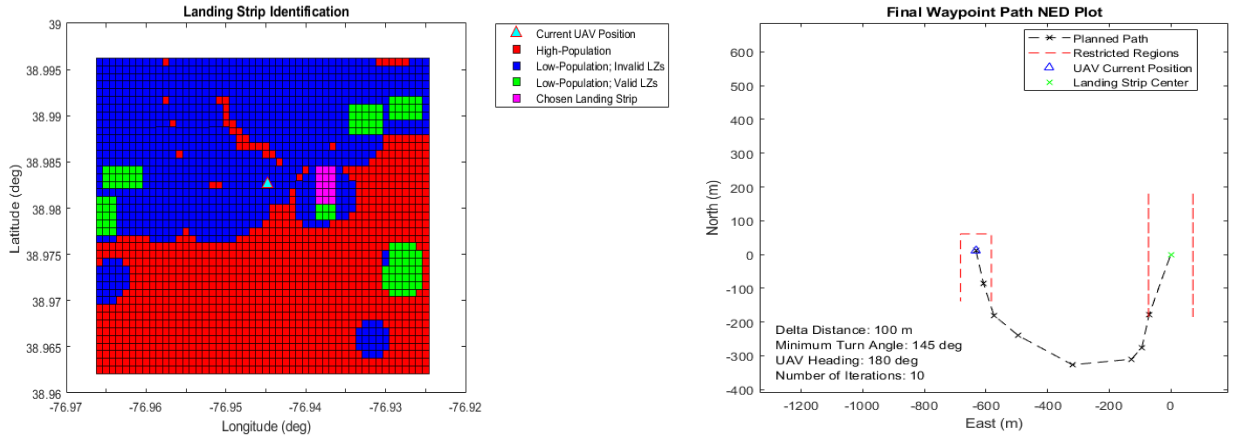
V. Results

Results from the analysis can be seen below for the two different cases considered for landing strip selection criteria (SSC) and weighting configurations (WC). Restricted SSC (Figures 12-13) corresponds to criteria (1) and relaxed SSC (Figures 14 -15) corresponds to criteria (2), as discussed in section IV.B.2. The strip quality WC corresponds to configuration (1) and the strip proximity WC corresponds to configuration (2), as discussed in section IV.B.3. For each case, plots are shown for the landing strip identification and resulting BRRT path plan.



(a) Landing Strip Identification. **(b) Final Waypoint Plan.**

Fig. 12 Results for Restricted SSC, Strip Quality WC.

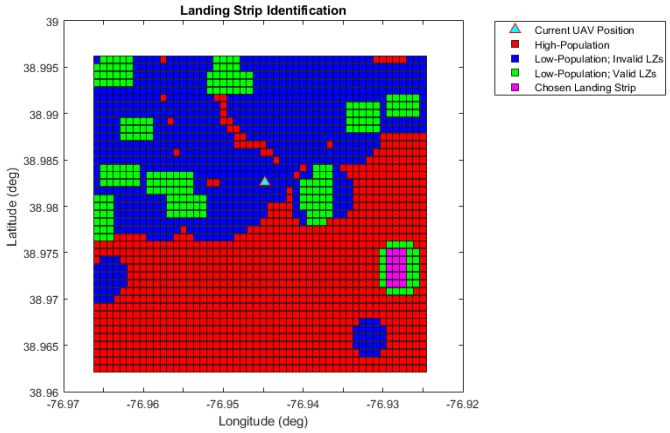


(a) Landing Strip Identification. **(b) Final Waypoint Plan.**

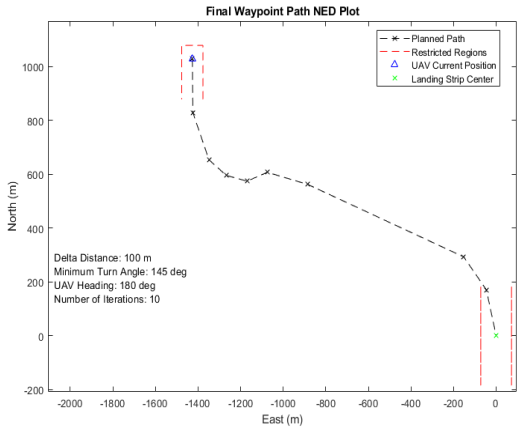
Fig. 13 Results for Restricted SSC, Strip Proximity WC.

In the landing strip identification plots, red tiles denote areas of high population as per the LandScan data, and blue tiles denote areas of low population which did not meet the specified SSC. Green tiles denote areas of low population which did meet the specified SSC, and purple tiles represent the highest-scoring landing strip based on the specified WC. The UAV is represented on the plot as a light blue triangle. In the final waypoint plan plot, the route generated by the BRRT algorithm is denoted as a series of black cross markers (representing the route waypoints) connected by a dashed black line. The red dashed lines denote the untraversable boundary lines, the green cross mark denotes the center of the landing strip, and the blue triangle denotes the UAV. The untraversable boundary lines shown the in the final waypoint plots align with the heading constraints discussed in the previous path planning section, as described in Figures 10a and 10b. Informational text is also displayed on the final waypoint plots, showing the delta distance, minimum turn

angle, UAV starting heading, and number of algorithm iterations used for each case. These values were kept constant in the cases considered here to more accurately demonstrate the differences in the results caused by the SSC and WC conditions.

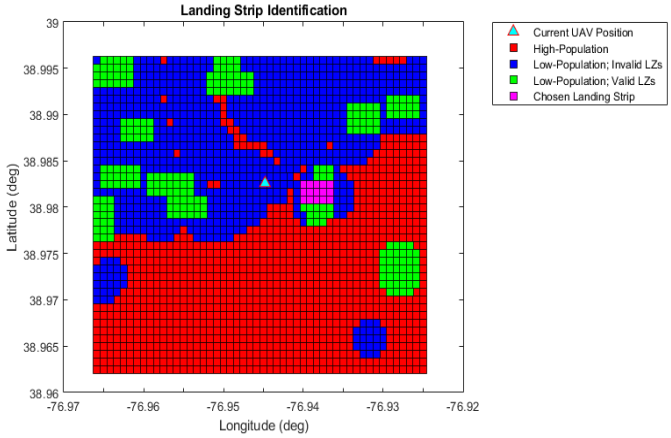


(a) Landing Strip Identification.

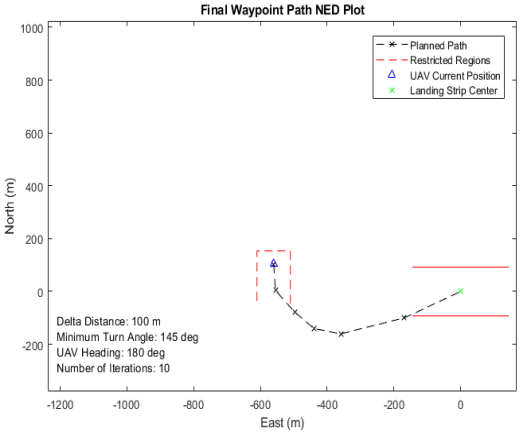


(b) Final Waypoint Plan.

Fig. 14 Results for Relaxed SSC, Strip Quality WC.



(a) Landing Strip Identification.



(b) Final Waypoint Plan.

Fig. 15 Results for Relaxed SSC, Strip Proximity WC.

It can be seen that the landing strip identification results for the restricted SSC cases (Figures 12a and 13a) have fewer green tiles, and therefore fewer valid landing strips than the relaxed SSC cases (Figures 14a and 15a). This aligns with expectations, as the more relaxed SSC case yields more valid landing strips, but of potentially lower quality than the restricted SSC. It is also apparent in the strip identification results that the strip quality WC cases (Figures 12a and 14a) result in different highest-scoring landing strips than the strip proximity WC cases (Figures 13a and 15a); it is clear that the highest-scoring strip for the proximity WC cases is significantly closer to the current UAV position than the highest-scoring strip for the strip quality WC cases.

It is interesting to note that the highest-scoring strip for the strip quality WC cases is the same regardless of the SSC condition (Figures 12a and 14a). This result aligns with expectations, as the additional landing strips made available by the relaxed SSC will be of lower topographic quality due to the relaxed constraints; therefore, none of these newly available, lower quality strips should be expected to be the highest-scoring strip for the strip quality WC. It is, however, possible that the additional valid strips in the relaxed SSC cases could result in a potential landing zone closer to the

UAV than what was available in the restricted SSC cases. This can be seen by comparing the strip proximity WC cases (Figures 13a and 15a). The relaxed SSC allowed for additional valid strips closer to the UAV, resulting in a different highest-scoring strip than that calculated for the restricted SSC.

In all cases considered here, the waypoint plan generated by the BRRT algorithm can be seen to follow the heading constraints required at both the start of the UAV path and on approach to the landing strip.

VI. Conclusion

The integration of these elements together into a UAV Matlab/Simulink model to provide emergency landing areas which are safe for both the surrounding population and the UAV, as well as generation of valid routes to those landing areas, presents a unique contribution to ongoing research of autonomous UAS operation in the National Airspace System.

Future work will expand on this system to incorporate additional untraversable regions based on geographic properties of the area surrounding the UAV and chosen landing strip. This would involve defining regions of high elevation that the UAV could not travel over in its fault state and supplying these as input to the path planning module. The current path planning algorithm can also be enhanced to utilize three-dimensional path planning; this would include a separate module to plan for the final landing flight segment.

Acknowledgments

The authors would like to thank Millennium Engineering and Integration Company for providing the LandScan, Digital Elevation Model, Terrain Ruggedness Index, and Land Cover Index data. They would especially like to thank Andrew Poissant and Adam McKee for their help and support.

References

[1] Stolaroff, J., "The need for a life cycle assessment of drone-based commercial package delivery," Tech. rep., Lawrence Livermore National Laboratory (LLNL), Livermore, CA, 2014.

[2] JONES IV, G. P., Pearlstine, L. G., and Percival, H. F., "An assessment of small unmanned aerial vehicles for wildlife research," *Wildlife Society Bulletin*, Vol. 34, No. 3, 2006, pp. 750–758.

[3] Bürkle, A., "Collaborating miniature drones for surveillance and reconnaissance," *Unmanned/Unattended Sensors and Sensor Networks VI*, Vol. 7480, International Society for Optics and Photonics, 2009, p. 74800H.

[4] Dalamagkidis, K., Valavanis, K. P., and Piegl, L. A., "On unmanned aircraft systems issues, challenges and operational restrictions preventing integration into the National Airspace System," *Progress in Aerospace Sciences*, Vol. 44, No. 7-8, 2008, pp. 503–519.

[5] Consiglio, M. C., Chamberlain, J. P., Munoz, C. A., and Hoffler, K. D., "Concepts of Integration for UAS Operations in the NAS," 2012.

[6] Sugarbaker, L. J., Constance, E. W., Heidemann, H. K., Jason, A. L., Lukas, V., Saghy, D. L., and Stoker, J. M., "The 3D Elevation Program initiative: a call for action," Tech. rep., US Geological Survey, 2014.

[7] Bhaduri, B., Bright, E., Coleman, P., and Urban, M. L., "LandScan USA: a high-resolution geospatial and temporal modeling approach for population distribution and dynamics," *GeoJournal*, Vol. 69, No. 1-2, 2007, pp. 103–117.

[8] "Landsat Surface Reflectance Level-2 Science Products," , Oct 2018. URL `https://landsat.usgs.gov/landsat-surface-reflectance-data-products`.

[9] Patterson, T., McClean, S., Morrow, P., Parr, G., and Luo, C., "Timely autonomous identification of UAV safe landing zones," *Image and Vision Computing*, Vol. 32, No. 9, 2014, pp. 568 – 578. doi:https://doi.org/10.1016/j.imavis.2014.06.006.

[10] Lange, S., Sünderhauf, N., and Protzel, P., "Autonomous landing for a multirotor UAV using vision," *International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAR 2008)*, 2008, pp. 482–491.

[11] Fitzgerald, D., Walker, R., and Campbell, D., "A vision based forced landing site selection system for an autonomous UAV," *Intelligent Sensors, Sensor Networks and Information Processing Conference, 2005. Proceedings of the 2005 International Conference on*, IEEE, 2005, pp. 397–402.

[12] Johnson, A., Montgomery, J., and Matthies, L., "Vision Guided Landing of an Autonomous Helicopter in Hazardous Terrain," *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005, pp. 3966–3971. doi:10.1109/ROBOT.2005.1570727.

[13] Von Bueren, S., Burkart, A., Hueni, A., Rascher, U., Tuohy, M., and Yule, I., "Deploying four optical UAV-based sensors over grassland: challenges and limitations," *Biogeosciences*, Vol. 12, No. 1, 2015, p. 163.

[14] Obermeyer, K., "Path planning for a UAV performing reconnaissance of static ground targets in terrain," *AIAA Guidance, Navigation, and Control Conference*, 2009, p. 5888.

[15] Scherer, S., Chamberlain, L., and Singh, S., "Autonomous landing at unprepared sites by a full-scale helicopter," *Robotics and Autonomous Systems*, Vol. 60, No. 12, 2012, pp. 1545 – 1562. doi:https://doi.org/10.1016/j.robot.2012.09.004, URL http://www.sciencedirect.com/science/article/pii/S0921889012001509.

[16] Johnson, A. E., Klumpp, A. R., Collier, J. B., and Wolf, A. A., "Lidar-Based Hazard Avoidance for Safe Landing on Mars," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 6, 2002, pp. 1091 – 1099. doi:10.2514/2.4988.

[17] Johnson, A. E., Huertas, A., Werner, R. A., and Montgomery, J. F., "Analysis of On-Board Hazard Detection and Avoidance for Safe Lunar Landing," *2008 IEEE Aerospace Conference*, 2008, pp. 1–9. doi:10.1109/AERO.2008.4526301.

[18] Serrano, N., Bajracharya, M., Howard, A., and Seraji, H., "A novel tiered sensor fusion approach for terrain characterization and safe landing assessment," *2006 IEEE Aerospace Conference*, 2006, pp. 10 pp.–. doi:10.1109/AERO.2006.1655795.

[19] Lange, S., Sunderhauf, N., and Protzel, P., "A vision based onboard approach for landing and position control of an autonomous multirotor UAV in GPS-denied environments," *2009 International Conference on Advanced Robotics*, 2009, pp. 1–6.

[20] Di Donato, P. F. A., and Atkins, E. M., "Evaluating Risk to People and Property for Aircraft Emergency Landing Planning," *Journal of Aerospace Information Systems*, Vol. 14, No. 5, 2017, pp. 259–278. doi:10.2514/1.I010513.

[21] Garg, M., Kumar, A., and Sujit, P. B., "Terrain-based landing site selection and path planning for fixed-wing UAVs," *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2015, pp. 246–251.

[22] Poissant, A., Castano, L., and Xu, H., "Ground Impact and Hazard Mitigation for Safer UAV Flight Response," *International Conference on Unmanned Aircraft Systems (ICUAS 2018)*, 2018.

[23] Isermann, R., "Terminology in fault detection and diagnosis," *Fault-Diagnosis Applications*, Springer, 2011, pp. 321–323.

[24] Bhaduri, B., Bright, E., Coleman, P., and Urban, M. L., "LandScan USA: a high-resolution geospatial and temporal modeling approach for population distribution and dynamics," *GeoJournal*, Vol. 69, No. 1, 2007, pp. 103–117. doi: 10.1007/s10708-007-9105-9, URL https://doi.org/10.1007/s10708-007-9105-9.

[25] Warren, M., Mejias, L., Yang, X., Arain, B., Gonzalez, F., and Upcroft, B., *Enabling Aircraft Emergency Landings Using Active Visual Site Detection*, Springer International Publishing, Cham, 2015, pp. 167–181. doi:10.1007/978-3-319-07488-7_12, URL https://doi.org/10.1007/978-3-319-07488-7_12.

[26] Riley, S. D. D., S. J., and Elliot, R., "A terrain ruggedness index that quantifies topographic heterogeneity, Intermountain Journal of Sciences," *Intermountain Journal of Sciences*, Vol. 5, No. 1-4, 1999, pp. 23–27.

[27] Meng, L., Qing, S., and Jun, Z. Q., "UAV path re-planning based on improved bidirectional RRT algorithm in dynamic environment," *2017 3rd International Conference on Control, Automation and Robotics (ICCAR)*, 2017, pp. 658–661. doi:10.1109/ICCAR.2017.7942779.

[28] Bhandari, S., Farinella, J., and Lay, C., AIAA SciTech Forum, American Institute of Aeronautics and Astronautics, 2016, Chaps. UAV Collision Avoidance using a Predictive Rapidly-Exploring Random Tree (RRT). doi:10.2514/6.2016-2197, URL https://doi.org/10.2514/6.2016-2197, 0.

[29] Dong, Y., Fu, C., and Kayacan, E., "RRT-based 3D path planning for formation landing of quadrotor UAVs," *2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 2016, pp. 1–6. doi:10.1109/ICARCV.2016.7838567.

[30] Garg, M., Kumar, A., and Sujit, P. B., "Terrain-based landing site selection and path planning for fixed-wing UAVs," *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2015, pp. 246–251. doi:10.1109/ICUAS.2015.7152297.