

# WHERE TO LAND

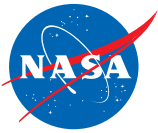
## A Reachability Based Forced Landing Algorithm for Aircraft Engine Out Scenarios

Jinu Idicula - NASA Armstrong

Kene Akametalu, Mo Chen, Claire Tomlin, Jerry Ding - UC Berkeley

Loyd Hook - UTulsa

11/17/2015



# Outline

1. Where To Land (WTL)
2. WTL1 → WTL2
3. Engine Out Case
4. Aircraft Reachability
5. Cost Map Development
6. Dynamics Model
7. NASA TCM Model
8. Optimal Trajectory Generation
9. WTL2 C code
10. Test Cases
11. Hardware in the Loop (HIL) Simulation
12. Future Work



# WTL Team

 <p>UC Berkeley</p>	<ul style="list-style-type: none"><li>- Algorithm Design</li><li>- Reachable Sets</li><li>- Hybrid Mode Switching</li></ul>
 <p>NASA Armstrong</p>	<ul style="list-style-type: none"><li>- WTL C Code</li><li>- S/W V&amp;V</li><li>- HIL Simulation</li></ul>
 <p>U. Tulsa</p>	<ul style="list-style-type: none"><li>- NYC Cost Map</li><li>- S/W Requirements</li></ul>

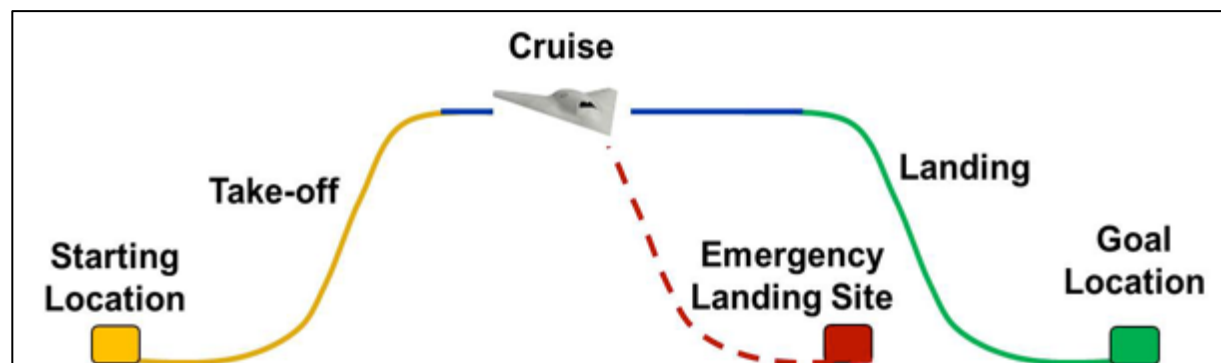
# Emergency Landings





# Where To Land

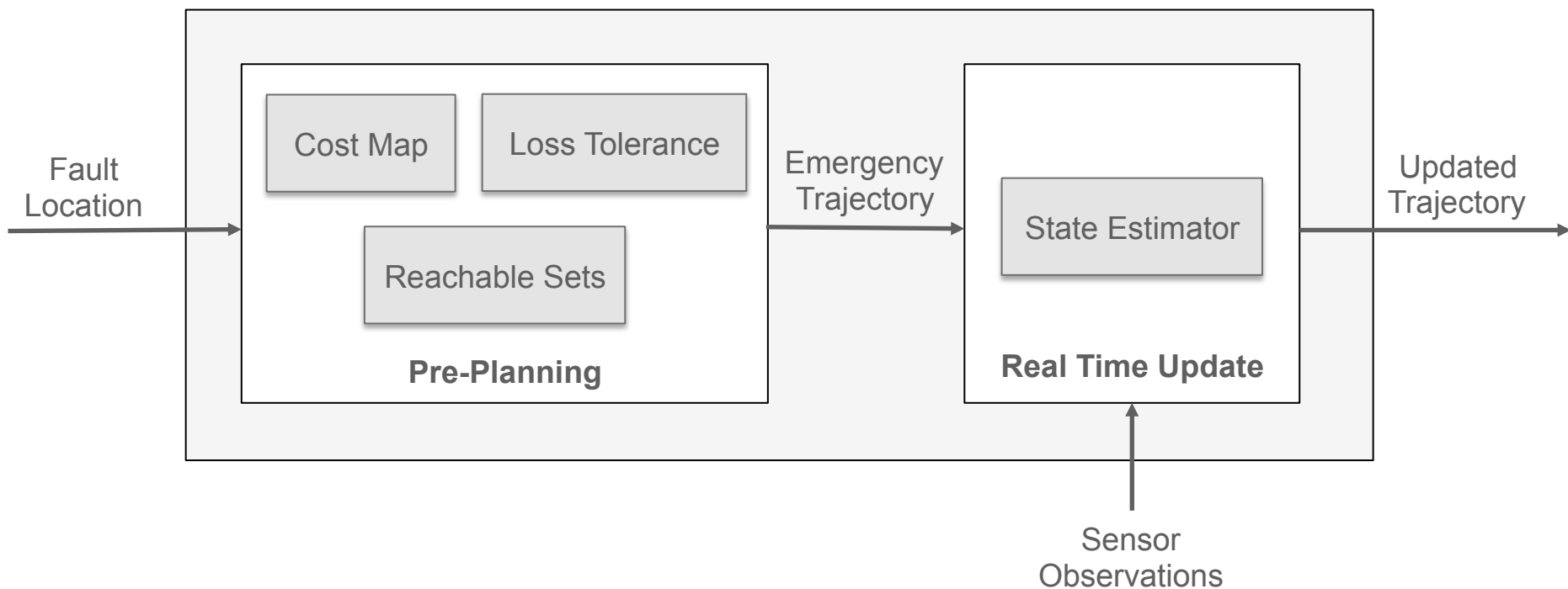
- **Where To Land (WTL)** is a emergency forced landing algorithm developed by UC Berkeley
- Inflight emergency → vehicle forced to land
  - What is the **optimal landing location** that will **minimize loss of life** and **minimize property damage** given a **set of constraints**
  - What is the optimal trajectory required for the aerial vehicle to reach optimal landing location?
- WTL attempts to mimic an expert pilot's decision making and land the aircraft



WTL Concept of Operations



# WTL Algorithm



**Pre-Planning** - pre-compute trajectories using fault location, maps and reachable sets

**Real Time Update** – adapt emergency trajectory based on real time data (weather, occupancy, etc.)



# Innovation

Prior Forced  
Landing Algorithms

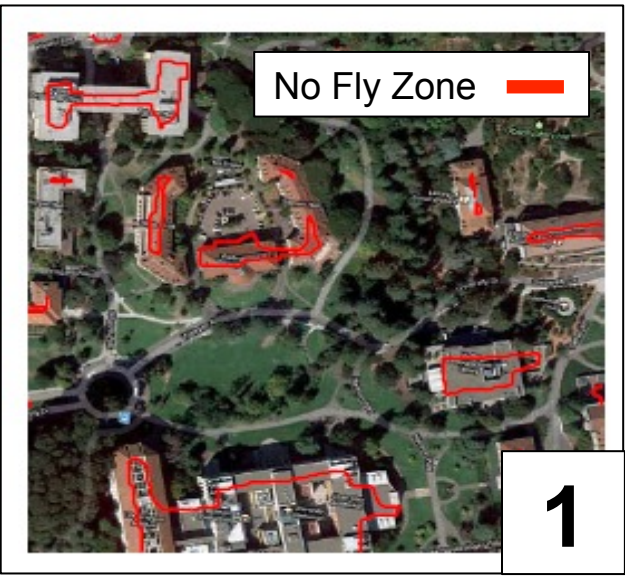
- Simple dynamics model
- Assumes aircraft can return to runway
- Difficult to apply to autonomous vehicles
- Haven't been flight tested

Where to Land  
Algorithm

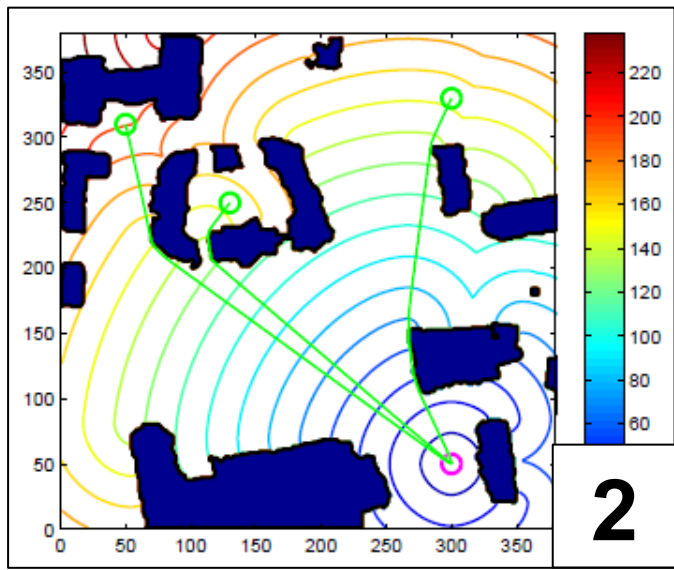
- Provides safety guarantees for S/W V&V
- Higher fidelity aircraft model
- Fast computation
- Manned or unmanned vehicles
- Modular design



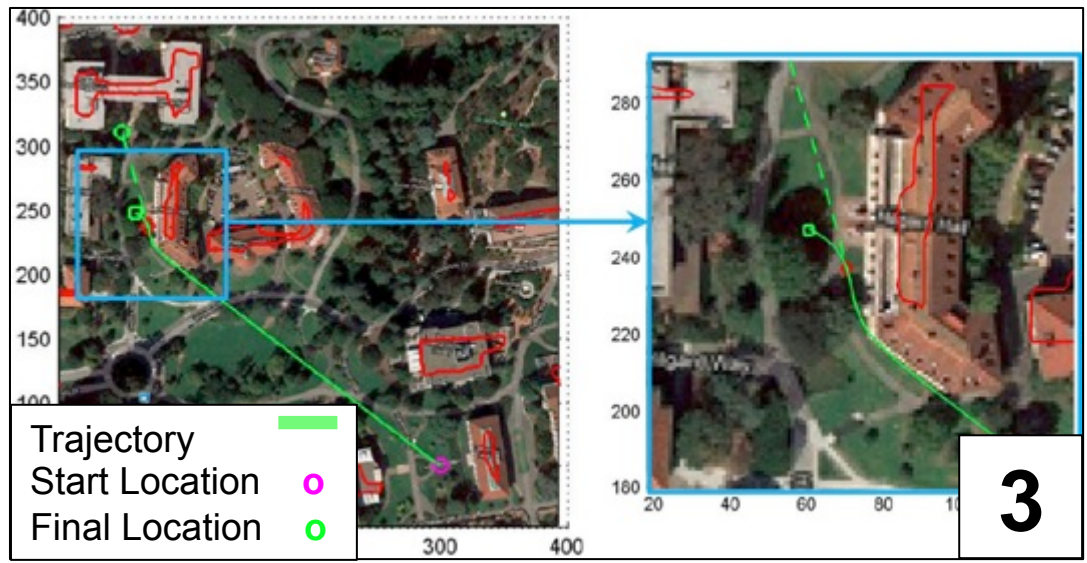
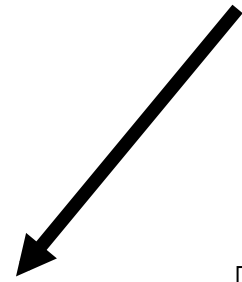
# WTL1 Phase 1 Results



UC Berkeley Campus



Emergency Trajectories



Emergency Landing Location

Demo: MATLAB sim  
 Location: UC Berkeley  
 Vehicle: Quadrotor  
 Failure: 90% thrust  
 2D Trajectory





# Phase 1 → Phase 2

- **Reduce the scope of WTL**
  - Simplify WTL → Speed up software development
  - Find “real world” design/implementation issues
  - Get pilot feedback with HIL simulation
  - Collect data to improve future versions
- **WTL1 → WTL2**
  - NASA TCM/B-757 aerodynamics model
  - No real time update → compute trajectories during fault
  - No global cost map → NYC/New Jersey area ~100+ miles
  - No Fault detection → One predefined fault, dual engine failure
  - HIL 6DOF nonlinear aircraft simulation

<b>PHASE 2 GOALS</b>	<ul style="list-style-type: none"><li>• Demonstrate WTL in HIL simulation</li><li>• Develop tools to generate reachable trajectories</li></ul>
--------------------------	--



# WTL Development Plan

## Phase 1 – WTL1

Demo: MATLAB Sim  
Location: UC Berkeley  
Vehicle: Quadrotor  
Failure: 90% reduction in thrust  
2D Trajectory

## Phase 2 – WTL2

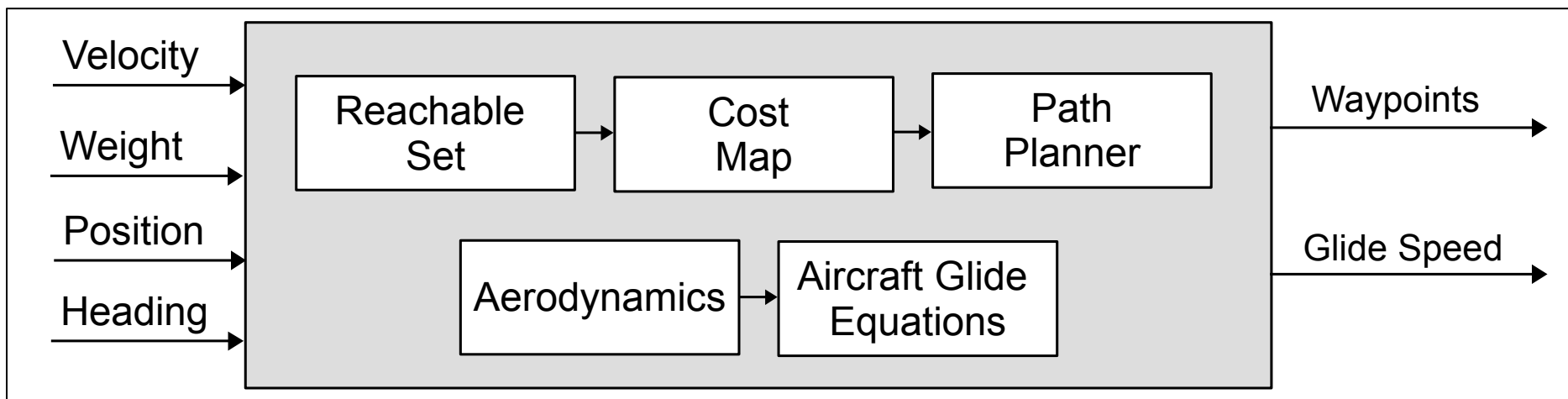
Demo: HIL Sim w/ FLS on embedded H/W  
Location: New York City +/- ~100 miles  
Vehicle: 757  
Failure: Loss of thrust  
2D Trajectory

## Future Work

Demo: Flight test RC Aircraft w/ Pixhawk  
Location: Edwards, CA  
Vehicle: RC Aircraft  
Failure: Loss of thrust  
2D Trajectory



# WTL2 Architecture



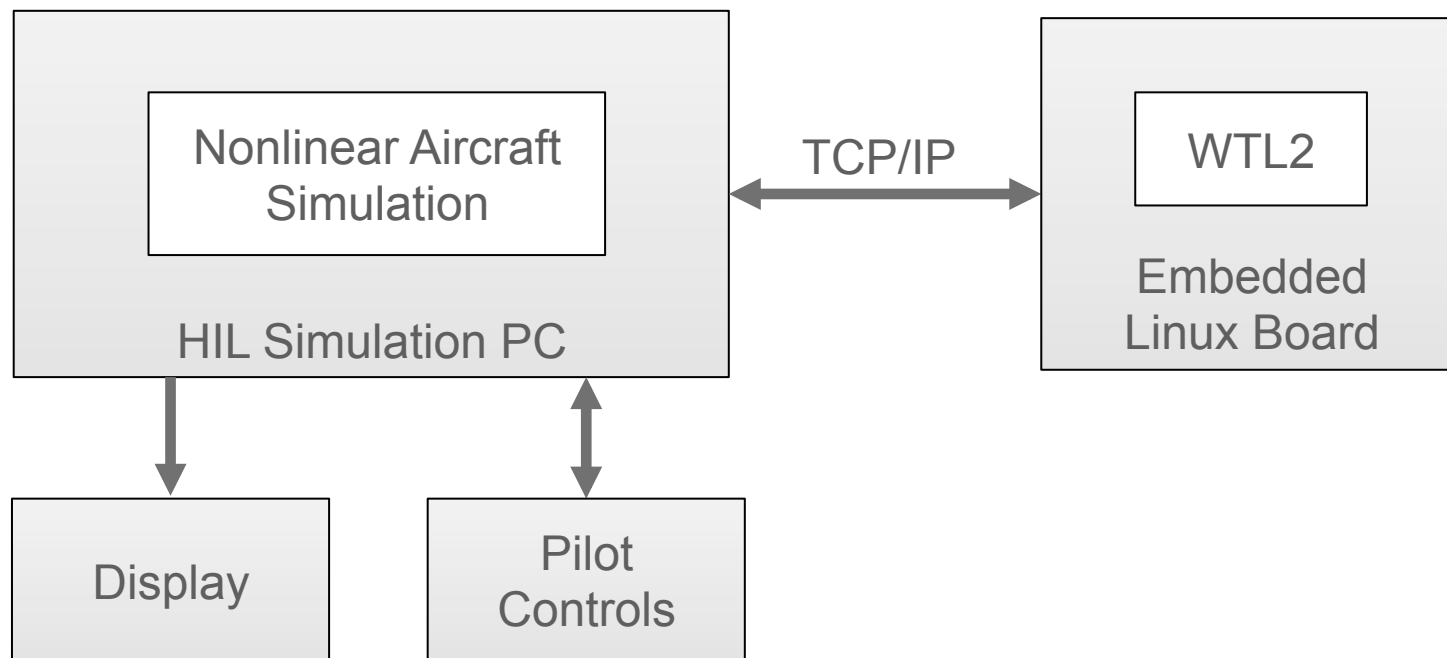


# WTL2 Algorithm

1. Get current aircraft state
  - Latitude/Longitude
  - Altitude/Heading/Velocity
2. Convert states to local frame
3. Compute maximum glide range
4. Window cost map with max range
5. Get reachable set for altitude
6. Scale and project reachable set over map with heading
7. Find best reachable landing location using 2D convolution
8. Generate trajectory using optimal path planner
9. Generate latitude/longitude waypoints
10. Generate target headings



# HIL Simulation Architecture





# Engine Out Scenario

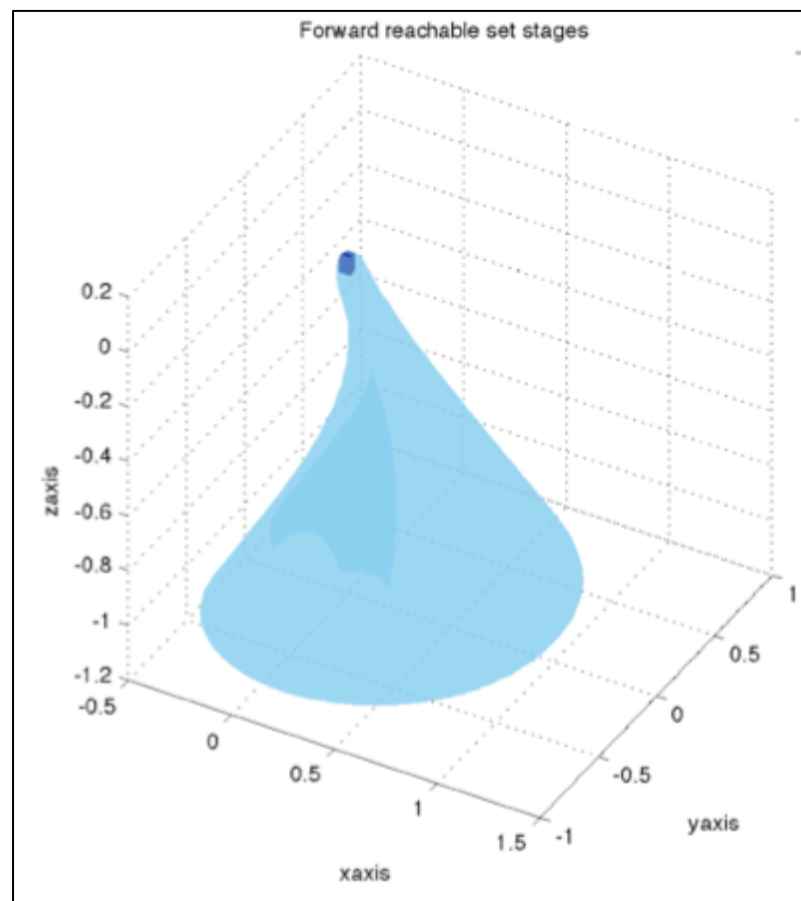
- Complete loss of thrust
- Engine out during takeoff is the most critical
  - WTL2 Operational Range: 1000 ft – 4000 ft
  - Less than 1000 ft → Can only land straight ahead
  - Greater than 4000 ft → Can often return to airport
  - Glide range will vary based on aircraft and configuration (i.e. weight, flaps)
- During failure → pilots must manage energy
- Flying at  $L/D_{MAX}$  maximizes aircraft range
- $L/D_{MAX} \rightarrow \alpha_{MAX} \rightarrow$  gross weight  $\rightarrow V_{GLIDE}$
- Flying at  $V_{GLIDE}$  will maximize aircraft range



# Reachability

Reachability - Given a dynamic system governed by some differential equation and input defined over some bounded state space. What are all the states visited by the trajectories of the system

- Reachability is a key technology for verifying safety critical systems<sup>7</sup>
- Reachability assures that a system can reach a target state while remaining within a safety envelope<sup>7</sup>
- Level Set Toolbox - computes reachable sets of hybrid systems with continuous dynamics using nonlinear ODE's<sup>3</sup>
- Grid based computation





# Aircraft Reachability

Aircraft Reachability is gliding aircraft model with NASA TCM aerodynamics formulated as a PDE (HJ) and solved using the Level Set Toolbox. Aircraft trajectory has two modes. The two mode states are stitched together using a hybrid system model.

## Mode 1 - Approach Mode

- TCM aerodynamics
- Glide equations
- Glide velocity
- Constant radius turns
- State constraints

## Mode 2 – Landing Mode

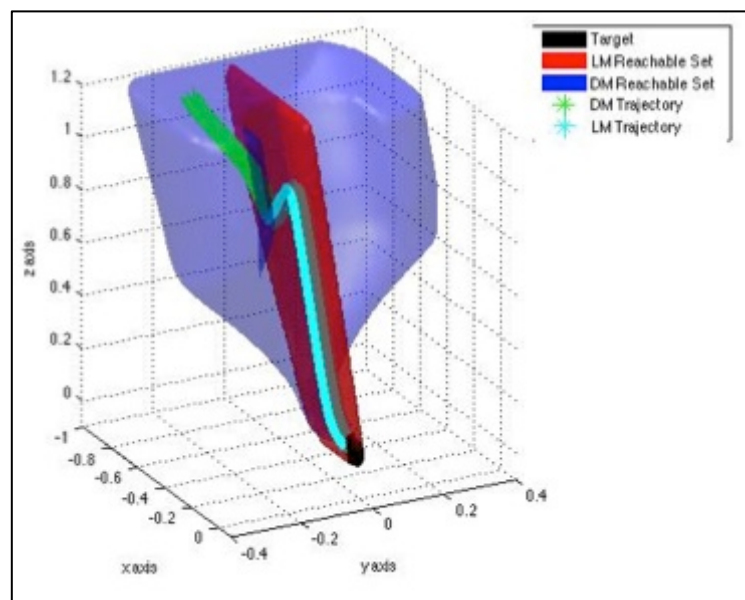
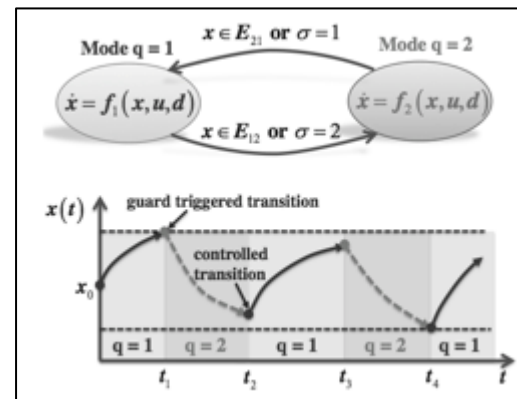
- TCM 30° flap aerodynamics
- Landing velocity
- State constraints

## States

- Aircraft position
- Velocity
- Flight path and heading angles

## Control

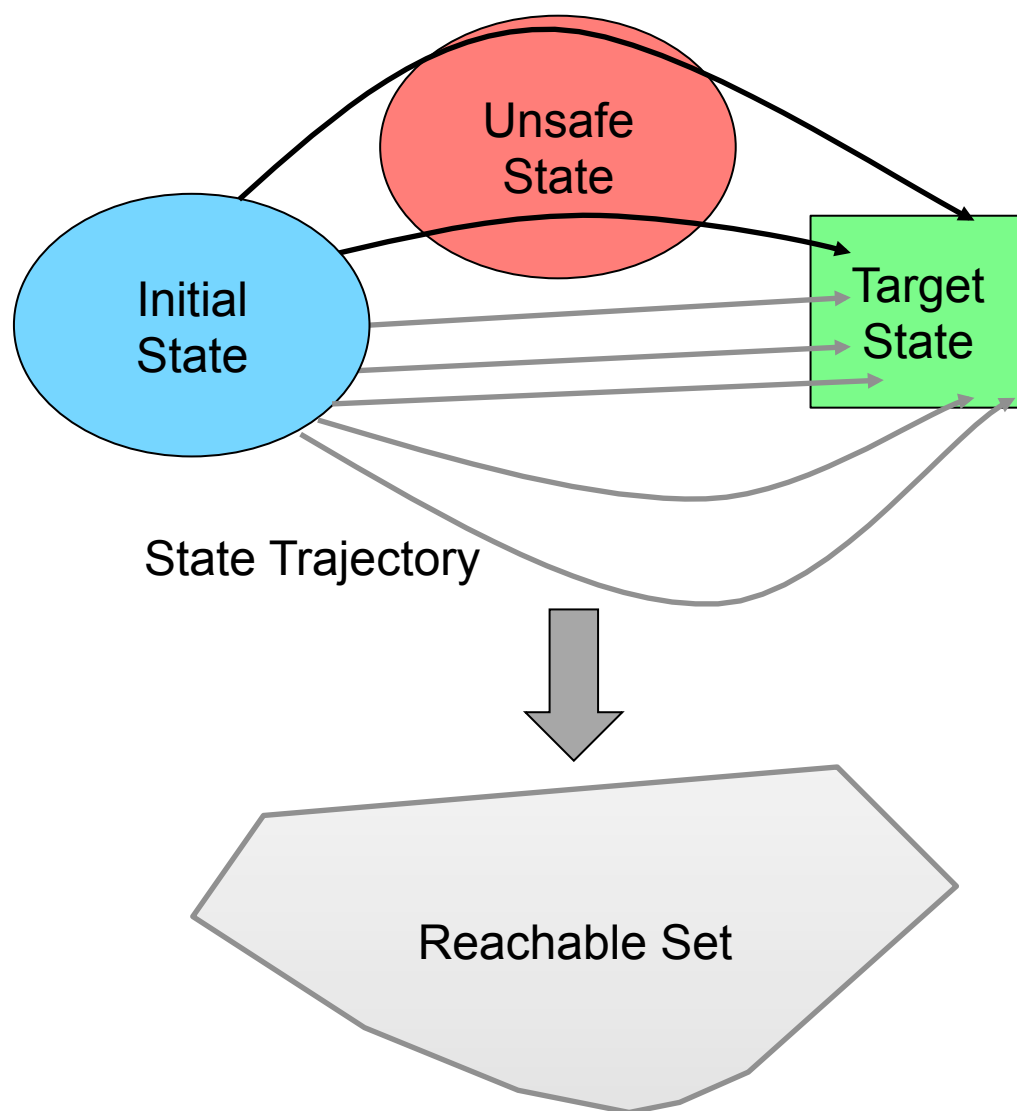
- Angle of attack
- Bank angle







# Reachable Set



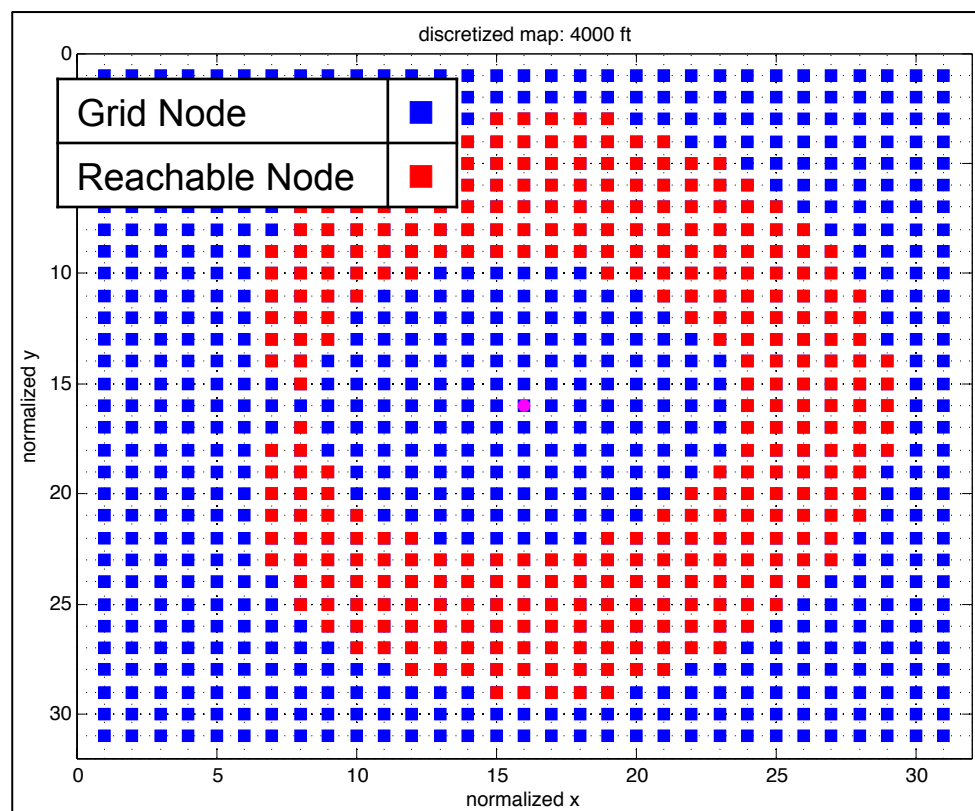
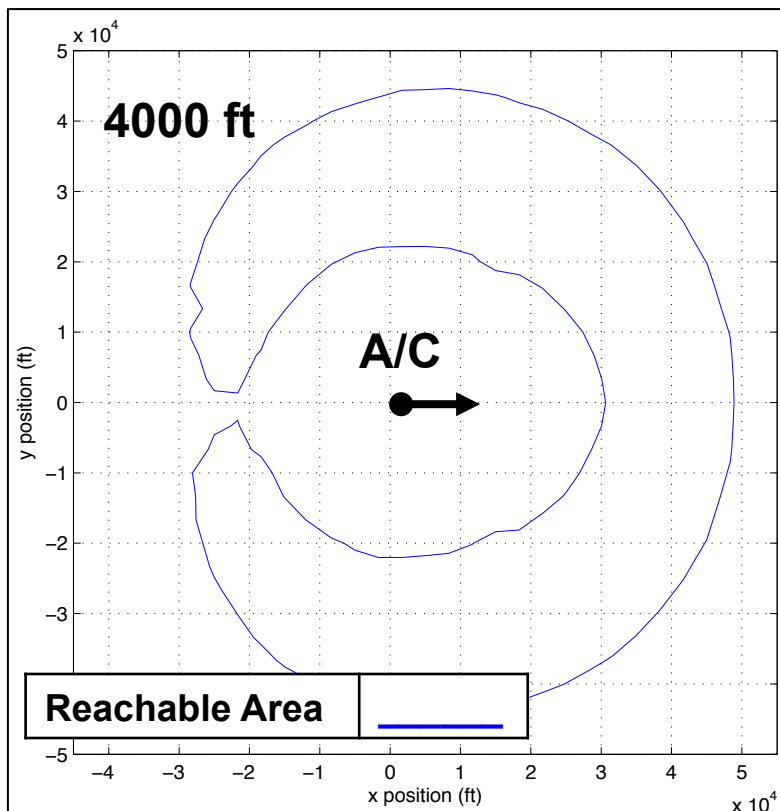
Reachable sets are a set of initial states from which the system is guaranteed to remain inside a safe region while eventually reaching a desired target<sup>3</sup>

## State Constraints

$V$  – Stall avoidance  
 $\alpha, \phi$  – Keeps aircraft within performance envelope  
Acceleration - structural load limits



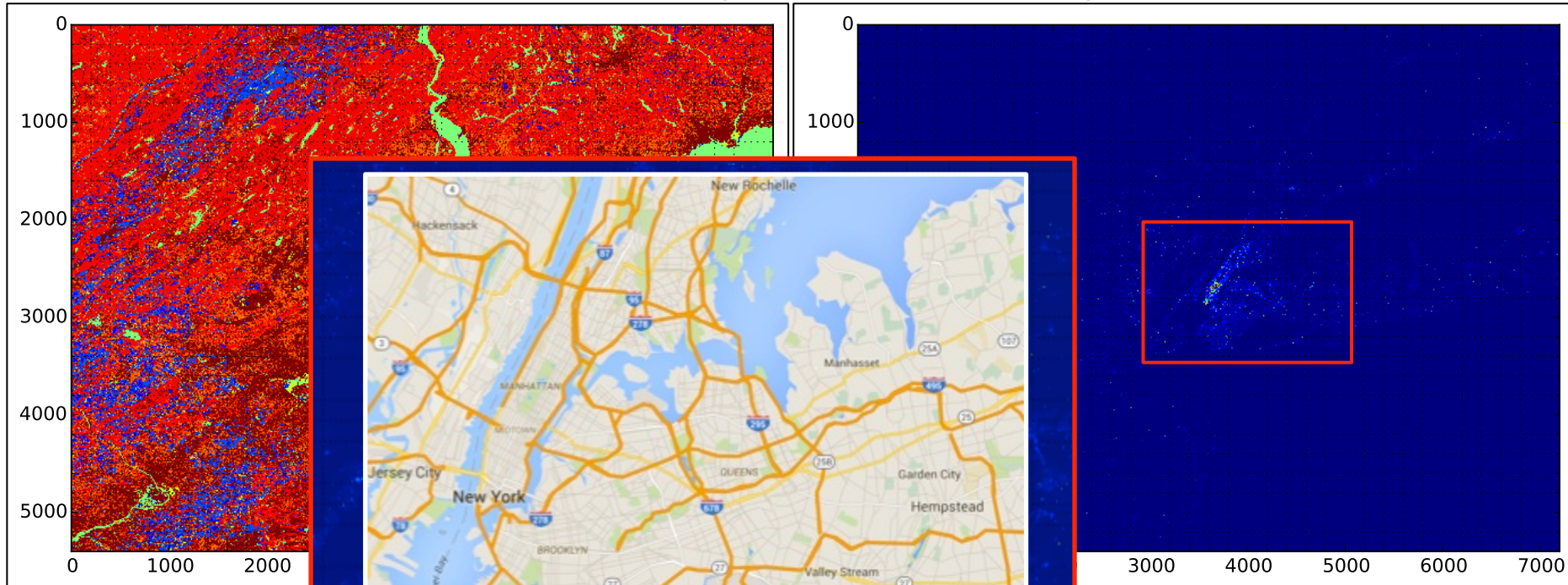
# Discrete Reachable Sets



- Reachable sets generated every 100 ft from 1000 ft - 4000 ft
- Grid size  $10E4 \times 10E4$  ft
- Normalized and stored as a binary map
- Oriented onto global map using aircraft heading

# Cost Map

- Hazard Map – constructed from population and geographical data
- Impact Map – constructed from density maps, land use maps, etc.
- Total Loss Map = Hazard Map + Impact Map
- Map Size: 7201x5401 pixels (3.5+ million pixels)



NYC

NYC Impact Map



# Gliding Aircraft Equations

- 3D motion of gliding aircraft over flat Earth
- Model assumes coordinated turns, no sideslip

$$\dot{X} = V \cos \gamma \cos \xi$$

$$\dot{Y} = V \cos \gamma \sin \xi$$

$$\dot{Z} = V \sin \gamma$$

$$\dot{V} = -\frac{D(\alpha, V)}{m} - g \sin \gamma$$

$$\dot{\gamma} = \frac{L(\alpha, V) \cos \phi}{mV} - \frac{g}{V} \cos \gamma$$

$$\dot{\xi} = \frac{L(\alpha, V) \sin \phi}{mV \cos \gamma}$$

Aircraft velocities

Aircraft acceleration

Flight path derivative

Heading derivative

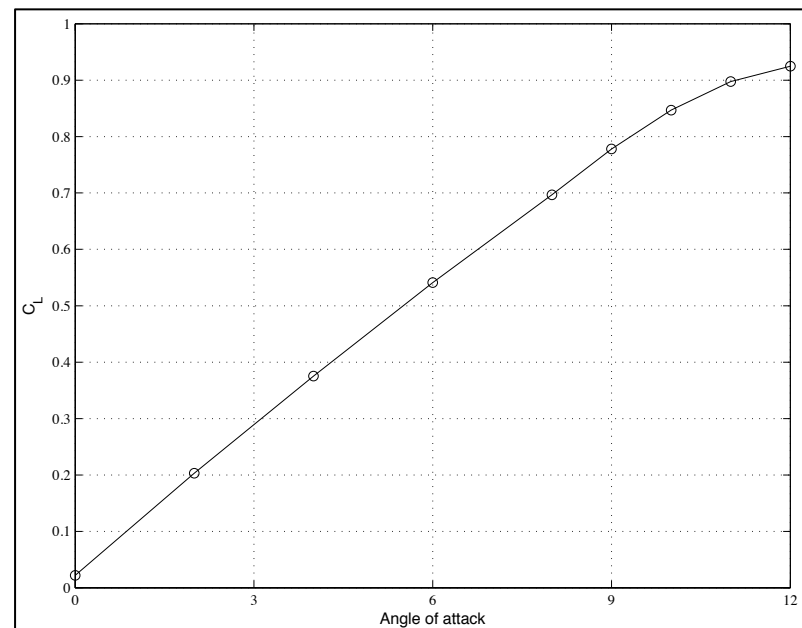
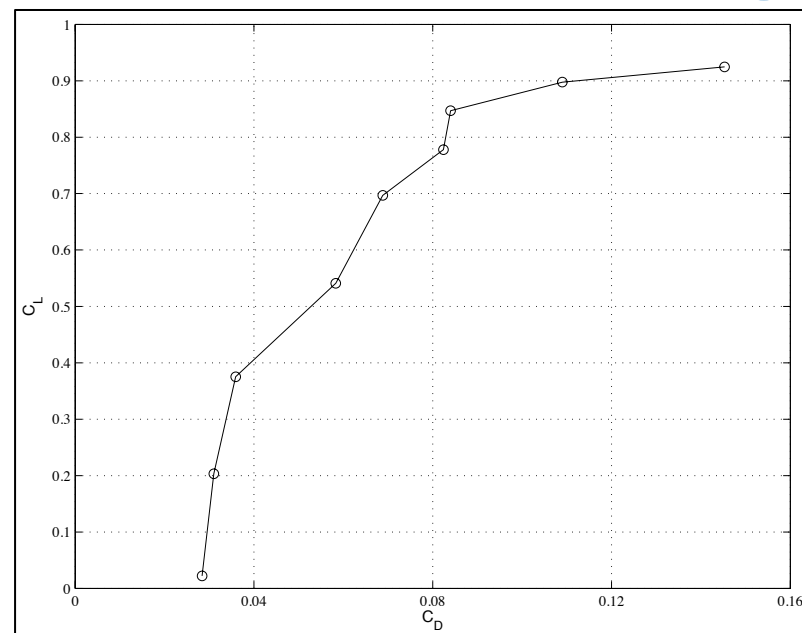
$$V_{glide} = \sqrt{\frac{2W}{\rho S \sqrt{C_D^2 + C_L^2}}}$$

Optimum glide velocity



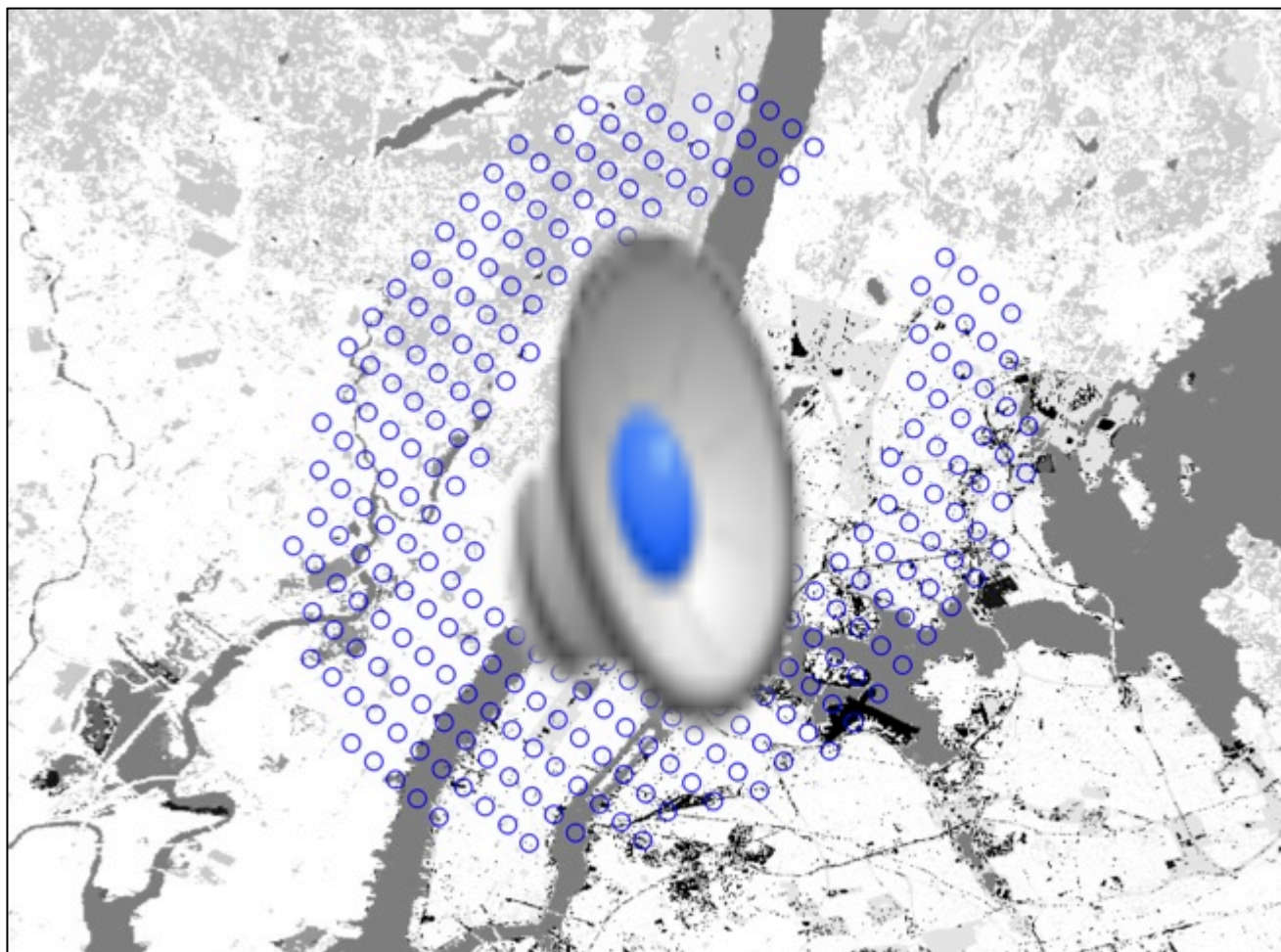
# NASA TCM Model

- Nonlinear aircraft model developed by NASA Langley for NASA's Aviation Safety Program
- Transport Class Model (TCM) closely replicates B-757 aerodynamics
- For WTL2, TCM aerodynamics tables ( $C_L, C_D$ ) are used
- On landing transition to 30° Flap aerodynamics
- Compute  $L/D_{MAX}$  and  $\alpha_{MAX}$





# Optimal Landing Location



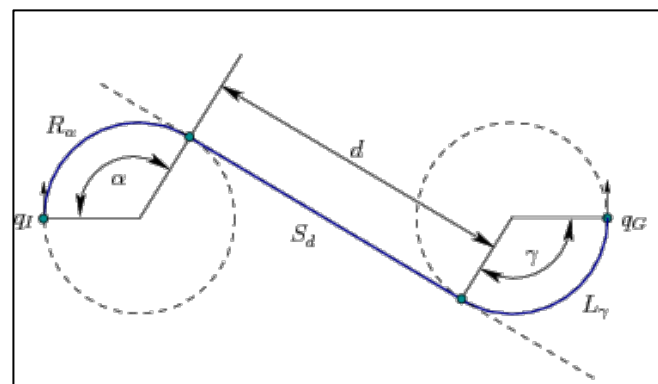
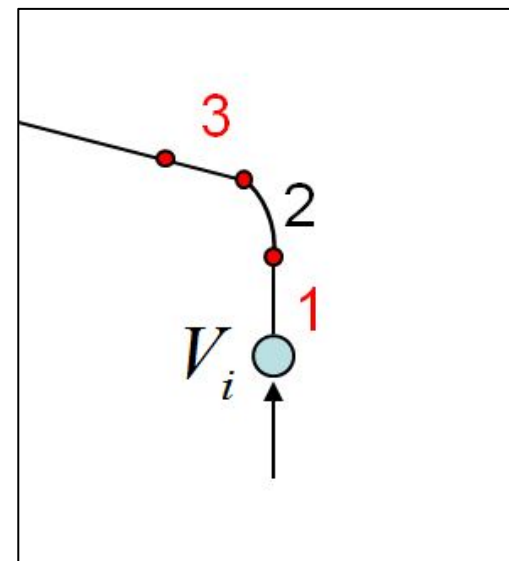
○	Reachable Node
●	Searched Node
■	Landing Footprint

- Landing footprint is based on aircraft ground roll and impact area
- Optimal landing location = smallest total sum cost over landing footprint
- Found using 2D Convolution with FFT



# Optimal Trajectory Generation

- Dubins trajectory – gives shortest path between two points
  - requires final location and final heading
  - target heading here is the heading required to reach final landing location
- Two basic maneuvers
  - Gliding (maximize glide range)
  - Turning (final orientation)
- Optimal turn radius – minimize energy loss with a constant radius turn





# WTL2 C Code

- Dependencies
  - GSL (Numerical Library)
  - GDAL (GIS Library)
- Makefile
  - generates executable for ARM, x86 processors
  - ccompcert → safety critical C compiler
- V&V
  - Use JPL Flight S/W Best Practices (JPL DOCID D-60411)
  - Run code coverage tool
  - Memory debugging tool
  - Unit tests for critical functions
  - Test Cases





# Test Cases

Test #	Altitude (ft)	Latitude	Longitude	Initial Heading
1	1000	40.70°	-73.8726°	270°
2	1000	40.70°	-73.8726°	15°
3	1000	40.85°	-73.70°	270°
4	4000	40.70°	-73.8726°	270°
5	4000	40.70°	-73.8726°	15°
6	4000	40.85°	-73.70°	270°
7	4000	40.85°	-73.70°	15°
8	3026	40.865	-73.88°	220

- Altitude variation – Bounded by two altitudes
  - Altitude < 1000 ft → Can only land straight ahead
  - Altitude > 4000 ft → Should be able to return to airport
- Heading variation – Show effects of initial heading on trajectory
- Position variation – Show effects of initial position on trajectory
- Case #8 replicates US Airways 1549 failure

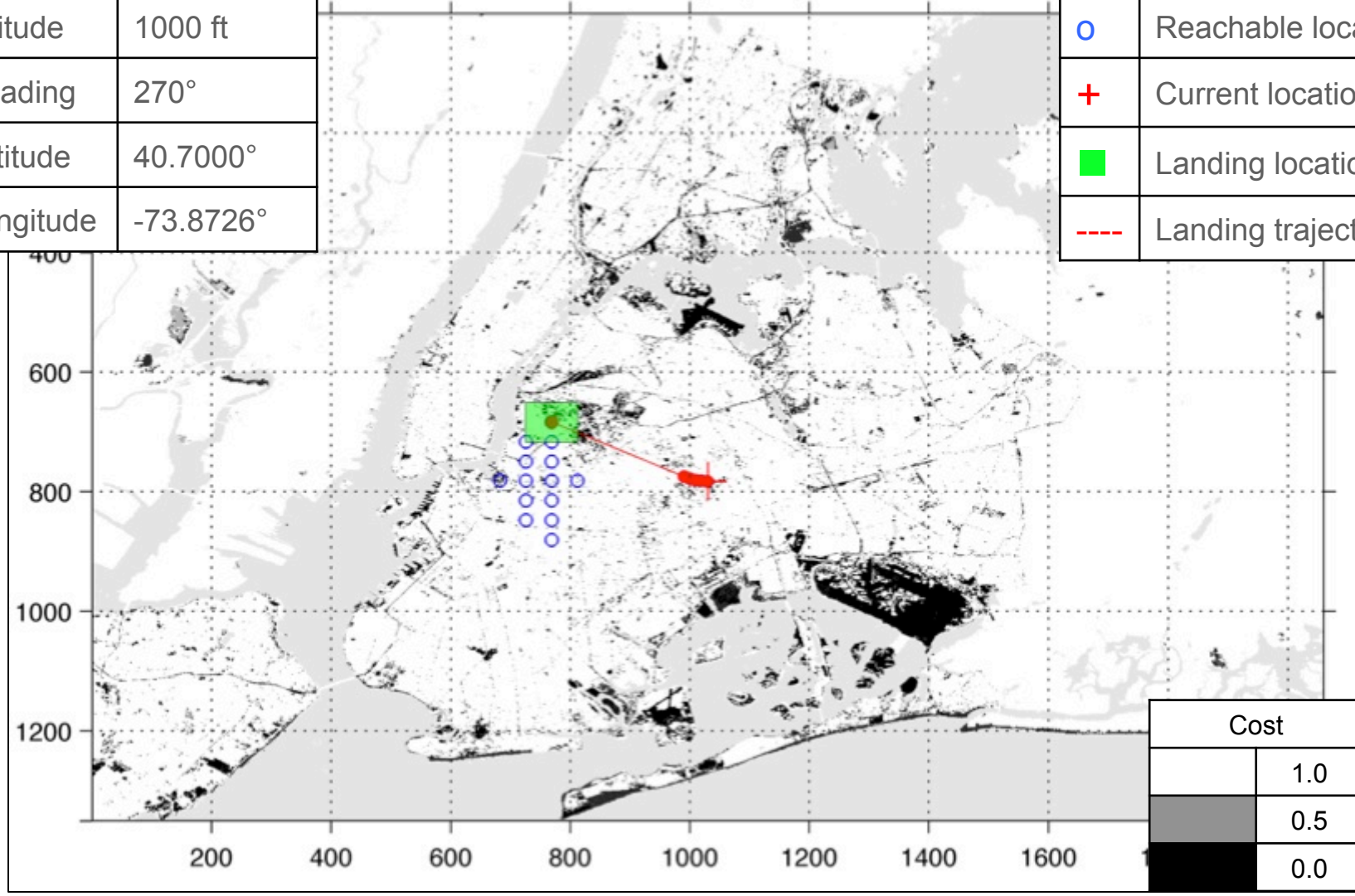


# Results: Test Case 1

land at [684,769]

Altitude	1000 ft
Heading	270°
Latitude	40.7000°
Longitude	-73.8726°

○	Reachable location
+	Current location
■	Landing location
---	Landing trajectory



Cost	
	1.0
■	0.5
■	0.0

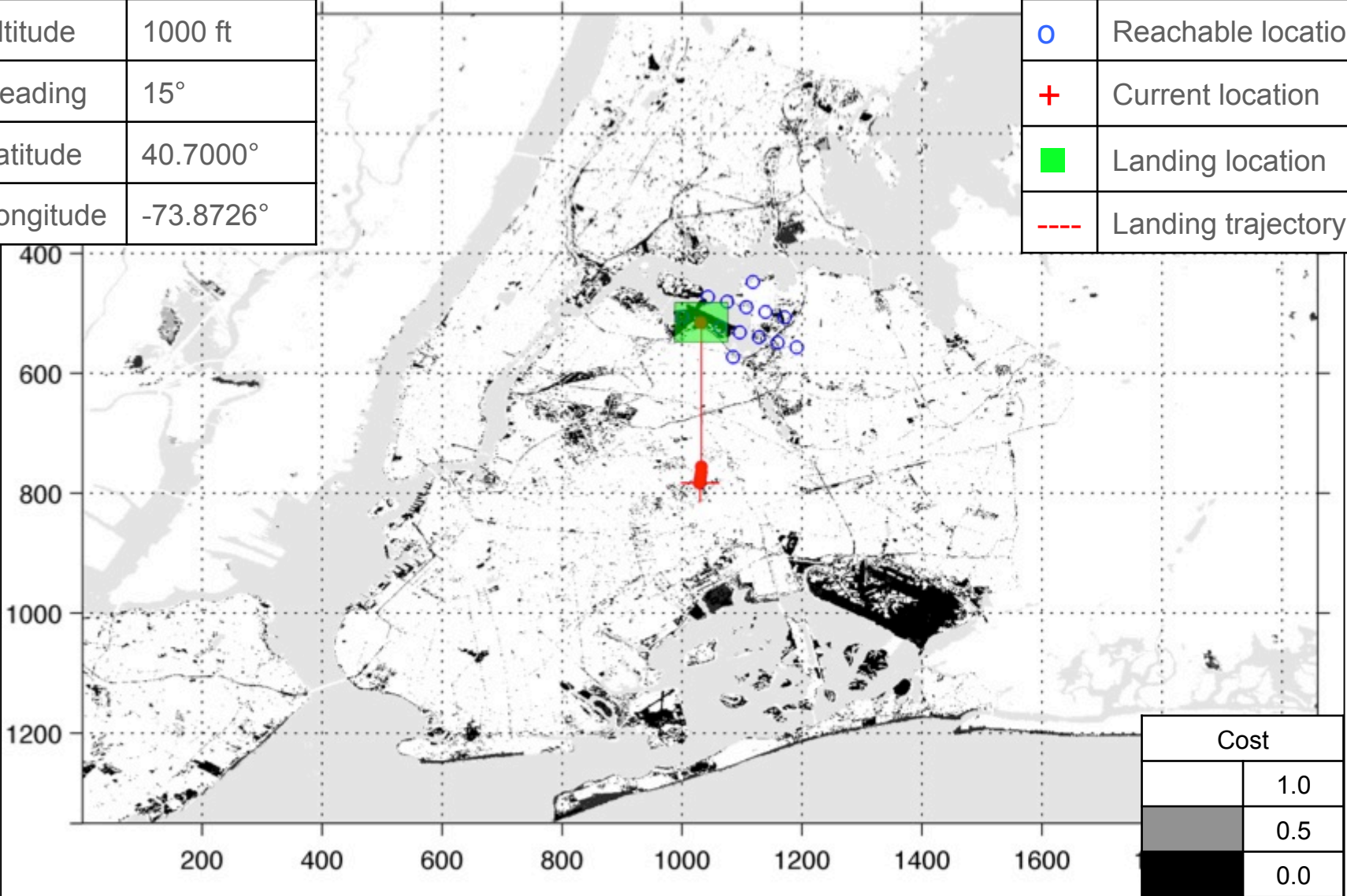


# Results: Test Case 2

land at [515,1032]

Altitude	1000 ft
Heading	15°
Latitude	40.7000°
Longitude	-73.8726°

○	Reachable location
+	Current location
■	Landing location
---	Landing trajectory



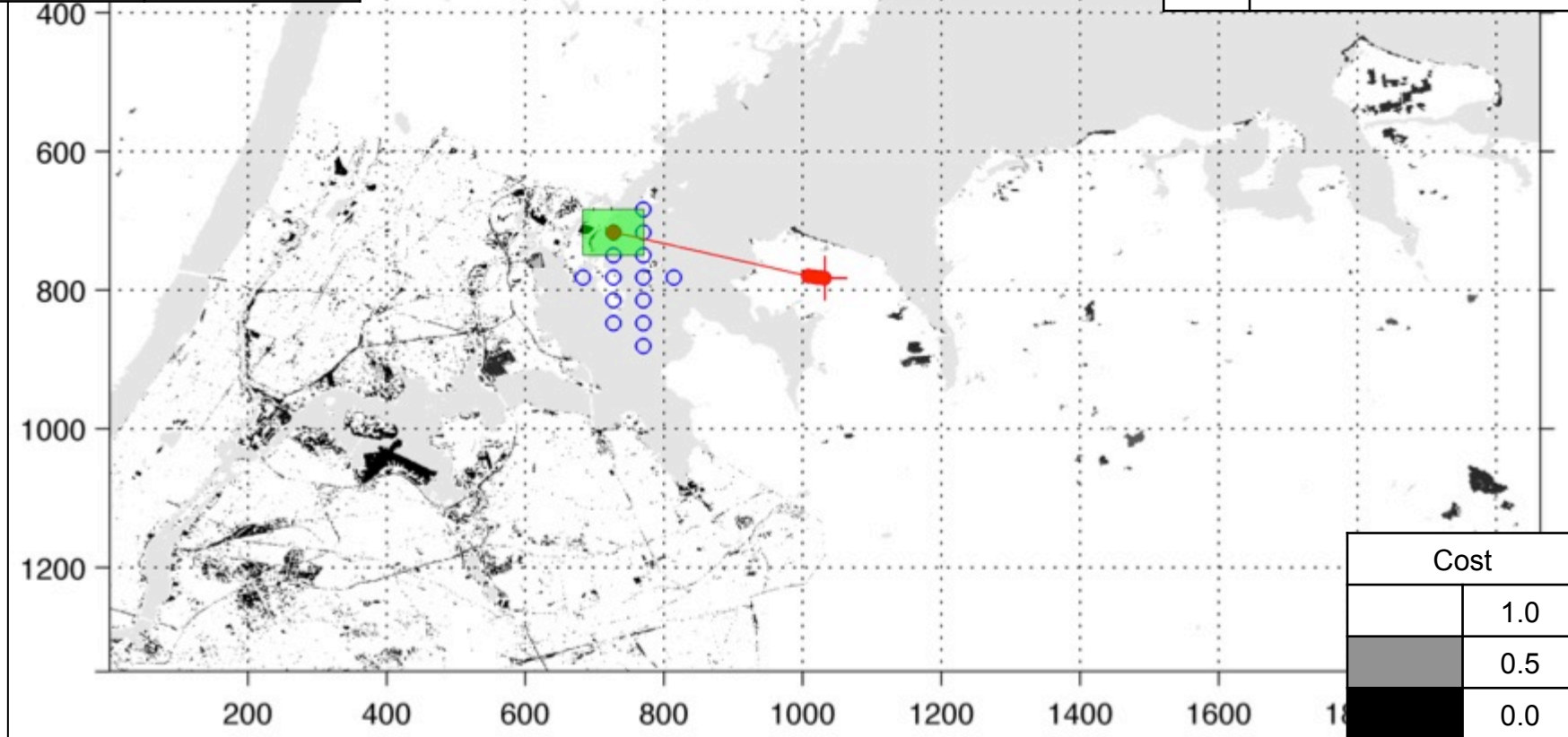


# Results: Test Case 3

land at [717,727]

Altitude	1000 ft
Heading	270°
Latitude	40.85°
Longitude	-73.70°

○	Reachable location
+	Current location
■	Landing location
---	Landing trajectory



Cost	
	1.0
	0.5
	0.0

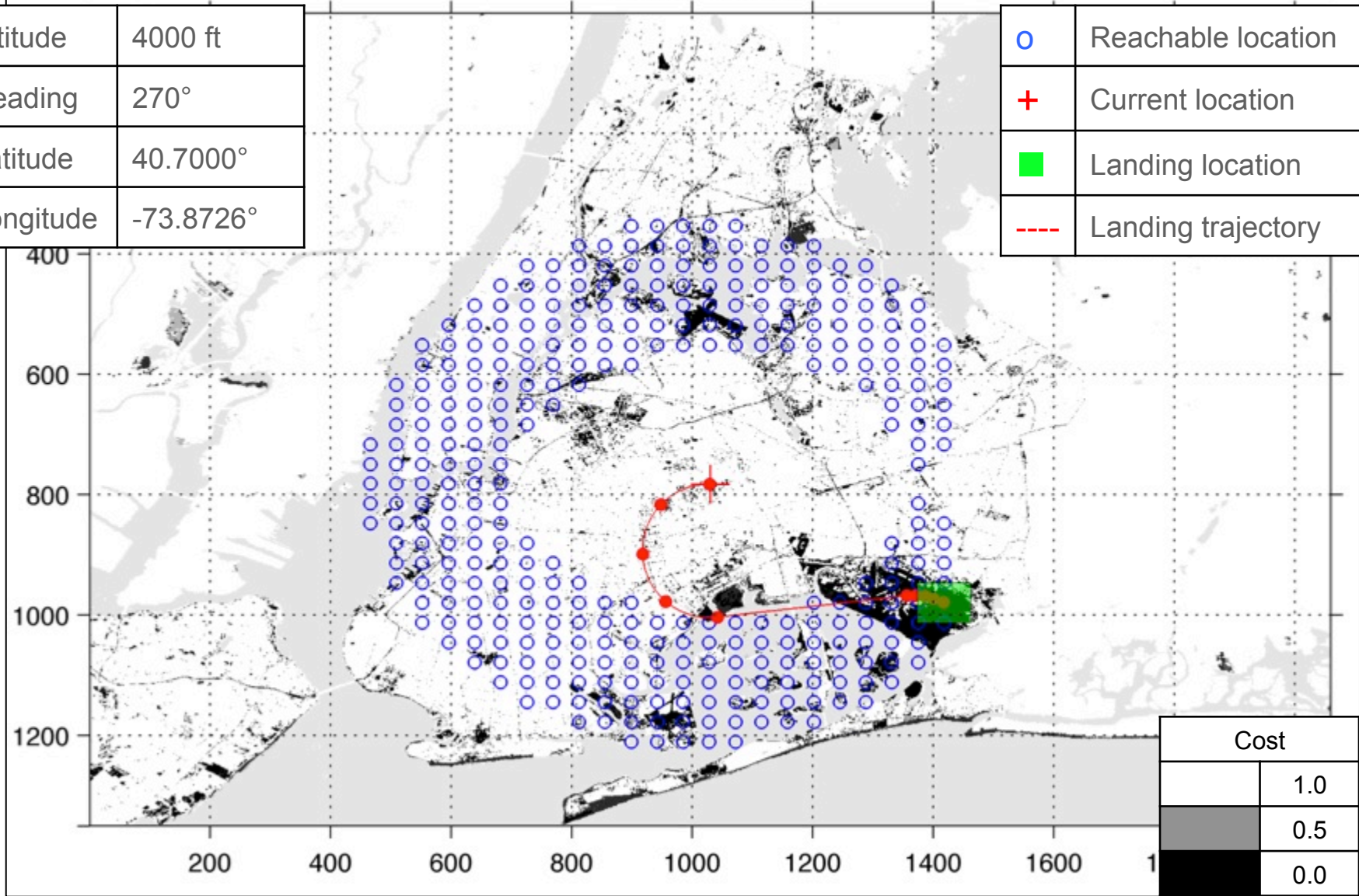


# Results: Test Case 4

land at [980,1418]

Altitude	4000 ft
Heading	270°
Latitude	40.7000°
Longitude	-73.8726°

○	Reachable location
+	Current location
■	Landing location
---	Landing trajectory



Cost	
	1.0
	0.5
	0.0

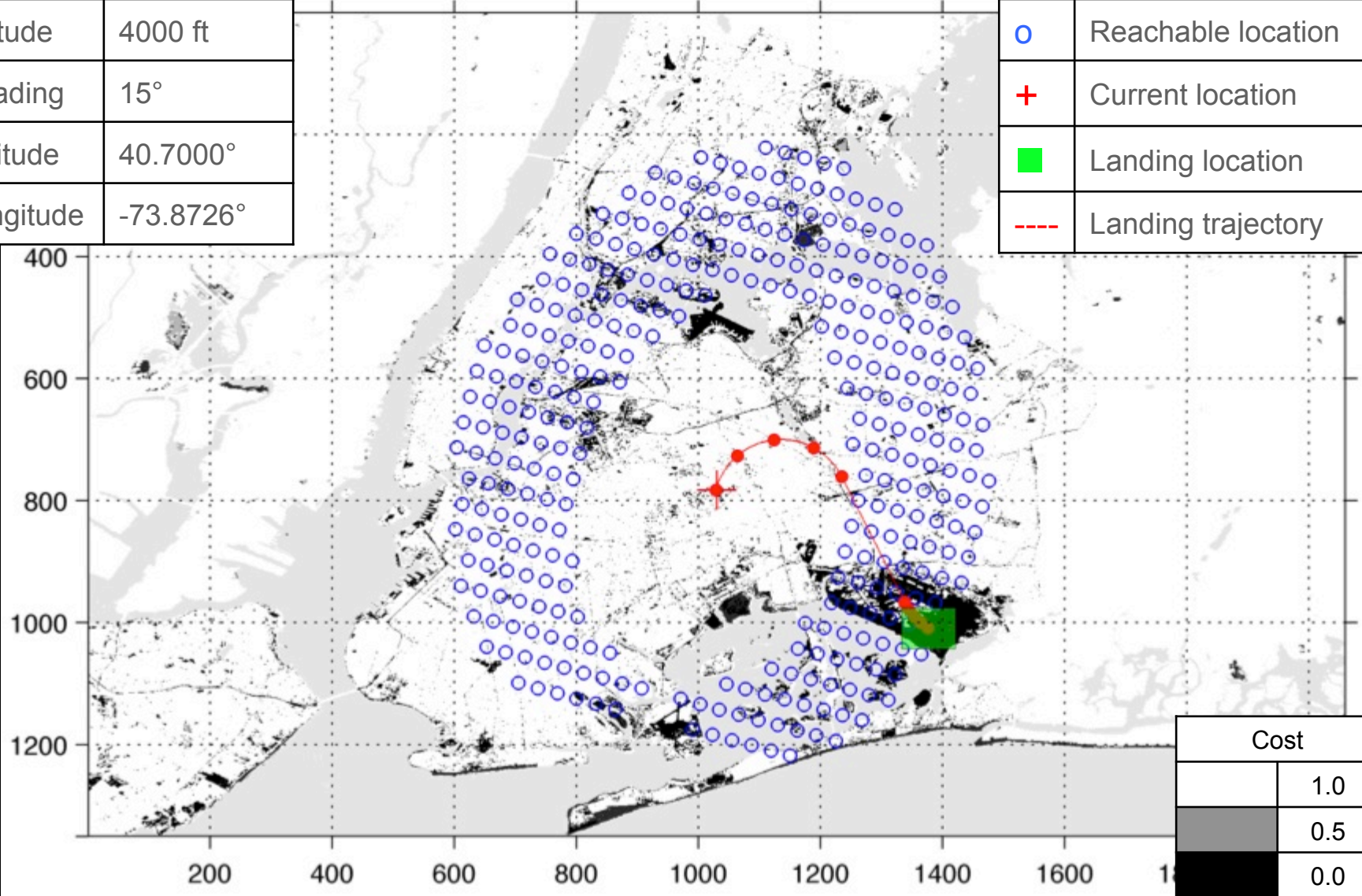


# Results: Test Case 5

land at [1010,1377]

Altitude	4000 ft
Heading	15°
Latitude	40.7000°
Longitude	-73.8726°

○	Reachable location
+	Current location
■	Landing location
---	Landing trajectory



Cost	
	1.0
	0.5
	0.0

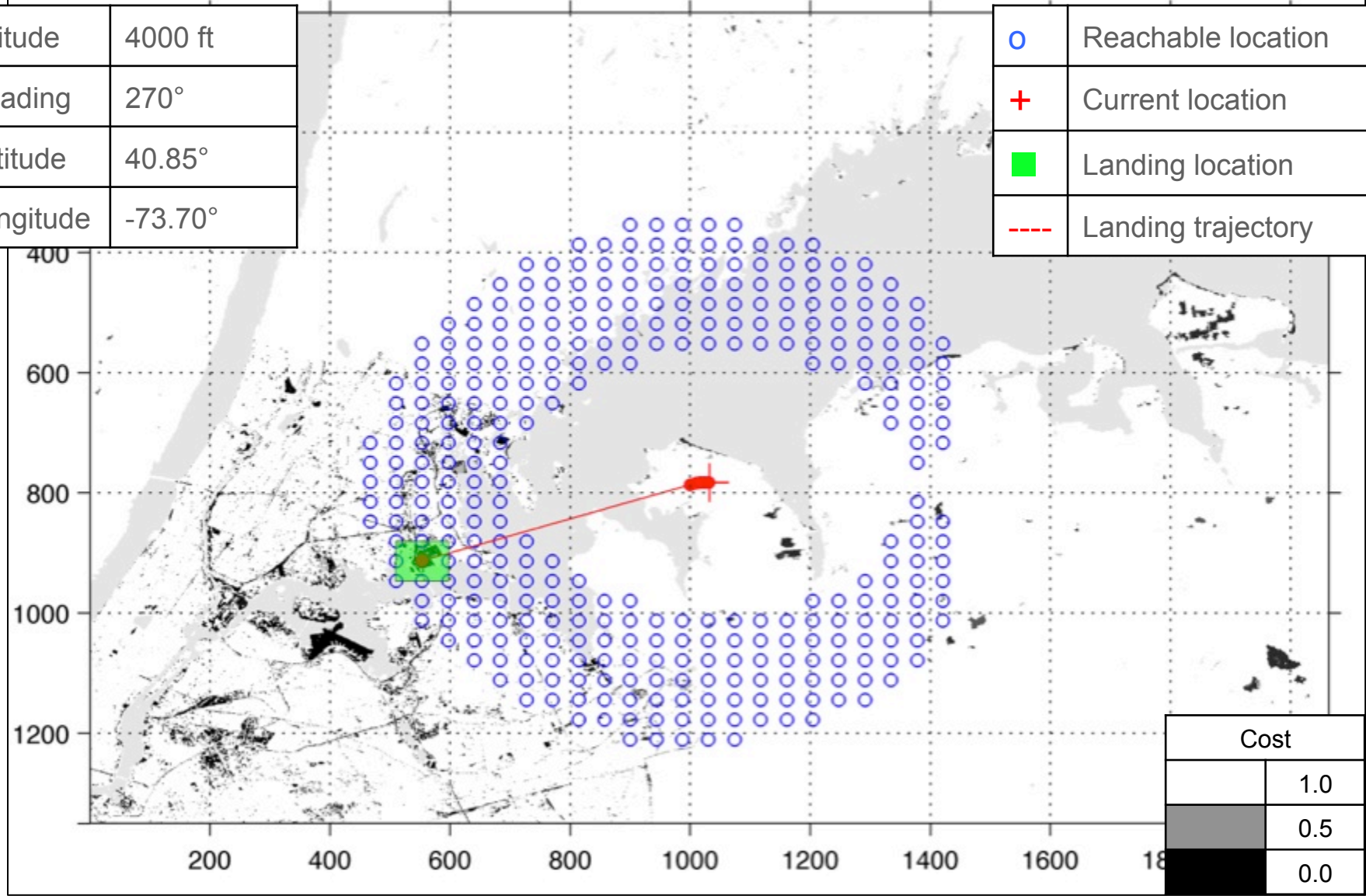


# Results: Test Case 6

land at [914,553]

Altitude	4000 ft
Heading	270°
Latitude	40.85°
Longitude	-73.70°

○	Reachable location
+	Current location
■	Landing location
---	Landing trajectory



Cost	
	1.0
	0.5
	0.0

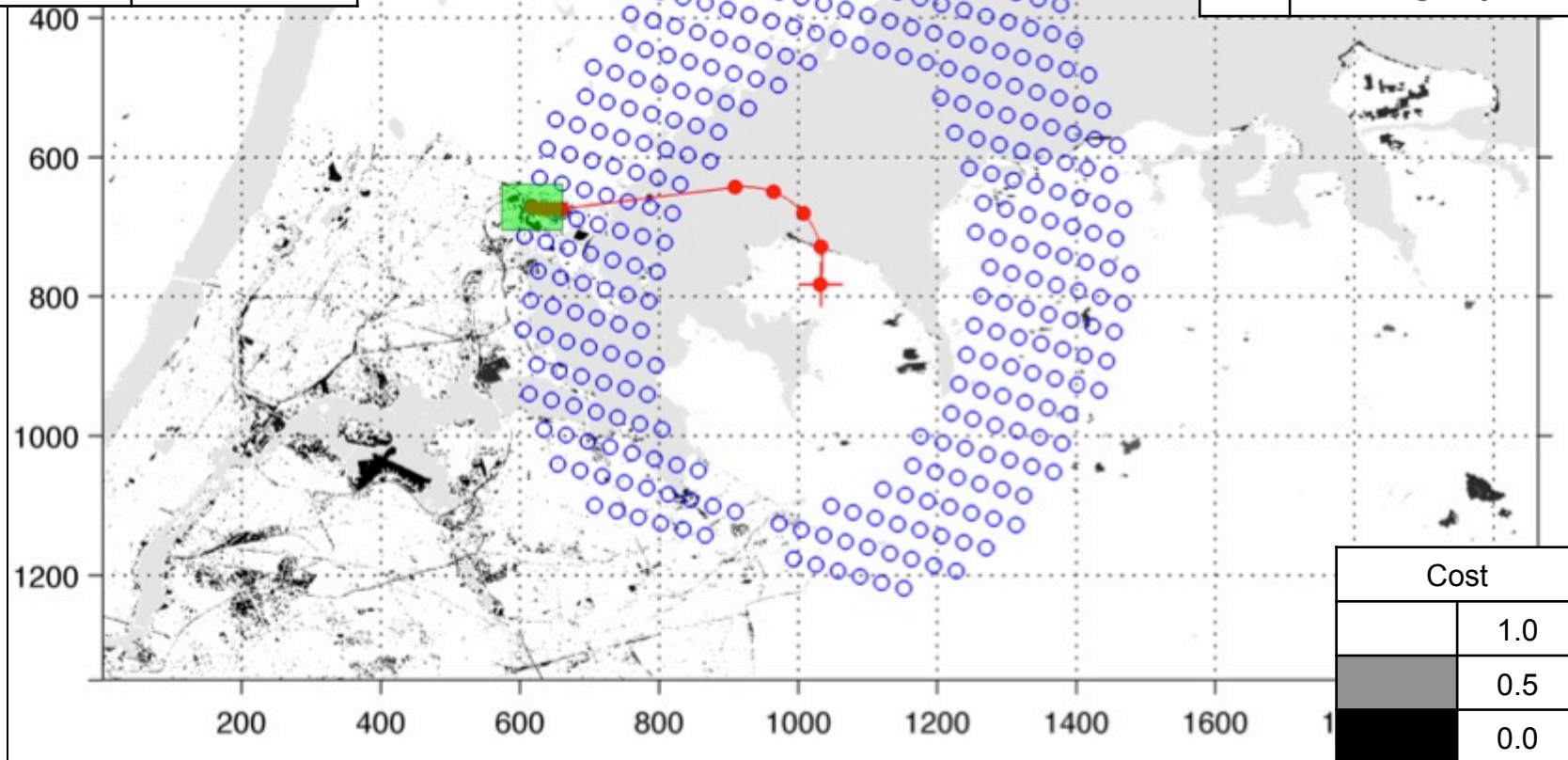


# Results: Test Case 7

land at [672,617]

Altitude	4000 ft
Heading	15°
Latitude	40.85°
Longitude	-73.70°

○	Reachable location
+	Current location
■	Landing location
---	Landing trajectory







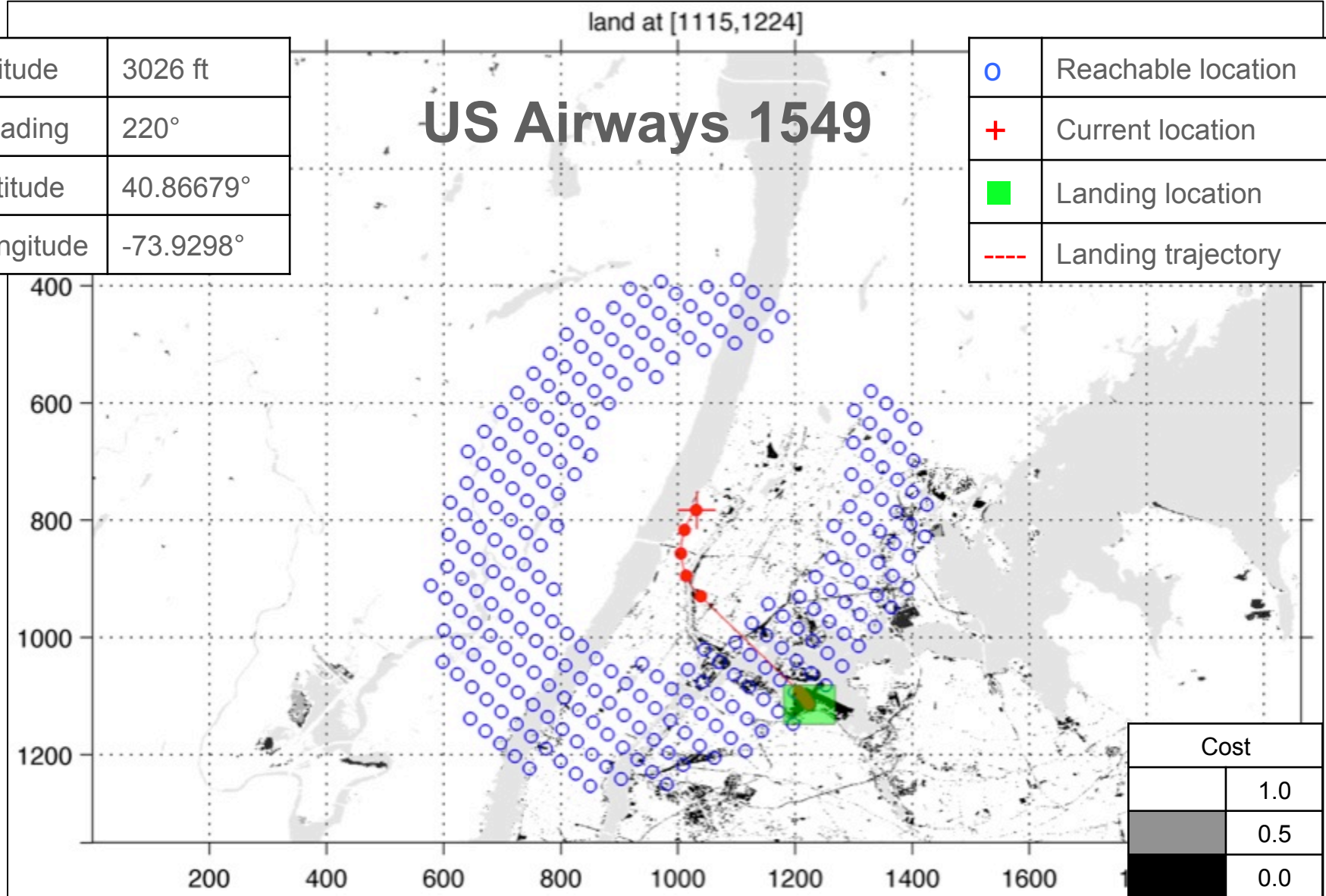
# Results: Test Case 8

land at [1115,1224]

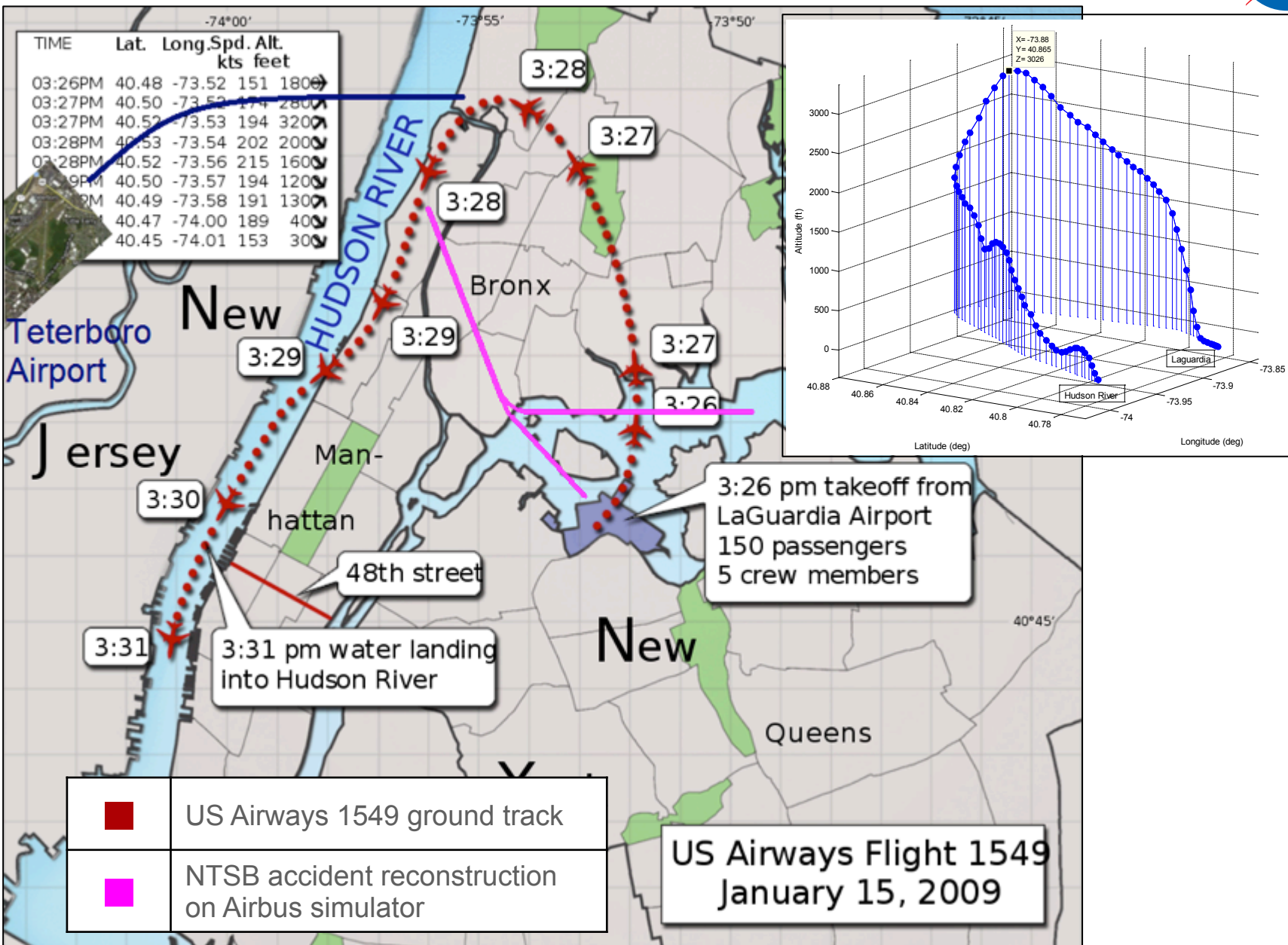
Altitude	3026 ft
Heading	220°
Latitude	40.86679°
Longitude	-73.9298°

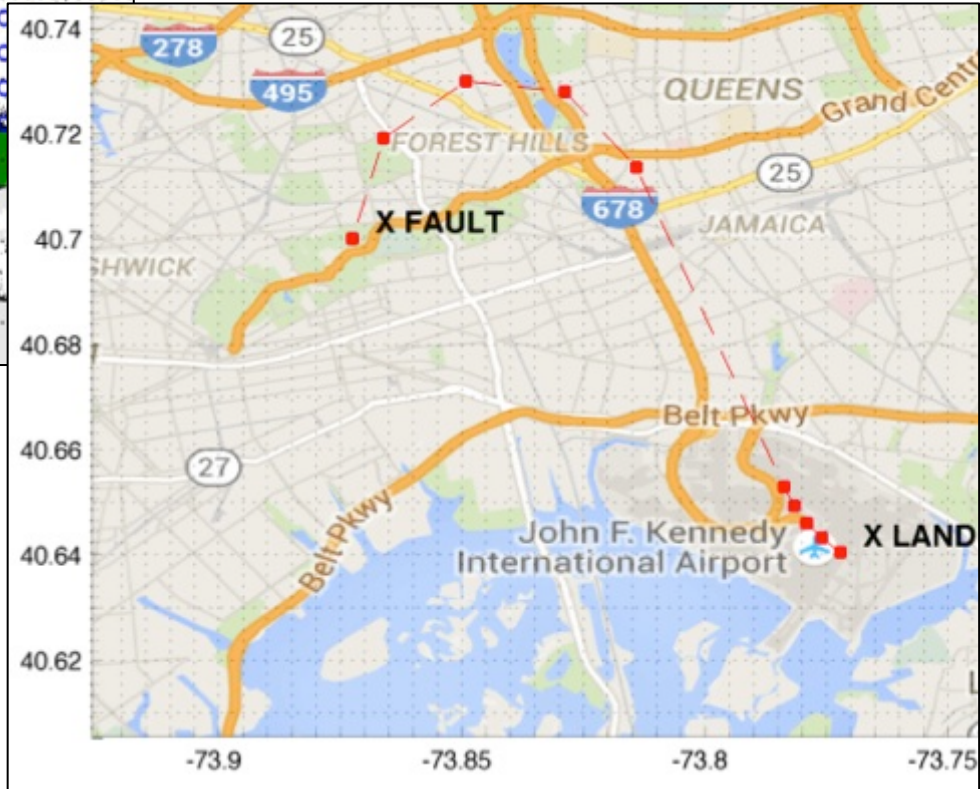
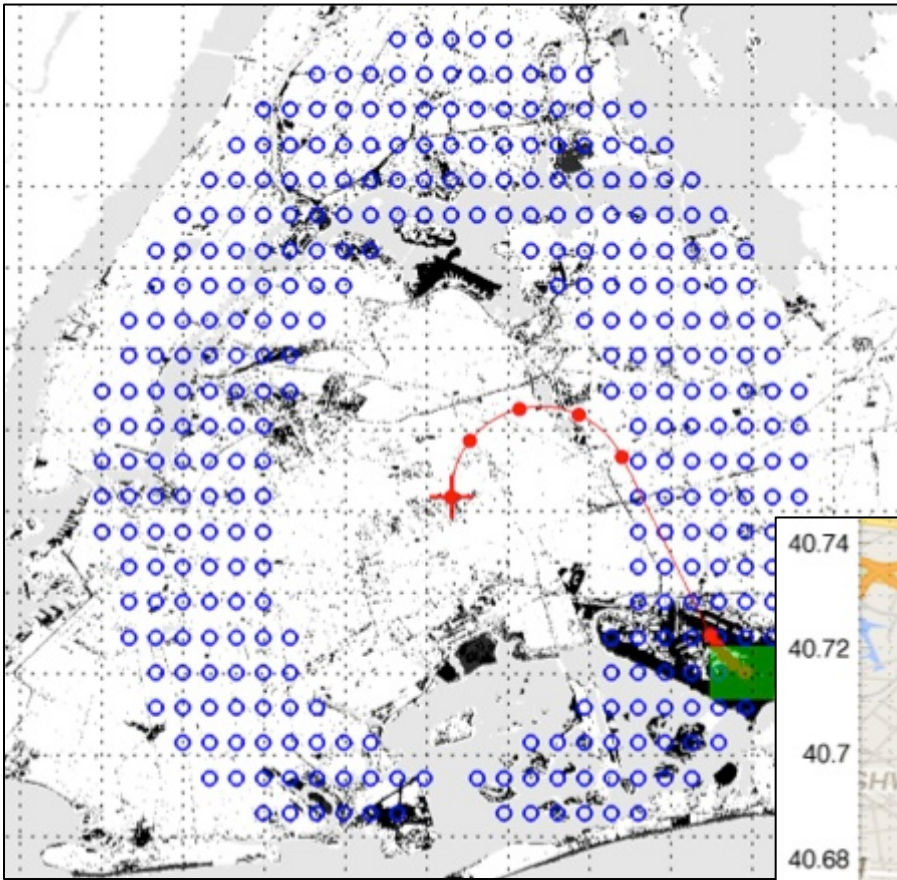
## US Airways 1549

○	Reachable location
+	Current location
■	Landing location
---	Landing trajectory



Cost	
	1.0
	0.5
	0.0







# WTL2 HIL Simulation



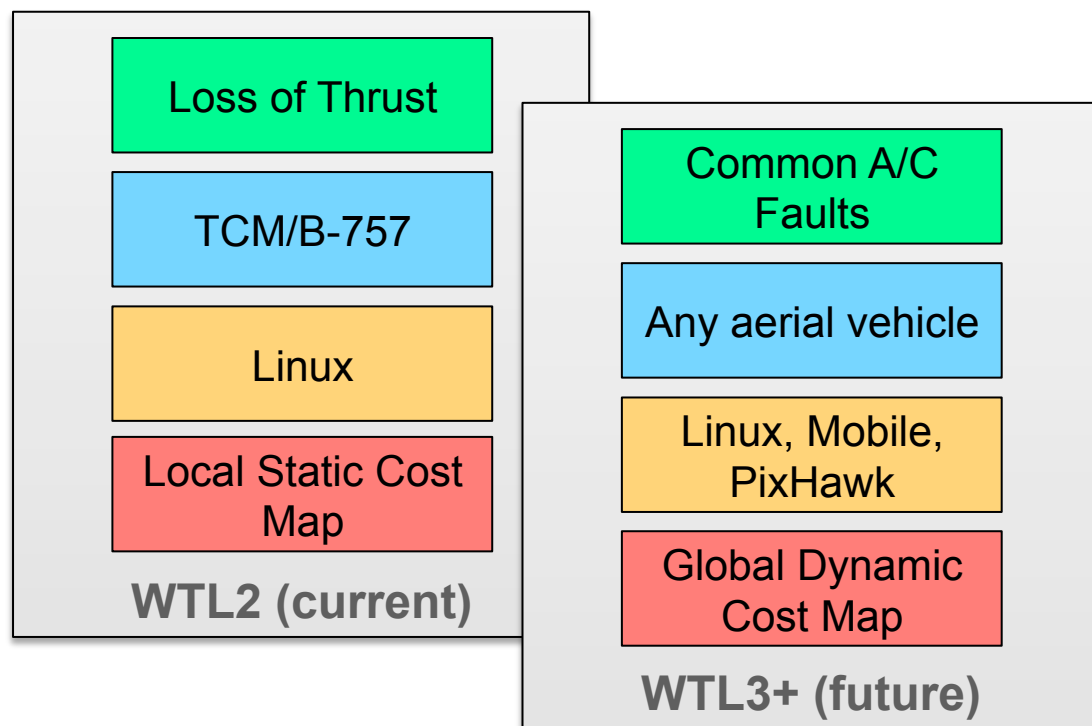
## HIL Simulation Data Overlay

WTL State	On/Off
Target V (kts)	###
Target Heading	###
Waypoint #	##



# Future Work

- “Online” WTL → Fast Estimator/Online Reachable Set
- “Adaptive” WTL → Dynamic trajectories
- WTL on Smartphones, Linux, PixHawk
- WTL + RTA (Run Time Assurance) framework
- WTL + Backward Reachable Controllers





# Impact

- **General Aviation**
  - Pilots tend to be less experienced
  - Mostly single engine aircraft
- **Commercial**
  - Pilots are experienced and well trained
  - Multi engine aircraft
- **Unmanned Vehicles**
  - Flight Termination Systems
  - Lost Link Mode

General Aviation	Can improve odds of survival
Commercial	Gives pilots more options
Unmanned Vehicles	Can enable expanded UAS in the NAS



# Distribution

1. WTL Design: AIAA Conference Paper
2. WTL2 Implementation: AIAA Conference Paper
3. WTL2 NASA Technical Memo
4. NASA NARI Presentation



# References

1. Mitchell, I., Bayen, A., and Tomlin, C.J., “Computing Reachable Sets for Continuous Dynamics Games Using Level Set Methods”
2. Tomlin, C., Lygeros, J., and Sastry, S., “A Game Theoretic Approach to Controller Design For Hybrid Systems”
3. Ding, J., Gillua, H., Huang, H., “Hybrid Systems in Robotics”
4. Adler, A., Bar-Gill and A., and Shimkin, N., “Optimal Flight Paths for Engine Out Emergency Landing”
5. Rogers, D., “The Possible ‘Impossible’ Turn”
6. Atkins, E., “Emergency Landing Automation Aids: An Evaluation Inspired by US Airways Flight 1549”
7. Bayen, A., Mitchell, I., Oishi, M., and Tomlin, C.J., “Aircraft Autolander Safety Analysis through Optimal Control Based Reach Set Computation”
8. Shkel, A., and Lumelsky, V., “Classification of the Dubins Set”
9. Hueschen, R., “Development of the Transport Class Model Aircraft Simulation from a Sub-Scale Generic Transport Model Simulation”