Automatic Dependent Surveillance - Broadcast Enabled, Wake Vortex Mitigation Using
Cockpit Display

A thesis presented to
the faculty of
the Russ College of Engineering and Technology of Ohio University

In partial fulfillment
of the requirements for the degree
Master of Science

Nikhil Tej Gandhi
December 2012
© 2012 Nikhil Tej Gandhi. All Rights Reserved.
This thesis titled
Automatic Dependent Surveillance - Broadcast Enabled, Wake Vortex Mitigation Using
Cockpit Display

by

NIKHIL TEJ GANDHI

has been approved for
the School of Electrical Engineering and Computer Science
and the Russ College of Engineering and Technology by

Michael S. Braasch
Professor of Electrical Engineering and Computer Science

Dennis Irwin
Dean, Russ College of Engineering and Technology
Abstract

GANDHI, NIKHIL TEJ, M.S., December 2012, Electrical Engineering

Automatic Dependent Surveillance - Broadcast Enabled, Wake Vortex Mitigation Using Cockpit Display

Director of Thesis: Michael S. Braasch

Wake turbulence has the potential to cause accidents during the landing phase of flight. This thesis work documents the development and testing of a prototype cockpit display to help pilots avoid this wake turbulence using ADS-B technology. Several landing profiles of the wake generating aircraft and the trailing aircraft were created using computer simulations to generate the synthetic data. A tracking algorithm was developed to compute the altitude of the leading aircraft to be displayed in the altitude strip of the cockpit display. This algorithm will search for all the points of flight data of a leading aircraft, within a predefined region near a trailing aircraft. This flight data is projected on a primary or multifunction display to create an imaginary scenario and tested for data acquisition and transmission from one aircraft to another. The parameters and the number of bits that are necessary to transmit these data via the ADS-B for avoiding the wake turbulence, especially in the case of busy airports are documented. As a final step, necessary hardware was installed in two GA (General Aviation) aircraft and a flight test was performed. The flight-tested prototype successfully provided the pilot of the trailing aircraft with intuitive information on a multifunction display to avoid wake turbulence.
Dedicated to Dad, Mom and Sister ........
Acknowledgements

Foremost, I would like to express my sincere and deepest gratitude to my advisor Dr. Braasch for his invaluable support throughout my research work. This thesis would not have been possible without your suggestions and guidance from the initial to the final level of my research work. I feel honored to have worked with you.

The FAA/NASA Joint University Program for air transportation research funded this research work. My heartfelt thanks to my advisor again for giving me an opportunity, to present my research topic during a quarterly meeting held in Ohio University and making it possible to be approved by the committee members of JUP to fund this research work.

I wish to express my warm and sincere thanks to Jamie Edwards, senior pilot and a research engineer at Ohio University. He has stood by me and guided me in each and every step of my research work and helped me understand several things that I was not aware of with utmost patience. You have always been like a very good friend throughout the course of my research work and made me feel very comfortable working with you. Thank you for everything.

I am grateful to Dr. James Rankin for creating a platform to my research topic and supporting me from distance. I would like to sincerely thank him for accepting me as his student during the initial stages of my research work, which has given me courage to move on with several other complicated things with ease.
I also would like to thank Scott Nykl and Chad Mourning for developing the software to create graphics of my research work.

Special thanks to Jay Clark and Paul Nilles from the Ohio University Avionics Engineering center for installing, testing the equipment in the aircraft.

I would like to thank my Committee members, Dr. Van Graas, Dr. Chris Bartone and Dr. Daniel Phillips for their support and helpful suggestions during my research work. I would also like to take this opportunity to thank my professors Dr. Pelgrum, Dr. Maarten Uijt De Haag along with my committee members and my advisor for their support and guidance during the course work, which has helped me to build my career and learn several new things.

Finally, I would like to thank my family, mom, dad and sister for their extending support in every step of my life.
Table of Contents

Abstract .............................................................................................................................................. 3
Dedication ........................................................................................................................................... 4
Acknowledgements .............................................................................................................................. 5
List of Figures ....................................................................................................................................... 10
List of Tables ......................................................................................................................................... 13
List of Acronyms .................................................................................................................................. 13
1. Introduction ................................................................................................................................. 15
2. Wake Avoidance ............................................................................................................................ 18
   2.1 Wake Vortex Generation: .......................................................................................................... 18
   2.2 Vortex Strength: ....................................................................................................................... 18
   2.3 Wake Turbulence Avoidance Procedures: ................................................................................ 18
   2.4 Wake Vortex Separation ............................................................................................................ 22
3. Survey of Wake Turbulence Avoidance Efforts ............................................................................. 25
   3.1 Stanford University Flight Test of Wake Cockpit Visualization .............................................. 25
   3.2 Proposed Initial wake avoidance system by FAA ................................................................. 29
4. Automatic Dependent Surveillance-Broadcast (ADS-B) ............................................................. 33
   4.1 1090 MHz ADS-B System ......................................................................................................... 34
   4.2 Benefits of ADS-B .................................................................................................................... 36
   4.3 Aircraft Parameters .................................................................................................................. 37
5. Proposed Solution ........................................................................................................................... 40
   5.1. Search Algorithm ..................................................................................................................... 42
   5.1.1 Downwind leg Phase ........................................................................................................... 46
List of Figures

Figure 2.1: Aircraft avoidance procedure during takeoff phase [6] ................................. 19
Figure 2.2: Aircraft avoidance procedure during landing phase [6] ............................... 20
Figure 2.3: Aircraft avoidance procedure when there are parallel runways [6] ........... 21
Figure 2.4: Aircraft avoidance procedure when there are crossing runways [6] ......... 21

Figure 3.1: Coordinate system with respect to the wake generating aircraft and wake
terminology. Reprinted, with permission of the author, from “Flight Deck Display of

Figure 3.2: Dimensions of hazardous portion of wake vortices contained within a box.
Reprinted, with permission of the author, from “Flight Deck Display of Airborne

Figure 3.3: Wake of an aircraft representing rectangular boxes in the following aircraft
cockpit display. Reprinted, with permission of the author, from “Flight Deck

Figure 3.4: High-level system diagram for Initial and Midterm Ground based Wake
applications using ADS-B [8] ................................................................. 30

Figure 4.1: 1090 MHz ADS-B system and its major components [9] ......................... 35

Figure 5.1: Airfield traffic pattern ................................................................. 42
Figure 5.2: Spherical cone showing cone axis and test vector .................................. 43
Figure 5.3: Flow diagram of the algorithm to search for points within a spherical cone
................................................................................................................. 45
Figure 5.4: Search algorithm implementation for downwind leg phase ...................... 46
Figure 5.5: Search algorithm implementation for base leg phase .............................. 47
Figure 5.6: Search algorithm implementation for final approach .................................................. 48

Figure 6.1: Aircraft arrival scenario 1 - KUNI airport and the altitude strip for the upwind and downwind phases of flight ........................................................................................................... 50

Figure 6.2: Aircraft arrival scenario 1 - KUNI airport and the altitude strip for the base leg phases of flight ........................................................................................................................................ 52

Figure 6.3: Aircraft arrival scenario 1 - KUNI airport and the altitude strip when the aircraft is on final approach ........................................................................................................................................ 52

Figure 6.4: Vertical plot for scenario 1 - KUNI airport ............................................................................. 53

Figure 6.5: Aircraft arrival scenario 2 KUNI airport — Parallel straight in approaches .................. 54

Figure 6.6: Vertical plot for scenario 2 - KUNI airport .............................................................................. 55

Figure 6.7: Enlarged image of vertical plot showing wake altitude and GA aircraft altitude .......................................................... 56

Figure 6.8: Enlarged image of 3d plot showing the trailing aircraft and GA aircraft ...... 57

Figure 6.7 Aircraft arrival scenario 3 for KBOS airport - parallel runways................................. 58

Figure 6.8 Vertical plot for scenario 3 - KBOS airport ........................................................................... 58

Figure 6.9 Aircraft arrival scenario 4 - May 2\textsuperscript{nd}, 2011 flight test data, KUNI airport ..... 60

Figure 6.10 Vertical plot for scenario 4- May 2\textsuperscript{nd}, 2011 flight test data, KUNI airport .... 61

Figure 6.11 Enlarged image of vertical plot for scenario 4 ................................................................. 62

Figure 6.12 Vertical plot for scenario 4 – Heavy aircraft flight path .............................................. 62

Figure A.1: Block Diagram for the equipment to be installed in the simulated ‘trailing’ aircraft (Baron) ............................................................................................................................................. 68

Figure A.2: Block diagram for the equipment to be installed in the simulated ‘heavy’ aircraft (Piper Saratoga) .............................................................................................................................................. 68
Figure A.3: Leading aircraft used for the flight test – Piper Saratoga .......................... 71

Figure A.4: Trailing aircraft used for the flight test – Baron ........................................ 71

Figure A.5: Equipment installed in the trailing aircraft – Baron ................................. 72

Figure A.6: Equipment installed in the leading aircraft – Piper Saratoga ....................... 73

Figure B.1: Multi function display installed in front of the pilot to visualize the real time
virtual world [15]. ........................................................................................................ 73

Figure B.2: Screen shots showing the Trailing aircraft joining the final approach of the
leading aircraft and the wake strip [15]. ........................................................................ 74
List of Tables

Table 1: Separation distance during en-route phase of flight [6]........................................23
Table 2: Separation timing during departing phase of flight [6] ........................................23
Table 3: Separation timing during landing phase of flight [6] ........................................24
Table 4: ADS-B airborne position message format ..........................................................38
**List of Acronyms**

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASRS</td>
<td>Aviation Safety Reporting System</td>
</tr>
<tr>
<td>ADS-B</td>
<td>Automatic Dependent Surveillance Broadcast</td>
</tr>
<tr>
<td>AHRS</td>
<td>Attitude and Heading Reference System</td>
</tr>
<tr>
<td>ATC</td>
<td>Air Traffic Controllers</td>
</tr>
<tr>
<td>UTC</td>
<td>Coordinated Universal Time</td>
</tr>
<tr>
<td>CPR</td>
<td>Compact Position Reporting format</td>
</tr>
<tr>
<td>EFIS</td>
<td>Electronic Flight Information System</td>
</tr>
<tr>
<td>FAA</td>
<td>Federal Aviation Administration</td>
</tr>
<tr>
<td>FIS-B</td>
<td>Flight Information Service Broadcast</td>
</tr>
<tr>
<td>FMS</td>
<td>Flight Management System</td>
</tr>
<tr>
<td>GA</td>
<td>General Aviation</td>
</tr>
<tr>
<td>GNSS</td>
<td>Global Navigation Satellite System</td>
</tr>
<tr>
<td>HAE</td>
<td>Height above the Ellipsoid</td>
</tr>
<tr>
<td>ICAO</td>
<td>International Civil Aviation Organization</td>
</tr>
<tr>
<td>ILS</td>
<td>Instrument Landing System</td>
</tr>
<tr>
<td>MCTOW</td>
<td>Maximum Certified Takeoff Weight</td>
</tr>
<tr>
<td>NASA</td>
<td>National Aeronautics and Space Administration</td>
</tr>
<tr>
<td>NMEA</td>
<td>National Marine Electronics Association</td>
</tr>
<tr>
<td>NTSB</td>
<td>National Transportation Safety Board</td>
</tr>
<tr>
<td>SSR</td>
<td>Secondary Surveillance Radar</td>
</tr>
<tr>
<td>TCAS</td>
<td>Traffic Alert and Collision Avoidance</td>
</tr>
<tr>
<td>TIS-B</td>
<td>Traffic Information Service Broadcast</td>
</tr>
<tr>
<td>UAT</td>
<td>Universal Access Transceiver</td>
</tr>
<tr>
<td>VOR</td>
<td>VHF Omni-directional Range</td>
</tr>
<tr>
<td>WAAS</td>
<td>Wide Area Augmentation System</td>
</tr>
</tbody>
</table>
1. Introduction

On July 27th, 1996 a Cessna 182G, N3135S, was scheduled to land at Portland International airport, Portland, Oregon. At approximately 7:27 PM PDT, this aircraft encountered the wake turbulence created by a Reno Air MD-80 and was substantially damaged when it collided with terrain following a loss of control on final approach. As per the full narrative stated in the National Transportation Safety Board (NTSB) report [1], the Cessna was given clearance to enter a left base for Runway 28R to follow the MD-80 which was on a straight-in final for the same runway and cautioned for wake turbulence. The separation distance between the Cessna and MD-80 at that time was “between 2.3 and 3.6 nautical miles”[1]. As per this report, the pilot of Cessna N3135S stated that he encountered wake turbulence generated by the MD-80 a couple of hundred feet above the ground. He also stated that no mechanical malfunction or failure was involved and recommended further separation be provided by air traffic control [1]. This accident was not a solitary incident: “NTSB has documented 51 instances of wake turbulence which resulted in 27 fatalities and 40 destroyed aircraft[2].

According to the statistics mentioned in the Flight Safety Digest release March-April 2002 for the US wake turbulence accidents and incidents, 130 accidents were reported from the NTSB data, out of which 116 were non-fatal accidents and, 14 were fatal accidents. Based on the Federal Aviation Administration (FAA) data, 60 incidents were reported in this release. From the US National Aeronautics and Space Administration (NASA) Aviation Safety Reporting System (ASRS) data on 165 events, the majority of the accidents happened during the landing and takeoff phases of the flight (37 accidents
during takeoff phase of the flight and 106 during the approach and landing phases of the flight). From the statistics mentioned in this document, out of 130 incidents that are reported, 92 of them involve a trailing aircraft of small category (weighing 5000 pounds or less) and 21 of them are the small transport aircraft (weighing in between 5,001 to 14,500 pounds) encountered for wake turbulence [3].

A violent air disturbance, which is created behind a passing aircraft whenever it is generating lift, is termed wake turbulence and it is due to counter rotating vortices trailing from the wing tips [4]. These vortices can remain in the air for up to three minutes after the passage of an aircraft and encountering wake might be extremely dangerous to smaller category aircraft. The effect of wake turbulence on an aircraft depends on several factors, which include size of an aircraft, its phase of flight and the weather conditions. As mentioned in the statistics above, this turbulence is especially hazardous during the landing and takeoff phases of the flight, as the aircraft in these phases operates close to its stall speed and close to the ground where there is very little margin for recovery.

Pilots generally follow a few basic rules in avoiding wake turbulence while arriving during visual meteorological conditions.

1. Not to get too close to the preceding aircraft,

2. Not to get below the preceding aircraft flight path as the wake generated by the aircraft descends to a point 400 to 900 feet below wake-generating aircraft.

3. To be distinctly cautious when light wind conditions exist [4].
Pilots rely totally on the Air Traffic Controllers (ATC) whenever they are unable to see the preceding aircraft (i.e., during the instrument meteorological conditions). The physical separation provided by ATC commands is used to avoid the wake [5].

Automatic Dependent Surveillance – Broadcast (ADS-B) is a system whereby an aircraft transmits its state vector (position and altitude information) periodically to air traffic controllers and other aircraft at distances up to 200 miles [9]. Unlike traditional ground-based radar surveillance, this system provides accurate information even at low altitudes and in mountainous areas.

Vortices can be caused by mountain waves and microbursts but wake caused by aircraft is the only type of vortex considered in this research. With the help of ADS-B technology, intuitive information will be provided on a multi-function display so that a pilot can maintain the safe separation distance and avoid the wake turbulence generated by preceding aircraft.
2. Wake Avoidance

2.1 Wake Vortex Generation:

Explanation for the generation of wake vortex is available from the document titled “Wake Turbulence” published by the Civil Aviation Authority of New Zealand [6]. According to this document “Lift is generated by the creation of pressure differential over wing surfaces. The lowest pressure occurs over the upper wing surface and the highest pressure occurs under the wing. Air will always want to move towards the area of low pressure. This causes it to move outwards under the wing towards the wingtip and curl up and over the upper surface of the wing. This starts the wake vortex”[6].

2.2 Vortex Strength:

The wake vortex strength of an aircraft depends on the gross weight, speed at which it is travelling, shape of the wing of that particular aircraft and increases proportionately with the aircraft operating weight. The strongest vortex strength occurs when the generating aircraft is heavy and flying in a clean configuration at high angles of attack [4].

2.3 Wake Turbulence Avoidance Procedures:

Wake vortices have some behavioral characteristics and hence pilots follow several ways to avoid wake turbulence during the departure and take off phases of the flight.

2.3.1 Takeoff Phase:

As the wake vortices are a by-product of the wing lift, they are generated from the moment the aircraft leaves the ground. As shown in Figure 2.1, the preceding aircraft is
considered to be of the heavy category and the trailing aircraft is considered to be of lighter category. During the take off phase of the flight, pilots try to maintain an altitude greater than the preceding aircraft by selecting a rotation point which comes earlier to that of the preceding aircraft. In the case of low missed approaches or a touch and go landing of a preceding aircraft, the vortices settle on the runway and move laterally near the ground when light wind conditions exist. In these situations pilots try to depart or land after an interval of 2 minutes has elapsed. [4]

**Figure 2.1: Aircraft avoidance procedure during takeoff phase** [6]

During the first 30 seconds of the flight, wake vortices of heavy category aircraft spread laterally up to 5 miles and descend at approximately 300 to 500 feet per minute [6]. Hence pilots try to choose a flight path at or above the preceding aircraft [4].
2.3.2 Landing Phase:

During the landing phase, as shown in the Figure 2.2, the pilot of the trailing aircraft tries to choose a flight path at or above the larger aircraft final approach flight path during visual meteorological conditions. The pilot typically will attempt to note the touchdown point of the leading aircraft and then try to land beyond it.

![Figure 2.2: Aircraft avoidance procedure during landing phase [6]](image)

In the case of parallel runways as shown in the Figure 2.3, the aircraft wake sinks towards the ground and may move laterally away from the runway even when the wind is calm. A 3 to 5 knot crosswind will keep the upwind side of the wake in the runway area and the downwind side to drift onto the parallel runway [6]. Hence, pilots try to adjust the takeoff and landing points accordingly.
Figure 2.3: Aircraft avoidance procedure when there are parallel runways [6]

Similarly in the case of a crossing runway showed in Figure 2.4, pilots aim to avoid the wake by either landing past the portion of affected runway, or by landing well before it [4].

Figure 2.4: Aircraft avoidance procedure when there are crossing runways [6]
2.4 Wake Vortex Separation

Based on the Maximum Certified Takeoff Weight (MCTOW) of the aircraft, the International Civil Aviation Organization (ICAO) divides aircraft into 3 categories.

- Light – MCTOW of 7,000 kilograms (15,000 lbs.) or less,
- Medium – MCTOW of greater than 7,000 kilograms, but less than 136,000 kilograms (300,000 lbs.),
- Heavy – MCTOW of 136,000 kilograms or greater. [6]

The wake vortex separation criteria are based upon these three categories for takeoff, landing and en-route phases of flight.

The separation distances during the en-route phase (listed in the Table 1 below) are applicable when the trailing aircraft is directly behind the preceding aircraft or is crossing behind, at the same level and up to 1000 feet below the preceding aircraft [6].

The separation timings listed in Table 2 and Table 3 are used for the operations on the same runway and parallel runways separated by a distance less than 760 meters [6].
Table 1: Separation distance during en-route phase of flight [6]

<table>
<thead>
<tr>
<th>Preceding aircraft</th>
<th>Following aircraft</th>
<th>Minimum radar separation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heavy</td>
<td>Heavy</td>
<td>4 NM</td>
</tr>
<tr>
<td></td>
<td>Medium</td>
<td>5 NM</td>
</tr>
<tr>
<td></td>
<td>Light</td>
<td>6 NM</td>
</tr>
<tr>
<td>Medium</td>
<td>Light</td>
<td>5 NM</td>
</tr>
</tbody>
</table>

Table 2: Separation timing during departing phase of flight [6]

<table>
<thead>
<tr>
<th>Preceding aircraft</th>
<th>Following aircraft</th>
<th>Minimum spacing at time aircraft</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>departing from same take off position</td>
</tr>
<tr>
<td>Heavy</td>
<td>Heavy</td>
<td>2 min</td>
</tr>
<tr>
<td></td>
<td>Medium</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Light</td>
<td></td>
</tr>
<tr>
<td>Medium</td>
<td>Light</td>
<td>3 min</td>
</tr>
</tbody>
</table>
Table 3: Separation timing during landing phase of flight [6]

<table>
<thead>
<tr>
<th>Preceding aircraft</th>
<th>Following aircraft</th>
<th>Minimum time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heavy</td>
<td>Heavy</td>
<td>2 min</td>
</tr>
<tr>
<td></td>
<td>Medium</td>
<td>2 min</td>
</tr>
<tr>
<td></td>
<td>Light</td>
<td>3 min</td>
</tr>
<tr>
<td>Medium</td>
<td>Light</td>
<td>3 min</td>
</tr>
</tbody>
</table>

Although these separation standards are intended to prevent wake encounters, encounters can still occur as these are considered to be the minimum separation standards [6].
3. Survey of Wake Turbulence Avoidance Efforts

Several procedures to avoid wake turbulence were proposed in recent years by distinguished professionals. In this chapter, these procedures and their limitations will be discussed.

3.1 Stanford University Flight Test of Wake Cockpit Visualization

In 2001, Holforty and Powell at Stanford University developed a flight deck display, showing the neighboring aircraft and the wake generated by it using graphical representations to the pilot. The model used in this application (to predict the geographical location of the wake) is based on theoretical equations which can be derived from the aircraft parameters readily available on-board [5].

3.1.1 Wake terminology

The aircraft parameters used in this model are shown in the Figure 3.1 to describe the wake terminology. They are listed as follows:

\[ b_g = \text{Wing span of the generating aircraft} \]

\[ b' = \text{Wake vortex span referred to as the distance between the centers of two counter rotating vortices.} \]

\[ L = \text{Lift} \]

\[ W = \text{Weight} \]

\[ \Theta \Theta = \text{Vortex strength} \]

\[ V_\infty = \text{True airspeed} \]

\[ u = \text{Horizontal component of wind speed excluding cross wind.} \]

\[ \rho = \text{Air density [5]} \]
Figure 3.1: Coordinate system with respect to the wake generating aircraft and wake terminology. Reprinted, with permission of the author, from “Flight Deck Display of Airborne Traffic Wake Vortices”, by Wendy L. Holforty[5].

The cockpit wake visualization model developed to avoid wake turbulence was derived from the research work on wake vortices conducted by Rossow in [7]. He developed a model by comparing the wake induced rolling moment of the wake vortex and the aileron induced rolling moment of the following aircraft and concluded that the hazardous portion of the wake could be wholly contained within a box \((y_g-z_g plane)\) the size of \(2b_g \times 1 b_g\) [7] as shown in the Figure 3.2.
3.1.2 Self induced descent velocity

Self induced descent velocity is derived from the vortex strength ($\Theta \Theta$) and the vortex span ($b'$), and is defined as how fast the wake descends behind the generating aircraft and out of the flight path of the follower aircraft. The time-averaged value for the vortex sink rate is given by the equation

$$w = \frac{-\Gamma}{2\pi b'}$$

where $\Gamma = \frac{L}{\rho V_\infty b'}$

A flight with constant velocity of Weight $W$ is substituted for lift $L$. Sink rate can be derived by combining the above two equations and is given as
\[ w = \frac{-W}{2 \pi \rho V_{\infty} b'^2}. \] [5]

### 3.1.3 Descent distance

Wake vortices descend downwards from the generating aircraft to a certain level and they remain in that level until they totally dissipate. The equation for the descent distance \( Z_{\text{limit}} \) is a function of time since the wake has been generated and given by the equation

\[ Z_{\text{limit}} = wt. \] [5]

From the research conducted by Crow and Bates referenced in [5], the wake from most commercial aircraft descends between 500 and 1000 feet which is around \( 6b' \) (\( b' \) is the distance between centers of two counter rotating vortices).

### 3.1.4 Graphical presentations of wake planes

Six different wake representations were generated in Holforty and Powell’s research, to display on the cockpit, for the pilots to avoid the wake turbulence. In Figure 3.3, filled rectangles are represented as wake and in order to show traffic, terrain, open rectangles or open oval representations are used. Different color schemes are also used for the wake representation to help pilot realize if he/she is at the beginning or end of the wake.
Figure 3.3: Wake of an aircraft representing rectangular boxes in the following aircraft cockpit display. Reprinted, with permission of the author, from “Flight Deck Display of Airborne Traffic Wake Vortices”, by Wendy L. Holforty[5].

The above Figure shows the graphical representation of the wake planes as a filled rectangle generated for every one second. The nominal path indicator shown in this representation guides the pilot through the tunnel, with the predictor symbol to avoid the wake. This picture also shows horizontal deviation indicator, vertical deviation indicator, roll indicator, ground speed indicator and altitude indicator [5].

3.2 Proposed Initial wake avoidance system by FAA

The FAA wake turbulence program has proposed an initial ground-based wake turbulence avoidance system. “ADS-B equipped aircraft with appropriate provisioning, which can measure and report meteorological data at a high resolution, under all weather
conditions and over regions of operational interest” are necessary for these applications [8].

As shown in the Figure 3.4, the aircraft ADS-B processor will obtain required meteorological data to be formatted from onboard sensors and computer systems. This data will be broadcast with the necessary frequency by ADS-B “In” (discussed in the next section). With the help of ground receivers, the ground ADS-B processor will obtain these data messages to parse them. The ground based wake processor will obtain the required data elements from the ADS-B processor, ground data networks and will construct atmospheric profiles using a meteorological data algorithm present in this processor. 4-D wake free trajectories are determined for each aircraft and are processed by a 4D trajectory processor. These processed trajectories will be provided to ATC ground automation for use in ATC separation functions.

Figure 3.4: High-level system diagram for Initial and Midterm Ground based Wake applications using ADS-B [8].
The data elements that are necessary for this wake avoidance system are listed below and can be obtained from the wake generating aircraft:

1. Wind speed
2. Wind direction
3. Static temperature
4. Static barometric pressure
5. Aircraft emitter category
6. Aircraft position
7. Pressure altitude
8. Aircraft speed and heading
9. Aircraft weight.
10. Atmospheric turbulence (Eddy dissipation rate / total kinetic energy)
11. Aircraft configuration (for future applications).

These elements that are proposed by the FAA are necessary for long-term applications such as real time prediction and movement of aircraft wake vortices [8].

Since wake depends on several characteristics such as weight, shape and size of the generating aircraft, wind conditions, proximity of the aircraft to the ground and time elapsed since generation; it is extremely complex to predict wake vortices accurately using mathematical modeling or theoretical equations. Hence the applications discussed in this chapter are limited by their ability to predict wake movement with safety-critical
levels of accuracy and integrity. By contrast, the system developed in this thesis avoids the prediction problem by providing the pilot with information sufficient to always stay above the wake.
4. Automatic Dependent Surveillance-Broadcast (ADS-B)

ADS-B is a technology that periodically broadcasts the aircraft state vector (horizontal and vertical position, horizontal and vertical velocity) and other information to airborne and surface aircraft, and other surface vehicles operating in the region of an airport surface movement area. Transmission of this data is also referred to as ADS-B “Out”. It uses the signals from the aircraft-positioning source to provide air traffic controllers and pilot with very accurate information so as to maintain safe separation distance in the sky and on runways between one another [9,10,16].

The ADS-B technology works using 1090 MHz ADS-B transponders or 978MHz Universal Access Transceiver (UAT) equipment that are installed on aircraft. Ground-based transmitters rebroadcast 1090 messages on 978 (and vice versa) so that all aircraft can receive all broadcasts regardless of the link used on an individual aircraft. The ADS-B equipment installed in each aircraft typically uses GPS to determine the precise location of the aircraft. The aircraft unit then forms and broadcasts a message with its position and velocity along with other data such as aircraft type, flight number and flight dynamics (turn, climb, descent) [10]. As the name is derived ADS-B is automatic because it periodically transmits data 1Hz and no pilot operator input is required; it is dependent because it relies on the 3D position and velocity vector which are derived from the aircraft’s own position-determination hardware/software (as opposed to ground-based radar) in order to provide surveillance (method of determining position of aircraft, vehicles or other asset) information to other users. Any aircraft or surface vehicle equipped with ADS-B and within the vicinity of the aircraft originating the broadcast
(transmitting the state vector available to anyone with appropriate receiver equipment) can receive the information without any permission. The communication channel can be either Air to Air, Air to Ground or Ground to Air [9, 10].

Aircraft equipped with the ADS-B and ground stations at distances up to 200 miles will receive the broadcast from the transponder. For aircraft that are not equipped with ADS-B, the ground stations will provide radar based targets and send the entire information back up to the ADS-B equipped aircraft, which is called the Traffic Information Service Broadcast (TIS-B). The ADS-B ground stations also send out graphical information from the National Weather Service and flight information, such as temporary flight restrictions, which is called the Flight Information Service Broadcast (FIS-B). Reception of ADS-B Out data from a nearby aircraft and services provided by the ground systems like TIS-B, FIS-B data is referred to as ADS-B “In” [10, 16].

4.1 1090 MHz ADS-B System

The ADS-B messages transmitted on the 1090 MHz are the Mode S (transponder based subsystems) extended squitter (Airborne position squitter, Airborne velocity squitter, Surface position squitter, etc.) [9].

Figure 4.1 shows the extent of the 1090 MHz ADS-B system and its major components. The 1090 MHz ADS-B system consists of a transmitting subsystem and a receiving subsystem.
Transmitting Subsystem – This system takes, as input, “the PVT (Position, Velocity, Time), status and intent inputs from the other systems onboard aircrafts” [9]. The message generation function assembles and structures the data to be delivered via the 1090 MHz frequency extended squitter. “ADS-B Message Exchange Function includes the radio equipment (modulator /transmitter) and transmitting antenna sub functions” [9]. This function transfers the data at the required rate across the 1090 MHz frequency link. “It encompasses both the transmission and reception of the data”[9].

The transmitting subsystem can be implemented either by transponder (Mode–S secondary surveillance radar transponder) or non-transponder based 1090 MHz transmitting equipment [9].

Figure 4.1: 1090 MHz ADS-B system and its major components [9]
Receiving Subsystem – This system takes the ADS-B messages and transmits the data to other systems onboard the aircraft. “ADS-B Report Assembly Function contains radio equipment (receiver/demodulation) sub functions”[9]. This function structures the received data into the appropriate ADS-B reports for use by onboard applications [9].

4.2 Benefits of ADS-B

There are several benefits of using the ADS-B technology for both pilots and air traffic controllers. Some of them are

- Pilots and air traffic controllers for the very first time, will be able to see the real time display of air traffic in the air around them which will help them to know the position of their own aircraft with greater accuracy which in turn increases the situational awareness [11].
- This will allow the pilot to maintain a safe separation distance from other aircraft during the en-route phase with fewer instructions from ground-based controllers.
- With the help of onboard avionics and terrain maps, pilots will be able to see where they are with respect to the ground during nights and in poor visual conditions [10].
- Both the pilots and air traffic controllers will be able to see the ADS-B equipped ground vehicles and aircraft on runway maps, along with the data that shows where they are moving. This will help reduce runway incursion [10].
- This is a substitute for secondary surveillance radar (SSR), and adopting ADS-B has several advantages. ADS-B directly receives data from the transmitters, but as
in radars they passively scan for inputs, which eliminate clutter. SSR are very expensive to deploy due to their very large structures and they need lots of maintenance whereas ADS-B ground stations are of the size of a refrigerator and are very inexpensive [10].

- In the future ADS-B may replace TCAS-II (Traffic alert and collision avoidance) entirely [11].

ADS-B is being deployed in several countries like the USA, Australia, Canada, China and Sweden. According to ADS-B final rule released in 2010, every aircraft must be ADS-B equipped by January 1st 2020. This makes ADS-B the primary source for surveillance. As discussed earlier, “the purpose of TIS-B is to provide proximate traffic information about targets that are not equipped with ADS-B [16]”. After 2020, the services of TIS-B will be used for ADS-B ‘In’ as a backup traffic service during GNSS outages or when individual target ADS-B system is not functional [16].

4.3 Aircraft Parameters:

The aircraft parameters that are considered for transmission via the ADS-B Out are listed below. Each parameter is assigned a specific number of bits and used to avoid wake turbulence [9].

1. Aircraft Type – 5 bits
2. Surveillance status – 2 bits
3. Single Antenna - 1 bit
4. Altitude – 12 bits
5. Time – 1 bit
6. Compact Position Report (CPR) format – 1 bit
7. Encoded Latitude – 17 bits
8. Encoded Longitude – 17 bits

The Airborne position message format with all the required parameters is shown below [9]

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Surveillance status</th>
<th>Single Antenna</th>
<th>Altitude</th>
<th>Time</th>
<th>CPR format</th>
<th>Encoded Latitude</th>
<th>Encoded Longitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>ME bit</td>
<td>1-5</td>
<td>6-7</td>
<td>8</td>
<td>9-20</td>
<td>21</td>
<td>22</td>
<td>23-39</td>
<td>40-56</td>
</tr>
<tr>
<td>Message</td>
<td>33-37</td>
<td>38-39</td>
<td>40</td>
<td>41-52</td>
<td>53</td>
<td>54</td>
<td>54-71</td>
<td>72-88</td>
</tr>
</tbody>
</table>

The “ME” field is a 56-bit field (Message bits 33 through 88) that occurs in every 1090 MHz Extended Squitter Message. The ME field carries the bulk of the data in ADS-B and TIS-B Messages [9].

The “Type” subfield is used to identify the specific ADS-B message among eight different message types that are available. Airborne position message type is one among them [9].

The “Surveillance Status” subfield is used to encode information from the aircraft’s Mode – A (non transponder based subsystem) transponder code. With this subfield, alert
condition when in emergency and special position identification condition can be chosen [9].

The “Single Antenna” subfield is used to identify whether the ADS-B transmitting subsystem is using single or two antennas [9].

The “Altitude” subfield is used to transmit either altitude or barometric altitude information. GNSS height above the Ellipsoid (HAE) is used to report in the altitude subfield. This data will be encoded in the altitude subfield [9].

The “CPR format” subfield indicates whether the frame is either odd or even. This format is developed to reduce the number of bits required to broadcast latitude and longitude data by encoding [9].

The “Time” subfield indicates whether the time of applicability of the position data is synchronized to an exact 0.2 seconds Coordinated Universal Time (UTC) epoch [9].

As the field name indicates, next two subfields contain Encoded Latitude and Longitude information of the airborne position.
5. Proposed Solution

Wake turbulence avoidance procedures proposed by Stanford University and the FAA, described in Chapter 3, essentially use meteorology to determine/predict the location of the wake and display it. Forecasting by its very nature is probabilistic and subject to error. The solution proposed in this thesis does not require any prediction of the wake.

The operational procedures to avoid wake turbulence while landing [4] recommend that the trailing aircraft stay high and land long while the wake descends to an altitude 400 to 900 feet below the wake-creating aircraft. ATC does not provide visual wake turbulence separation to arrival aircraft and it is the responsibility of the pilot to avoid the wake. The system proposed in this research involves the use of a red strip on the altitude tape (of the primary flight display) to indicate the altitude of the wake-generating aircraft when it was close to the current location of the trailing aircraft. The system in the trailing aircraft uses the ADS-B transmissions of the wake-generating aircraft to know its position history.

To demonstrate this, let us consider a scenario where the pilot in the GA (i.e., trailing) aircraft has been cleared to land behind the heavy aircraft and has been cautioned about possible wake turbulence by ATC. The GA and the Heavy aircraft have the new ADSB equipped wake turbulence information system incorporated into its avionics suite. When the aircraft is on the downwind leg the system is enabled and the pilot selects the appropriate landing runway. The pilot looks at the "red tape" on the altitude strip of the electronic flight information system (EFIS) display and uses it to help determine when to
perform the turn onto base leg, to stay constantly above the wake on the base and final approaches.

At 4 NM from runway threshold the leading aircraft begins descending down the glide slope and the GA aircraft continues on the downwind leg. The "red tape" area is progressively lowering as both aircraft continue on their respective flight paths. The pilot of the GA aircraft can commence its turn to base when the "red tape" area is below his current altitude. At this point the GA aircraft is above the path of the leading aircraft and the pilot can begin his descent while aiming for a runway touchdown point beyond that of where the leading aircraft touched down. The pilot continues to monitor the "red tape" information to assure that he remains above the flight path of the leading aircraft until landing. Note: The system accounts for the fact that wake turbulence older than 3 minutes is not a factor and is no longer displayed.

During departure, air traffic controllers will provide wake turbulence separation by applying time intervals. ATC will inform the pilot to “hold for wake turbulence”. As discussed earlier, when a small aircraft is departing behind a large passenger jet on the same runway, separation time is 2 min for same direction and 3 min for opposite direction [4]. Also as described earlier, small aircraft rotate and depart the runway at a shorter distance down the runway than larger aircraft and are then vectored away from the departure path of the larger, previously departing, aircraft.
5.1. Search Algorithm

An algorithm was developed to compute the highest altitude of a pre-defined portion of the flight path of a leading aircraft. This altitude will be displayed on the altitude strip of the primary flight display in the trailing aircraft. This algorithm searches for all the points of flight data of a leading aircraft that lie within a specified region of a trailing aircraft. As shown in the Figure 5.1, three phases of flight are considered: i.e., when the plane is flying on: 1) the downwind leg of the traffic pattern, 2) the base leg of the traffic pattern and 3) the final approach. As will be described further in sections 5.1.1 through 5.1.3, a spherical cone volume is used to search for the relevant portion of the leading aircraft’s flight path. A spherical cone was chosen as the search volume to trade off the computational complexity of the search algorithm with the need for a focused search volume. Figure 5.2 shows the spherical cone with all the parameters, cone axis and test vector. The apex of the spherical cone is located at the current position of the trailing aircraft and the cone axis is directed approximately along the aircraft velocity vector. Since, in normal operation, the aircraft is either flying level or is on a glide path angle of approximately 3 degrees, the cone axis is set to have only a local-horizontal component (i.e., no vertical component).

*Figure 5.1: Airfield traffic pattern*
In this algorithm, it is assumed that the wake turbulence is created for every position of the leading aircraft and it remains stationary for a fixed amount of time until it dissipates. Hence, the flight path of the leading aircraft, in the vicinity of the trailing aircraft is considered to be a quite conservative prediction of the wake.

**Figure 5.2: Spherical cone showing cone axis and test vector**

- A – Apex of the spherical cone (Position of the trailing aircraft in ECEF coordinates)
- B – A given position of the leading aircraft in ECEF coordinates (test point)
- h – Height of the spherical cone
- d – Distance between points A and B
- $\theta$ – Half angle which defines the angle between one edge of the spherical cone and the major axis. This angle is given as an input and varies with the phases of flight.
- $\hat{a}$ – Vector defining spherical cone major axis of unit length, which begins at the apex and points in the direction that the cone expands.

Spherical cone axis is given by the formula
Spherical cone axis \( \hat{a} = \frac{V}{|V|} \)

Where, \( V \) is the horizontal velocity vector

\( \vec{b} \) — Test vector of unit length from A to B, which is given by

\[
\vec{b} = \frac{B-A}{|B-A|}
\]

Angle between spherical cone axis and test vector is calculated using the formula

\[
\phi = \arccos (\hat{a} \cdot \vec{b})
\]

If angle ‘\( \phi \)’ is less than half angle of the cone ‘\( \theta \)’, then the point B might be inside the cone subject to a distance check (to be described shortly). Since some angle calculations also work on points inside the negative cone (the mirror image of the cone in the opposite direction along the major axis) the result of the dot product of the two vectors is used only when it is positive. In the actual algorithm, distance ‘\( d \)’ and height ‘\( h \)’ are first compared. If distance ‘\( d \)’ is less than or equal to ‘\( h \)’, then the point B might be inside the cone. The angle check is then performed to determine if the point is indeed inside the cone.

Among all the points that lie within the volume of this cone, the one with the highest altitude is then displayed on the altitude strip.

The flow diagram of the search for points within the spherical cone is shown in Figure 5.3:
Figure 5.3: Flow diagram of the algorithm to search for points within a spherical cone.
In the following sections, the geometry of the intersection volumes for every phase of the flight is discussed. It must be noted that the cone lengths and angles described are an ad hoc attempt to capture the relevant portion of the leading aircraft’s flight path. Optimization of these parameters is a topic for future research.

5.1.1 Downwind leg Phase

During the downwind phase of flight, the pilot in a trailing aircraft has to determine when to turn onto the base leg in order to stay above the path of the previously landing heavy aircraft which is on a straight in final as shown in the Figure 5.4. In order to stay above the path of the leading aircraft, the pilot needs to know the altitude of the lead aircraft path at a point on the final approach that is nearly abeam the trailing aircraft. This information will help the pilot of the trailing aircraft to make the decision regarding when to turn onto the base leg.

Figure 5.4: Search algorithm implementation for downwind leg phase
For this situation, the spherical cone height/length is set at $\frac{1}{2}$ NM and the cone angle is set such that the base of the cone extends $\frac{1}{4}$ NM past the final approach course. This is estimated to give the pilot sufficient time to execute the turn onto the base leg while staying above the wake-generating aircraft’s flight path.

### 5.1.2 Base leg Phase

After turning base, the altitude of points on the leading aircraft flight path that are almost directly in front of the trailing aircraft pilot is needed. For this situation, as shown in Figure 5.5, the perpendicular distance from the apex to the base of the spherical cone is set such that it extends a quarter mile past the final approach path.

![Figure 5.5: Search algorithm implementation for base leg phase](image)

This will help the pilot of the trailing aircraft to maintain an altitude higher than that of the leading aircraft, which is on the straight in final approach. The cone angle is set such that the base of the cone spans $\frac{1}{4}$ NM of the final approach path.
5.1.3 Final approach

When the aircraft is on final approach, the pilot needs to know the altitude of the leading aircraft when the lead aircraft was just ahead of the current position of the trailing aircraft. During this phase of flight, the spherical cone is directed along the nose of the trailing aircraft. As shown in Figure 5.6, the cone height/length is 1/10 NM and the cone angle is 45 degrees.

![Figure 5.6: Search algorithm implementation for final approach](image)

The wide angle is set so as to capture the points of the wake-generating aircraft flight path despite the purposeful altitude separation. The short height/length is set so as only to capture the points near the current location of the trailing aircraft.

5.2 Mode Determination

The pilot could, in theory, input the runway information to allow the system to determine which mode to use. To ease pilot workload, a button or soft key could be used to toggle between the aforementioned three modes.
6. Simulation and Flight Testing for Performance Analysis

For the practical implementation of the proposed solution described in the previous chapter, a few realistic scenarios are considered that would be typical of a general aviation aircraft landing behind a large passenger aircraft. Although the Ohio University airport cannot support wide-body aircraft, it is used for simulation purposes for a couple of scenarios and was utilized as the base for the flight test that is described in Appendix B. Boston Logan International airport (KBOS airport) is used for simulation purposes for the aircraft arrival scenario on parallel runways.

The flight paths for the scenarios and the search algorithm used for computing the pre-defined portion of the flight path of a leading aircraft were simulated in Matlab® and are found in Appendix C.

6.1 Scenario 1:

An aircraft of heavy category (wake turbulence generator) joins the Runway 25 final approach course at an altitude of 3000 feet and 8 NM from the Ohio University airport-Snyder Field (KUNI airport). This aircraft will intercept the final approach fix (4 NM from threshold) at 3000 feet and fly the Instrument Landing System (ILS) (3.00 degree approach) all the way to touchdown. A GA aircraft will join a left downwind to land on Runway 25 at an altitude of 1500 feet.

Aircraft arrival scenario during the last 10 minutes approximately before touchdown is shown in Figure 6.1. This simulation shows time ‘T’, altitude information ‘A’ and wake-altitude information ‘WA’. The red path is the heavy aircraft flight path and the blue path
is the trailing aircraft flight path. This designation is followed for all the simulations described in this section. The altitude information of the trailing aircraft and its respective wake altitude information calculated using the search algorithm are displayed in the call out boxes.

Figure 6.1: Aircraft arrival scenario 1 - KUNI airport and the altitude strip for the upwind and downwind phases of flight.

In Figure 6.1, wake altitude information is shown when the aircraft is on the upwind and downwind phases of flight by considering two time epochs. When the aircraft is on the upwind phase i.e., when $T= 117$ sec, $WA$ is marked n/a (not applicable) as the points on the heading aircraft do not fall within the volume of the spherical cone. When the trailing aircraft is at $T= 390.5$sec i.e., on the downwind phase of flight, the heavy aircraft which was on the straight in final has already landed. Altitude of the trailing aircraft at this time epoch is 1740 feet and $WA = 701.71$ feet and the same are displayed on the red strip and
altitude strip in Figure 6.1. Start time and stop time for both heading and trailing aircrafts are shown in the call out box. Altitudes of the heading aircraft that fall in the spherical cone at this time epoch are also shown in the call out box and the highest altitude is considered to display on the altitude strip.

In Figure 6.2, wake altitude information is shown when the trailing aircraft is on the base leg phase of flight. At time epoch $T = 445.5$ sec, altitude of the trailing aircraft is 955.4 feet and the wake altitude $WA = 804.87$ feet. This information is also displayed on the wake strip and altitude strip.

In Figure 6.3, wake altitude information is shown when the trailing aircraft is on the final approach. At time epoch $T = 485.5$ sec, altitude of the trailing aircraft is 305.6 feet and the wake altitude $WA = 300.55$ feet.

With the help of this wake altitude information during every phase of flight, pilot can decide when to turn base and hence he/she can constantly stay above wake on the base and final approach.
Figure 6.2: Aircraft arrival scenario 1 - KUNI airport and the altitude strip for the base leg phases of flight.

Figure 6.3: Aircraft arrival scenario 1 - KUNI airport and the altitude strip when the aircraft is on final approach.
Vertical plot in Figure 6.4 gives altitude information of the trailing aircraft and the wake strip altitude for scenario 1. Whenever the points on the heading aircraft do not fall within the volume of the spherical cone wake strip altitude is zero (marked n/a in the above figures).

![Vertical plot for scenario 1 KUNI Airport](image)

**Figure 6.4: Vertical plot for scenario 1 - KUNI airport**

### 6.2 Scenario 2

In scenario 2, the heavy aircraft and the GA aircraft performs a straight in approach to runway 25. The heavy aircraft maintains an altitude of around 3000 feet until it has been cleared for landing and flies the ILS (3.00 degree approach) all the way to touchdown. The GA aircraft maintains an altitude of 2500 feet until it has been cleared for landing and starts its descent by flying the 3.00-degree approach until touchdown. The start time and stop time for the heading aircraft and trailing aircraft are shown in the Figure 6.5.
This figure also depicts the wake turbulence when the trailing aircraft is at the time epoch $T = 230$ sec.

**Figure 6.5: Aircraft arrival scenario 2 KUNI airport — Parallel straight in approaches**

In the call out box, the altitudes that fall in the volume of the spherical cone are listed and the one with the highest altitude is considered by the search algorithm to display on the wake altitude strip.

Figure 6.6 shows the vertical plot for scenario 2 plotted against time in seconds on X-axis and altitude in feet on Y-axis. Wake strip altitude starts approximately at $T = 115$ sec when the trailing aircraft is cleared for landing and flies 3.00 degree glide slope.
Figure 6.6: Vertical plot for scenario 2 - KUNI airport

Figure 6.7 shows enlarged image of the vertical plot to depict that the trailing aircraft is constantly maintaining high altitude with the help of wake strip altitude.
Figure 6.7: Enlarged image of vertical plot showing wake altitude and GA aircraft altitude

Figure 6.8 shows the enlarged image of the 3d plot of scenario2. The pilot in the GA aircraft constantly monitors the red tape in the altitude strip to stay at an altitude greater than that of the heavy aircraft and chooses a touchdown point ahead of the heavy aircraft to avoid the wake turbulence during this scenario. This can be clearly observed in this enlarged image of the 3d plot.
6.3 Scenario 3

In scenario 3, the heavy aircraft and the GA aircraft perform the straight in approach to final for parallel runways 22L and 22R in Boston/General Edward Lawrence Logan Intl (BOS) airport respectively. The runway 22L has the dimensions 10005 X 150 feet, which is generally used for heavy aircraft for arrivals and departures. The runway 22R has dimensions 7861 X 150 feet, used for the GA aircraft. The heavy aircraft maintains an altitude of around 3000 feet until it has been cleared for landing and flies the ILS (3.00 degree approach) all the way to touchdown. The GA aircraft maintains an altitude of 2500 feet until it has been cleared for landing and starts its descent by flying the 3.00-degree approach until touchdown. Figure 6.7 depicts the wake turbulence when the trailing aircraft is at the time epoch $T = 201$ sec. Figure 6.8 shows the vertical plot for this scenario with GA aircraft altitude and wake strip altitude.
Figure 6.7 Aircraft arrival scenario 3 for KBOS airport - parallel runways

Figure 6.8 Vertical plot for scenario 3 - KBOS airport

As discussed earlier in the previous section, during the case of parallel runways there is a chance for the wake to drift from one runway to the other if there is a crosswind of 3 to 5
knots. Hence, in this scenario the pilot in the GA aircraft constantly monitors the red tape in the altitude strip to stay at an altitude greater than that of the heavy aircraft, which has arrived in runway 22L and avoids wake turbulence.

### 6.4 Flight Testing:

Ohio University Computer Science graduate students Scott Nykl and Chad Mourning led the Flight-testing of the wake strip display [14-15]. Two GA aircraft were used to fly the wake turbulence scenarios. One aircraft flew the straight-in approach profile simulating the heavy aircraft and the other aircraft flew the GA profile. The architecture block diagrams used in these two aircraft are described in Appendix A. While actual ADS-B equipment was not used because of the cost (approximately $7000 per aircraft) and complexity, a simplified telemetry system was utilized to broadcast GPS position, altitude and speed of the "heavy aircraft" to the GA aircraft. The GA aircraft had a computer system onboard that processed position, altitude, roll and pitch information from an XSens attitude and heading reference system (AHRS) unit to feed the EFIS display mounted in front of the pilot. A telemetry receiver also onboard fed the computer system with position information from the "heavy aircraft" just as an ADS-B system would. The computer software, written by Nykl and Mourning, processed the data and generated the appropriate "red tape" area on the EFIS display that the pilot used for wake turbulence avoidance.
6.5 Scenario 4

In this scenario, the real flight data obtained on May 2\textsuperscript{nd}, 2011 at KUNI airport is taken to perform the simulation of search algorithm. Figure 6.9 shows the 2-D plot of GA aircraft flight path and Heavy aircraft flight path.

![Aircraft Arrival scenario - Real flight data, KUNI Airport](image)

**Figure 6.9** Aircraft arrival scenario 4 - May 2\textsuperscript{nd}, 2011 flight test data, KUNI airport

Figure 6.10 shows the vertical plot for the flight test data. This figure shows the GA aircrafts flight path altitude and wake strip altitude caused due to Heavy aircraft. In this plot, we can observe that the wake altitude drops to zero several times. This happens whenever the points on the heading aircraft do not fall within the volume of the spherical cone.
Figure 6.10 Vertical plot for scenario 4- May 2\textsuperscript{nd}, 2011 flight test data, KUNI airport

Figure 6.11 shows the enlarged image of the vertical plot. The flight test has been performed with varying altitudes constantly to test the altitude strip and the wake strip and this can be observed in this enlarged image. Trailing aircraft altitude is above the wake strip altitude for few instances and vice versa.

Figure 6.12 shows the vertical plot with the heavy aircraft flight path performed on May 2\textsuperscript{nd}, 2011.
Figure 6.11 Enlarged image of vertical plot for scenario 4

Figure 6.12 Vertical plot for scenario 4 – Heavy aircraft flight path
7. Conclusion and Future Work:

The flight-tested prototype successfully provided the pilot of the trailing aircraft with intuitive information to avoid wake turbulence with ease. Ohio University’s Avionics Engineering Center’s acting chief pilot Jamie Edwards described his experience with the prototype “The displayed wake strip information gives a pilot actual guidance to stay above the flight path of the aircraft ahead of him. It can give the pilot of the light aircraft in the situation a peace-of-mind never known before.” He further described the prototypes usage during flight, “The red wake strip concept is intuitive to fly. It overlays the altitude strip and gave me a direct indication as to my vertical position relative to the aircraft in front of me when he was at my same distance from the airport. It easily allowed me to set decent rate that kept the red wake strip below my current altitude and effectively clear of his wake turbulence” [15]. Although the wake strip display is beneficial primarily to pilots, the technology would also provide air traffic controllers with a greater level of comfort when they need to land a GA aircraft behind a heavy.

It is envisioned to make the system a feature in the “glass cockpits” of modern GA aircraft. These cockpits already have altitude strips in their primary flight displays and the addition of the wake altitude strip requires no additional display hardware. This system can be further developed to improvise the search algorithm by taking the input of runway information to determine which mode to use. Additionally, the spherical cone’s length and angles for each phase of flight need to be optimized. Future work should also be focused on testing the algorithm in airports that have multiple crossing and parallel runways and addressing high traffic congestion (i.e., handling more than one aircraft in
the vicinity). Finally, flight-tests with actual ADS-B equipment, at major airports with heavy aircraft need to be performed for validation.
References:

[1] National Transportation Safety Board (NTSB) Identification: SEA96LA173. Full narrative available:


[2011, June 1]


Appendix

Flight testing of the wake strip display was performed by Chad Mourning and Scott Nykl, Ph.D. students in Computer Science at Ohio University. This appendix summarizes the hardware and software that they put together and the next appendix summarizes the results of their testing.

A. Flight Test

The architecture developed for the preceding and trailing aircraft to perform the flight test and the components that are necessary for the installation are listed in the following sections.

A.1 Architecture:

The components are installed accordingly as shown in this architecture block diagram in the two aircraft, used for testing the prototype developed for avoiding wake turbulence. In the case of a trailing aircraft, AHRS, an altitude encoder and a telemetry receiver are connected to Laptop PC using three RS-232 data links via a USB hub. The AHRS unit was built near the center of gravity of the aircraft. The Yoke mounted LCD display is connected directly to the Laptop PC to visualize the wake strip and the altitude strip in the real time.
Figure A.1: Block Diagram for the equipment to be installed in the simulated ‘trailing’ aircraft (Baron)

For the leading aircraft, a GPS receiver, an altitude encoder and a telemetry transmitter are connected to Laptop PC using three RS-232 data links via a USB hub.

Figure A.2: Block diagram for the equipment to be installed in the simulated ‘heavy’ aircraft (Piper Saratoga)
A.2 Equipment:

1. Garmin’s GPS receiver

A GPSMAP 295 Garmin’s receiver is used in Piper Saratoga, which is pilot configurable. It has a 4-inch 16-color LCD display, which makes the pilot convenient to use its features in locating VOR’s (VHF Omni directional range), controlling agencies, vertical boundaries etc. It also has the feature to enable WAAS (Wide area augmentation system) for providing high accuracy and can provide position accuracy up to <3m.

This GPS unit has dimensions 8.0x17.3x6.5 cm and weighs 1.4lbs. It has an operating voltage of 10-35v. The position accuracy is 15m RMS and velocity accuracy is 0.1knot RMS steady state with an update rate of 1Hz[12].

2. Attitude and Heading Reference System

The AHRS (Attitude and heading reference system) unit is used in the trailing aircraft to calculate roll and pitch information. The model XSENS MTI-G1 unit has an integrated GPS receiver along with the accelerometer, gyroscope, static pressure sensor and temperature sensor. It provides 3D orientation, 3D acceleration, 3D position and velocity data.

The AHRS unit has dimensions 58x58x53 mm and weighs 68g. It has a wide operating voltage of 5-30v with a very low power consumption ranging from 610mW – 690mW [13].
3. Telemetry Transmitter

The telemetry transmitter used in the leading aircraft has a baud rate of 1200 with a frequency of 329MHz transmitted the pseudo ADS-B message (Latitude, Longitude and altitude information from the GPSMAP 295 Garmin’s receiver) to the trailing aircraft at 5Hz and GPS update frequency was 1Hz. The GPS receiver transmitted its 3D location and accuracy data in $GPGGA$ (Global positioning system fix data) format which is an NMEA (National Marine Electronics Association) sentence with an update frequency of 1Hz.

A.3 Aircraft

The Avionics Engineering center’s 1980 Piper Saratoga and Beach Baron 58 have been used as a leading and trailing aircraft respectively, to perform the flight tests. The Piper Saratoga (N8238C) runs on a single engine Lycoming IO-540-K1G5 (300Hp), low-wing aircraft that has seating capacity of up to 6 occupants. This aircraft can carry up to 1384lbs apart from its own empty weight of 2216lbs.
The Beechcraft Baron 58 aircraft is a twin-engine, light–medium aircraft. It runs on a continental engine IO 520 (330Hp). This aircraft also has a seating capacity of up to 6 occupants and can carry up to 5500lbs including its own empty weight.
These aircraft served as an excellent research platform due to their large cabin space. The equipment that was installed in these aircraft was placed on a rack behind the seat of the safety pilot.

**A.4 Equipment Installation:**

The equipment mentioned above is installed in both aircraft on a 17” rack behind the pilot seat by taking necessary measures such that it stays firm. A laptop computer is placed in the trailing aircraft as shown in the Figure 6 to monitor the data transmission and acquisition.

![Figure A.5: Equipment installed in the trailing aircraft – Baron](image)
Figure A.6: Equipment installed in the leading aircraft – Piper Saratoga

B. Flight Test Results

The prototype designed to avoid wake vortices was tested in May 2011 at Ohio University Synder field airport using the Piper Saratoga as the leading aircraft and Beach Baron 58 as a trailing aircraft.

Figure B.1: Multi function display installed in front of the pilot to visualize the real time virtual world [15].
As described in the proposed solution the wake generating aircraft (heavy aircraft), here in this case a Piper Saratoga joined the Runway 25 final approach course at an altitude of 3000 feet and 10 NM. A GA aircraft, in this case a Beach Baron 58 will be circled at a distance of 5-7 NM and performs a base leg maneuver to join a left downwind on to Runway 25 at an altitude of 1500 feet.

![Figure B.2: Screen shots showing the Trailing aircraft joining the final approach of the leading aircraft and the wake strip [15].](image)

Initially, when the trailing aircraft has approached final after performing the maneuver, it was observed that the entire altitude strip was red as the trailing aircraft is 2000ft below the heavy aircraft. At 4 NM from threshold, the leading aircraft has started descending and the red tape area has progressively lowered. At this point when the GA aircraft pilot is above the altitude of the leading aircraft, started descent aiming for a touchdown point beyond the touchdown point of the leading aircraft and has successfully landed avoiding the wake turbulence. This system never lost communication in the entire flight test and it worked perfectly even at the maximum tested range of about 13 NM. For a brief you tube video of actual flight test follow the link given in [14][15].
C. Software Listings (MATLAB)

C.1 Aircraft Arrival Scenario 1 – KUNI airport

%%Scenario 1
%%Aircraft Arrival profile 1 – KUNI airport
%%Author – Nikhil Tej Gandhi
%%Ohio University
%%10/1/2012

%%%%%%%%%%For the Heading aircraft flight
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
cclc
close all
clear all
lat1 = 39+13/60;
lon1 = 82+14/60;
pi = 3.1415;
lat1_radians = 39.2118*(pi/180);
lon1_radians = -82.2292*(pi/180);
distance_miles= 3.522;
dist_decr = 0.0222; %for every 135 feet at 0.5 seconds when aircraft is travelling at 160kts( 1nm = 6076 feet)
j=1;
runway_info = 247; %(in degrees)
distance = 12.5;
for j=1:1:57 %121 samples
distance = distance-0.051; %when travelling at 160knots/sec
R = 6371;
d= distance*1.85200;% in km
theta = runway_info*(pi/180);
lat3(j,1)= asin(sin(lat1_radians)*cos(d/R) +
cos(lat1_radians)*sin(d/R)*cos(theta));
lon3(j,1)= lon1_radians +
atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-
sin(lat1_radians)*sin(lat3(j,1)));
lat3(j,1)=lat3(j,1)*(180/pi);
lon3(j,1)=lon3(j,1)*(180/pi);
altitude2(j,1) = 3084;
time_stamp1(j,1) = j/2;
end

distance = 9.6;
for j=58:1:159 %121 samples
distance = distance-0.051; %when travelling at 160knots/sec
R = 6371;
d= distance*1.85200;% in km
theta = runway_info*(pi/180);
lat3(j,1)= asin(sin(lat1_radians)*cos(d/R) +
cos(lat1_radians)*sin(d/R)*cos(theta));
lon3(j,1) = lon1_radians + 
atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-
  sin(lat1_radians)*sin(lat3(j,1)));
lat3(j,1)=lat3(j,1)*(180/pi);
lon3(j,1)=lon3(j,1)*(180/pi);
altitude2(j,1) = (0.0524*distance)*6076; %tan(3)*dist, in feet
time_stamp1(j,1) = j/2;
end

distance = 4.4;
for j=160:1:403 %121 samples
  distance = distance-0.018; %when travelling at 130knots
  R = 6371;
  d= distance*1.85200; % in km
  theta = runway_info*(pi/180);
  lat3(j,1)= asin(sin(lat1_radians)*cos(d/R) +
    cos(lat1_radians)*sin(d/R)*cos(theta));
  lon3(j,1)= lon1_radians +
  atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-
    sin(lat1_radians)*sin(lat3(j,1)));
  lat3(j,1)=lat3(j,1)*(180/pi);
  lon3(j,1)=lon3(j,1)*(180/pi);
  altitude2(j,1) = (0.0524*distance)*6076; %tan(3)*dist, in feet
  time_stamp1(j,1) = j/2;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%For the Trailing aircraft flight path%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%% 1

lat1 = 39+13/60;
lon1 = 82+14/60;
pi = 3.1415;
lat1_radians = 39.2118*(pi/180);
lon1_radians = -82.2290*(pi/180);
distance_miles= 3.522;
dist_decr = 0.0222; %for every 135 feet at 0.5 seconds when aircraft is
  travelling at 160kts( 1nm = 6076 feet)
i=1;
runway_info = 247; %(in degrees)

%%%% % 1
for i=1:1:90 % 44 seconds
distance_miles=distance_miles-dist_decr;
dist_km = distance_miles*1.85200; % in km
R = 6371;
theta = (runway_info-45)*(pi/180);
theta_calc(i,1) = theta*(180/pi); % for observation in degrees
d = -(dist_km)/sin(theta);
dist(i,1) = d/1.85200; % for observation
lat2(i,1)= asin(sin(lat1_radians)*cos(d/R) +
 \cos(lat1_radians)*sin(d/R)*cos(theta));
lon2(i,1)= lon1_radians +
atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-
\sin(lat1_radians)*sin(lat2(i,1)));
altitude(i,1) = 2400;
lat2(i,1)=lat2(i,1)*(180/pi); %converting to degrees
lon2(i,1)=lon2(i,1)*(180/pi);
time_stamp(i,1) = i/2;
end

%%%%% 1 For the curvature of the
%%%%% aircraft%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
x1=1.54;
y1=-1.54;
x2=1.5;
y2=-1.46;
r = 0.8;
d = sqrt((x2-x1)^2+(y2-y1)^2); % Distance between points
a = atan2(-(x2-x1),y2-y1); % Perpendicular bisector angle
b = asin(d/2/r); % Half arc angle
c = linspace(a-b,a+b,28); % Arc angle range
e = sqrt(r^2-d^2/4); % Distance, center to midpoint
x = (x1+x2)/2-e*cos(a)+r*cos(c); % Cartesian coords. of arc
y = (y1+y2)/2-e*sin(a)+r*sin(c);

x1= 0;
y1 =0;
theta_up = theta_calc(i,1);
for i =90:1:117
    dist(i,1) = sqrt((x(i-89)-x1)^2+(y(i-89)-y1)^2);
d = dist(i,1)*1.8520;
theta = -asin(y(i-89)/dist(i,1))+(theta_up-45)*pi/180;
theta_calc(i,1) = theta*180/pi;
lat2(i,1)= asin(sin(lat1_radians)*cos(d/R) +
\cos(lat1_radians)*sin(d/R)*cos(theta));
lon2(i,1)= lon1_radians +
atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-
\sin(lat1_radians)*sin(lat2(i,1)));
lat2(i,1)=lat2(i,1)*(180/pi);
lon2(i,1)=lon2(i,1)*(180/pi);
altitude(i,1) = 2400;
time_stamp(i,1) = i/2;
end

distance = 1.44;
for i=118:1:182
    distance = distance-0.0222;
dist(i,1) = sqrt(1.5^2+distance^2); % for observation
R = 6371;
theta = runway_info*(pi/180)-asin(1.5/ dist(i,1));
theta_calc(i,1) = theta*(180/pi); % for observation in degrees
d= dist(i,1)*1.85200; % in km
lat2(i,1) = asin(sin(lat1_radians)*cos(d/R) +
cos(lat1_radians)*sin(d/R)*cos(theta));
lon2(i,1) = lon1_radians +
atan2(sin(theta)*sin(d/R)*cos(lat1_radians),
  cos(d/R) -
  sin(lat1_radians)*sin(lat2(i,1)));
lat2(i,1) = lat2(i,1)*(180/pi);
lon2(i,1) = lon2(i,1)*(180/pi);
altitude(i,1) = 2400;
time_stamp(i,1) = i/2;
end

%%%%% 3
theta_up = theta_calc(i,1);
distance = 0;
for i=183:1:349 %166 samples
distance = distance+0.0222;
dist(i,1) = sqrt((x(i-349)-x1)^2+(y(i-349)-y1)^2); % for observation
R = 6371;
theta = theta_up*(pi/180) - asin(distance/ dist(i,1));
theta_calc(i,1) = theta*(180/pi); % for observation in degrees
d = dist(i,1)*1.85200; % for observation
lat2(i,1) = asin(sin(lat1_radians)*cos(d/R) +
  cos(lat1_radians)*sin(d/R)*cos(theta));
lon2(i,1) = lon1_radians +
atan2(sin(theta)*sin(d/R)*cos(lat1_radians),
  cos(d/R) -
  sin(lat1_radians)*sin(lat2(i,1)));
lat2(i,1) = lat2(i,1)*(180/pi);
lon2(i,1) = lon2(i,1)*(180/pi);
altitude(i,1) = 2400;
time_stamp(i,1) = i/2;
end

%%%%% 2 For the curvature of the
% aircraft
x1=1.5;
y1=3.7;
x2=0.8;
y2=4.5;
r = 0.8;
d = sqrt((x2-x1)^2+(y2-y1)^2); % Distance between points
a = atan2(-(x2-x1),y2-y1); % Perpendicular bisector angle
b = asin(d/2/r); % Half arc angle
c = linspace(a-b,a+b,56); % Arc angle range
e = sqrt(r^2-d^2/4); % Distance, center to midpoint
x = (x1+x2)/2-e*cos(a)+r*cos(c); % Cartesian coords. of arc
y = (y1+y2)/2-e*sin(a)+r*sin(c);

x1 = 0;
y1 = 0;
for i =350:1:405 %56 samples considered 55
dist(i,1) = sqrt((x(i-349)-x1)^2+(y(i-349)-y1)^2);
d = dist(i,1)*1.85200;
theta_calc(i,1) = (theta_up*(pi/180) -
  asin(y(i-349)/dist(i,1)))*180/pi;
theta = theta_calc(i,1) *pi/180;
l= asin(sin(lat1_radians)*cos(d/R) +
cos(lat1_radians)*sin(d/R)*cos(theta));
lon2(i,1)= lon1_radians +
atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R) -
sin(lat1_radians)*sin(lat2(i,1)));
l= lat2(i,1)*180/pi;
lon2(i,1)=lon2(i,1)*(180/pi);
altitude(i,1) = 2400;
time_stamp(i,1) = i/2;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
distance = 0.7;
for i=406:1:436 %30 samples
distance = distance-0.0222;
dist(i,1) = sqrt(4.5^2+(distance)^2); % for observation
R = 6371;
theta = theta_up*(pi/180)-asin(4.5/ dist(i,1));
theta_calc(i,1) = theta*(180/pi); % for observation in degrees
d= dist(i,1)*1.85200; % in km
lat2(i,1)= asin(sin(lat1_radians)*cos(d/R) +
cos(lat1_radians)*sin(d/R)*cos(theta));
lon2(i,1)= lon1_radians +
atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R) -
sin(lat1_radians)*sin(lat2(i,1)));
l= lat2(i,1)*180/pi;
lon2(i,1)=lon2(i,1)*(180/pi);
altitude(i,1) = 2400;
time_stamp(i,1) = i/2;
end
theta_up= theta_calc(i,1);
distance = 0;
for i=437:1:467 %30 samples
distance = distance+0.0222;
dist(i,1) = sqrt(4.5^2+(distance)^2); % for observation
R = 6371;
theta = theta_up*(pi/180)-asin(distance/ dist(i,1));
theta_calc(i,1) = theta*(180/pi); % for observation in degrees
d= dist(i,1)*1.85200; % in km
lat2(i,1)= asin(sin(lat1_radians)*cos(d/R) +
cos(lat1_radians)*sin(d/R)*cos(theta));
lon2(i,1)= lon1_radians +
atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R) -
sin(lat1_radians)*sin(lat2(i,1)));
l= lat2(i,1)*180/pi;
lon2(i,1)=lon2(i,1)*(180/pi);
altitude(i,1) = 2400;
time_stamp(i,1) = i/2;
end
%%% 3 For the curvature of the aircraft

\[ x_1 = -0.7; \]
\[ y_1 = 4.5; \]
\[ x_2 = -1.5; \]
\[ y_2 = 3.7; \]
\[ r = 0.8; \]
\[ d = \sqrt{(x_2-x_1)^2+(y_2-y_1)^2}; \] % Distance between points
\[ a = \arctan2(-(x_2-x_1),y_2-y_1); \] % Perpendicular bisector angle
\[ b = \arcsin(d/2/r); \] % Half arc angle
\[ c = \text{linspace}(a-b,a+b,56); \] % Arc angle range
\[ e = \sqrt{r^2-d^2/4}; \] % Distance, center to midpoint
\[ x = (x_1+x_2)/2-e\cos(a)+r\cos(c); \] % Cartesian coords. of arc
\[ y = (y_1+y_2)/2-e\sin(a)+r\sin(c); \]

\[ x_1 = 0; \]
\[ y_1 = 0; \]
\[ \text{for} \ i = 468:1:523 \]
\[ \text{dist}(i,1) = \sqrt{(x(i-467)-x_1)^2+(y(i-467)-y_1)^2}; \]
\[ d = \text{dist}(i,1)*1.8520; \]
\[ \text{theta} = \text{theta}_{\text{up}}*(\pi/180)+\arcsin(x(i-467)/\text{dist}(i,1)); \]
\[ \text{theta}_{\text{calc}}(i,1) = \text{theta}*180/\pi; \]
\[ \text{lat2}(i,1) = \arcsin(\sin(\text{lat1}_{\text{radians}})*\cos(d/R) + \cos(\text{lat1}_{\text{radians}})*\sin(d/R)*\cos(\text{theta})); \]
\[ \text{lon2}(i,1) = \text{lon1}_{\text{radians}} + \arctan2(\sin(\text{theta})*\sin(d/R)*\cos(\text{lat1}_{\text{radians}}), \cos(d/R)-\sin(\text{lat1}_{\text{radians}})*\sin(\text{lat2}(i,1)))); \]
\[ \text{alt}(i,1) = 2400; \]
\[ \text{time}_{\text{stamp}}(i,1) = i/2; \]
\[ \text{end} \]

\[ \text{distance} = 3.7; \]
\[ \text{for} \ i = 524:1:689 \% 165 samples \]
\[ \text{distance} = \text{distance}-0.0222; \]
\[ \text{dist}(i,1) = \sqrt{1.5^2+\text{distance}^2}; \] % for observation
\[ R = 6371; \]
\[ \text{if} \ i>666 \]
\[ \text{theta} = 360*(\pi/180)+\arcsin(\text{distance}/\text{dist}(i,1))-23*(\pi/180); \]
\[ \text{else} \]
\[ \text{theta} = \arcsin(\text{distance}/\text{dist}(i,1))-23*(\pi/180); \]
\[ \text{end} \]
\[ \text{theta}_{\text{calc}}(i,1) = \text{theta}*(180/\pi); \] % for observation in degrees
\[ d = \text{dist}(i,1)*1.85200; \] % for observation
\[ \text{lat2}(i,1) = \arcsin(\sin(\text{lat1}_{\text{radians}})*\cos(d/R) + \cos(\text{lat1}_{\text{radians}})*\sin(d/R)*\cos(\text{theta})); \]
\[ \text{lon2}(i,1) = \text{lon1}_{\text{radians}} + \arctan2(\sin(\text{theta})*\sin(d/R)*\cos(\text{lat1}_{\text{radians}}), \cos(d/R)-\sin(\text{lat1}_{\text{radians}})*\sin(\text{lat2}(i,1)))); \]
\[
\text{lat2}(i, 1) = \text{lat2}(i, 1) \times \frac{180}{\pi}; \\
\text{lon2}(i, 1) = \text{lon2}(i, 1) \times \frac{180}{\pi}; \\
\text{if } i \geq 689 \\
\text{altitude}(i, 1) = (0.0524 \times (7.5 + \text{distance})) \times 6076; \quad \% \tan(3) \times \text{dist, in feet} \\
\text{else} \\
\text{altitude}(i, 1) = 2400; \\
\text{end} \\
\text{time\_stamp}(i, 1) = i/2; \\
\text{end} \\
\text{for } i=1:1:689 \\
\text{max\_altitude}(i, 1) = 0; \\
\text{end} \\
\%	ext{Altitude starts descent from here. Aircraft on downwind.} \\
\text{theta\_up} = \text{theta\_calc}(i, 1); \\
\text{distance} = 0.0148; \\
\text{for } i=690:1:787 \quad \% 97 samples \\
\text{distance} = \text{distance} + 0.0222; \\
\text{dist}(i, 1) = \sqrt{1.5^2 + \text{distance}^2}; \quad \% \text{for observation} \\
R = 6371; \\
\text{theta} = \text{theta\_up} \times (\pi/180) - \arccos(1.5/\text{dist}(i, 1)); \\
\text{theta\_calc}(i, 1) = \text{theta} \times (180/\pi); \quad \% \text{for observation in degrees} \\
d = \text{dist}(i, 1) \times 1.85200; \quad \% \text{in km} \\
\text{lat2}(i, 1) = \arcsin(\sin(\text{lat1\_radians}) \times \cos(d/R) + \cos(\text{lat1\_radians}) \times \sin(d/R) \times \cos(\text{theta})); \\
\text{lon2}(i, 1) = \text{lon1\_radians} + \arctan2(\sin(\text{theta}) \times \sin(d/R) \times \cos(\text{lat1\_radians}), \cos(d/R) - \sin(\text{lat1\_radians}) \times \sin(\text{lat2}(i, 1))); \\
\text{lat2}(i, 1) = \text{lat2}(i, 1) \times (180/\pi); \\
\text{lon2}(i, 1) = \text{lon2}(i, 1) \times (180/\pi); \\
\text{initial\_point} = \{664700, -4904400, 4012400\}; \\
\text{altitude}(i, 1) = (0.0524 \times (7.5 - \text{distance})) \times 6076; \quad \% \tan(3) \times \text{dist, in feet} \\
\text{dist\_radius} = 3000; \quad \% \text{In feet} \\
\text{theta\_cone} = 80; \\
\text{max\_altitude}(i, 1) = \text{search\_algorithm\_function}(i, \text{lat2}, \text{lon2}, \text{lat3}, \text{lon3}, \text{altitude2}, \text{altitude}, \text{initial\_point}(1,:), \text{dist\_radius}, \text{theta\_cone}) ; \\
\text{time\_stamp}(i, 1) = i/2; \\
\text{end} \\
\%	ext{ Aircraft on its base leg and} \\
\%	ext{Final} \\
x_1 = -1.5; \\
y_1 = -2.2; \\
x_2 = 0; \\
y_2 = -2.2; \\
r = 0.8; \\
d = \sqrt{(x_2-x_1)^2 + (y_2-y_1)^2}; \quad \% \text{Distance between points} \\
a = \arctan2(-(x_2-x_1), y_2-y_1); \quad \% \text{Perpendicular bisector angle} \\
b = \arcsin(d/2/r); \quad \% \text{Half arc angle} \\
c = \text{linspace}(a-b, a+b, 112); \quad \% \text{Arc angle range} \\
e = \sqrt{r^2 - d^2/4}; \quad \% \text{Distance, center to midpoint} \\
x = (x_1+x_2)/2 - e \times \cos(a) + r \times \cos(c); \quad \% \text{Cartesian coords. of arc} \\
y = (y_1+y_2)/2 - e \times \sin(a) + r \times \sin(c);
x1 = 0;
y1 = 0;
distance = 2.1904;

for i = 788:1:899 %112 samples
    distance = distance + 0.0222;
    dist(i,1) = sqrt((x(i-787)-x1)^2+(y(i-787)-y1)^2);
    d = dist(i,1)*1.8520;
    theta = runway_info*(pi/180)-asin(x(i-787)/dist(i,1));
    theta_calc(i,1) = theta*180/pi;
    lat2(i,1) = asin(sin(lat1_radians)*cos(d/R) +
        cos(lat1_radians)*sin(d/R)*cos(theta));
    lon2(i,1) = lon1_radians +
        atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-
        sin(lat1_radians)*sin(lat2(i,1)));)
lat2(i,1) = lat2(i,1)*(180/pi);
lon2(i,1) = lon2(i,1)*(180/pi);
d = distance*1.85200; % in km
theta = runway_info*(pi/180);
lat2(i,1) = asin(sin(lat1_radians)*cos(d/R) +
    cos(lat1_radians)*sin(d/R)*cos(theta));
lon2(i,1) = lon1_radians +
    atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-
    sin(lat1_radians)*sin(lat2(i,1)));)
if i <= 930
    altitude(i,1) = altitude(i-1,1) - 10;
else
    altitude(i,1) = (0.0524*distance)*6076; %tan(3)*dist, in feet
end

initial_point = [664600,-4904400,4012300];
dist_radius = 4000; %In feet
theta_cone = 70;
max_altitude(i,1) = search_algorithm_function(i,lat2,lon2,
lat3,lon3,altitude2,altitude,initial_point(1,:),dist_radius,theta_cone);

end

%%%%%%%%%%%%%%%%Aircraft on Final on 3deg glide
slopes%%%%%%%%%%%%%%%%%%%%%%%%%%%%

distance = 2.22;
for i = 900:1:1024 %121 samples
    distance = distance - 0.018; %when travelling at 130knots
    R = 6371;
    d = distance*1.85200; % in km
    theta = runway_info*(pi/180);
    lat2(i,1) = asin(sin(lat1_radians)*cos(d/R) +
        cos(lat1_radians)*sin(d/R)*cos(theta));
    lon2(i,1) = lon1_radians +
        atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-
        sin(lat1_radians)*sin(lat2(i,1)));)
lat2(i,1) = lat2(i,1)*(180/pi);
lon2(i,1) = lon2(i,1)*(180/pi);
if i <= 930
    altitude(i,1) = altitude(i-1,1) - 10;
else
    altitude(i,1) = (0.0524*distance)*6076; %tan(3)*dist, in feet
end

initial_point = [665800,-4905300,4009800];
dist_radius = 528; %1/10th mile in feet
theta_cone = 90;
max_altitude(i,1) = search_algorithm_function(i,lat2,lon2,
lat3,lon3,altitude2,altitude,initial_point(1,:),dist_radius,theta_cone);
time_stamp(i,1) = i/2;
end

time_stamp1 = sort(time_stamp1,'ascend');
time_stamp = sort(time_stamp,'ascend');
time_stamp1 = time_stamp1+100;

figure,
plot3(lon2,lat2,altitude,lon3,lat3,altitude2,'r','LineWidth',2);
xlabel('Longitude in degrees');
ylabel('Latitude in degrees');
zlabel('Altitude in feet');
legend('GA aircraft flight path','Heavy aircraft flight path');
title('Aircraft Arrival scenario 1 KUNI Airport 3 Dimensional');
grid on;

figure,
plot(lon2,lat2,lon3,lat3,'r','LineWidth',2);
xlabel('Longitude in degrees');
ylabel('Latitude in degrees');
legend('GA aircraft flight path','Heavy aircraft flight path');
title('Aircraft Arrival scenario 1 KUNI Airport');
grid on;

figure,
plot(time_stamp,altitude,time_stamp,max_altitude,'r','LineWidth',2);
xlabel('Time in Sec');
ylabel('Altitude in feet');
legend('GA aircraft flight path','Wake strip altitude');
title('Vertical plot for scenario 1 KUNI Airport');
grid on;
C.2 Aircraft Arrival Scenario 2 – KUNI airport

```matlab
%%Scenario 2
%%Aircraft Arrival profile 2 – KUNI airport
%%Author - Nikhil Tej Gandhi
%%Ohio University
%%10/1/2012

%%%%%%%%%%%%%%%%%Heavy Aircraft flight path%%%%%%%%%%%%%%%%%%%%%%%%%
clc
close all
clear all

lat1 = 39+13/60;
lon1 = 82+14/60;
pi = 3.1415;
lat1_radians = 39.2118*(pi/180);
lon1_radians = -82.2292*(pi/180);
distance_miles = 3.522;
dist_decr = 0.0222; %for every 135 feet at 0.5 seconds when aircraft is travelling at 160kts (1nm = 6076 feet)
j=1;
runway_info = 247; %in degrees

distance = 12.5;
for j=1:1:57 %121 samples
    distance = distance-0.051; %when travelling at 160knots/sec
    R = 6371;
    d= distance*1.85200; % in km
    theta = runway_info*(pi/180);
    lat3(j,1) = asin(sin(lat1_radians)*cos(d/R) + cos(lat1_radians)*sin(d/R)*cos(theta));
    lon3(j,1) = lon1_radians + atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-sin(lat1_radians)*sin(lat3(j,1)));
    lat3(j,1)=lat3(j,1)*(180/pi);
    lon3(j,1)=lon3(j,1)*(180/pi);
    altitude2(j,1) = 3084;
    time_stamp1(j,1) = j/2;
end

distance = 9.6;
for j=58:1:159 %121 samples
    distance = distance-0.051; %when travelling at 160knots/sec
    R = 6371;
    d= distance*1.85200; % in km
    theta = runway_info*(pi/180);
    lat3(j,1) = asin(sin(lat1_radians)*cos(d/R) + cos(lat1_radians)*sin(d/R)*cos(theta));
    lon3(j,1) = lon1_radians + atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-sin(lat1_radians)*sin(lat3(j,1)));
    lat3(j,1)=lat3(j,1)*(180/pi);
    lon3(j,1)=lon3(j,1)*(180/pi);
```
distance = 4.4;
for j=160:1:403 %121 samples
    distance = distance-0.018; %when travelling at 130 knots
    R = 6371;
    d= distance*1.85200; % in km
    theta = runway_info*(pi/180);
    lat3(j,1)= asin(sin(lat1_radians)*cos(d/R) + cos(lat1_radians)*sin(d/R)*cos(theta));
    lon3(j,1)= lon1_radians + atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-sin(lat1_radians)*sin(lat3(j,1))));
    lat3(j,1)=lat3(j,1)*(180/pi);
    lon3(j,1)=lon3(j,1)*(180/pi);
    altitude2(j,1) = (0.0524*distance)*6076; %tan(3)*dist, in feet
    time_stamp1(j,1) = j/2;
end

%%%%%%%%%%%%%%%%%Trailing Aircraft flight path%%%%%%%%%%%%%%%%%%%%

lat1 = 39+13/60;
lon1 = 82+14/60;
pi = 3.1415;
lat1_radians = 39.2118*(pi/180);
lon1_radians = -82.2290*(pi/180);
distance_miles= 3.522;

dist_decr = 0.0222; %for every 135 feet at 0.5 seconds when aircraft is travelling at 160kts (1nm = 6076 feet)
i=1;
runway_info = 247; % (in degrees)

distance = 10.5;
for i=1:1:45 %121 samples
    distance = distance-0.051; %when travelling at 160 knots/sec
    R = 6371;
    d= distance*1.85200; % in km
    theta = runway_info*(pi/180);
    lat2(i,1)= asin(sin(lat1_radians)*cos(d/R) + cos(lat1_radians)*sin(d/R)*cos(theta));
    lon2(i,1)= lon1_radians + atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-sin(lat1_radians)*sin(lat2(i,1))));
    lat2(i,1)=lat2(i,1)*(180/pi);
    lon2(i,1)=lon2(i,1)*(180/pi);
    altitude(i,1) = 2500;
    initial_point = [652400,-4912200,4006200];
    dist_radius = 2000; %In feet
    theta_cone = 90;
    max_altitude(i,1)= search_algorithm_function(i,lat2,lon2,
lat3,lon3,altitude2,altitude,initial_point(1,:),dist_radius,theta_cone)
time_stamp(i,1) = i/2;
end

distance = 7.725;
for i=46:1:105 %121 samples
    distance = distance-0.051; %when travelling at 160knots/sec
    R = 6371;
    d= distance*1.85200; % in km
    theta = runway_info*(pi/180);
    lat2(i,1)= asin(sin(lat1_radians)*cos(d/R) +
    cos(lat1_radians)*sin(d/R)*cos(theta));
    lon2(i,1)= lon1_radians +
    atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-
    sin(lat1_radians)*sin(lat2(i,1)));
    lat2(i,1)=lat2(i,1)*(180/pi);
    lon2(i,1)=lon2(i,1)*(180/pi);
    altitude(i,1) = (0.0524*distance)*6076; %tan(3)*dist, in feet
    initial_point = [656200,-4910700,4007500];
    dist_radius = 2000; %In feet
    theta_cone = 90;
    max_altitude(i,1)= search_algorithm_function(i,lat2,lon2,
    lat3,lon3,altitude2,altitude,initial_point(1,:),dist_radius,theta_cone)
end
    time_stamp(i,1) = i/2;
end

distance = 4.665;
for i=106:1:364 %121 samples
    distance = distance-0.018; %when travelling at 130knots
    R = 6371;
    d= distance*1.85200; % in km
    theta = runway_info*(pi/180);
    lat2(i,1)= asin(sin(lat1_radians)*cos(d/R) +
    cos(lat1_radians)*sin(d/R)*cos(theta));
    lon2(i,1)= lon1_radians +
    atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-
    sin(lat1_radians)*sin(lat2(i,1)));
    lat2(i,1)=lat2(i,1)*(180/pi);
    lon2(i,1)=lon2(i,1)*(180/pi);
    altitude(i,1) = (0.0524*distance)*6076; %tan(3)*dist, in feet
    initial_point = [656200,-4910700,4007500];
    dist_radius = 528; %1/10th mile in feet
    theta_cone = 90;
    max_altitude(i,1)= search_algorithm_function(i,lat2,lon2,
    lat3,lon3,altitude2,altitude,initial_point(1,:),dist_radius,theta_cone)
end
    time_stamp(i,1) = i/2;
end

    time_stamp1 = sort(time_stamp1,'ascend');
    time_stamp = sort(time_stamp,'ascend');
time_stamp = time_stamp+100;

figure,
plot3(lon2, lat2, altitude, lon3, lat3, altitude2, 'r', 'LineWidth', 2);
xlabel('Longitude in degrees');
ylabel('Latitude in degrees');
zlabel('Altitude in feet');
legend('GA aircraft flight path', 'Heavy aircraft flight path');
title('Aircraft Arrival Profile 2 KUNI Airport 3 Dimensional');
grid on;

figure,
plot(lon2, lat2, lon3, lat3, 'r', 'LineWidth', 2);
xlabel('Longitude in degrees');
ylabel('Latitude in degrees');
legend('GA aircraft flight path', 'Heavy aircraft flight path');
title('Aircraft Arrival Profile 2 KUNI Airport');
grid on;

figure,
plot(time_stamp, altitude, time_stamp, max_altitude, 'r', 'LineWidth', 2);
xlabel('Time in Sec');
ylabel('Altitude in feet');
legend('GA aircraft flight path', 'Wake strip altitude');
title('Vertical plot for scenario 2 KUNI Airport');
grid on;
C.3 Aircraft Arrival Scenario 3 – Parallel Runways – BOS airport

%%Scenario 3
%%Aircraft Arrival profile 3 – KBOS airport
%%Author – Nikhil Tej Gandhi
%%Ohio University
%%10/1/2012

%%%%Heavy Aircraft flight path%%%%%%%%%%%%%%%%%%%%%
clear all
pi = 3.1415;
lat1_radians = 42.36684*(pi/180);
lon1_radians = -70.9836*(pi/180);
distance_miles = 3.522;
dist_decr = 0.0222; % for every 135 feet at 0.5 seconds when aircraft is travelling at 160kts (1nm = 6076 feet)
j=1;
runway_info = 216; % (in degrees)

distance = 12.5;
for j=1:1:57 % 121 samples
    distance = distance-0.051; % when travelling at 160knots/sec
    R = 6371;
    d= distance*1.85200; % in km
    theta = runway_info*(pi/180);
    lat3(j,1)= asin(sin(lat1_radians)*cos(d/R) + cos(lat1_radians)*sin(d/R)*cos(theta));
    lon3(j,1)= lon1_radians + atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-sin(lat1_radians)*sin(lat3(j,1)));
    lat3(j,1)=lat3(j,1)*(180/pi);
    lon3(j,1)=lon3(j,1)*(180/pi);
    altitude2(j,1) = 3084;
    time_stamp1(j,1) = j/2;
end

distance = 9.6;
for j=58:1:159 % 121 samples
    distance = distance-0.051; % when travelling at 160knots/sec
    R = 6371;
    d= distance*1.85200; % in km
    theta = runway_info*(pi/180);
    lat3(j,1)= asin(sin(lat1_radians)*cos(d/R) + cos(lat1_radians)*sin(d/R)*cos(theta));
    lon3(j,1)= lon1_radians + atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-sin(lat1_radians)*sin(lat3(j,1)));
    lat3(j,1)=lat3(j,1)*(180/pi);
    lon3(j,1)=lon3(j,1)*(180/pi);
    altitude2(j,1) = (0.0524*distance)*6076; % tan(3)*dist, in feet
time_stamp1(j,1) = j/2;

end

distance = 4.4;
for j=160:1:403  %121 samples
    distance = distance-0.018;  %when travelling at 130knots
    R = 6371;
    d = distance*1.85200;  % in km
    theta = runway_info*(pi/180);
    lat3(j,1)= asin(sin(lat1_radians)*cos(d/R) +
                      cos(lati_radians)*sin(d/R)*cos(theta));
    lon3(j,1)= lon1_radians +
                      atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-
                      sin(lati_radians)*sin(lat3(j,1)));
    lat3(j,1)=lat3(j,1)*(180/pi);
    lon3(j,1)=lon3(j,1)*(180/pi);
    altitude2(j,1) = (0.0524*distance)*6076;  %tan(3)*dist, in feet
    time_stamp1(j,1) = j/2;
end

%%%%%%%%%Trailing Aircraft flight path%%%%%%%%%%%%%%%%%%%%

pi = 3.1415;
lati_radians = 42.36686*(pi/180);
lon1_radians = -71.00008*(pi/180);
distance_miles = 3.522;
dist_decr = 0.0222;  %for every 135 feet at 0.5 seconds when aircraft is
    travelling at 160kts( 1nm = 6076 feet)
i=1;
runway_info = 216;  %(in degrees)

distance = 10.5;
for i=1:1:45  %121 samples
    distance = distance-0.051;  %when travelling at 160knots/sec
    R = 6371;
    d = distance*1.85200;  % in km
    theta = runway_info*(pi/180);
    lat2(i,1)= asin(sin(lat1_radians)*cos(d/R) +
                      cos(lati_radians)*sin(d/R)*cos(theta));
    lon2(i,1)= lon1_radians +
                      atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-
                      sin(lati_radians)*sin(lat2(i,1)));
    lat2(i,1)=lat2(i,1)*(180/pi);
    lon2(i,1)=lon2(i,1)*(180/pi);
    altitude(i,1) = 2500;
    lat2_radians = lat2*(pi/180);
    initial_point = [1530000,-4478000,4265800];
    dist_radius = 2000;  %In feet
    theta_cone = 90;
    max_altitude(i,1)= search_algorithm_function(i,lat2,lon2,
                      lat3,lon3,altitude2,altitude,initial_point(1,:),dist_radius,theta_cone);
    time_stamp(i,1) = i/2;
end
distance = 7.725;
for i=46:1:105 %121 samples
    distance = distance-0.051; %when travelling at 160knots/sec
    R = 6371;
    d= distance*1.85200;% in km
    theta = runway_info*(pi/180);
    lat2(i,1)= asin(sin(lat1_radians)*cos(d/R) + cos(lat1_radians)*sin(d/R)*cos(theta));
    lon2(i,1)= lon1_radians + atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-sin(lat1_radians)*sin(lat2(i,1)));
    lat2(i,1)=lat2(i,1)*(180/pi);
    lon2(i,1)=lon2(i,1)*(180/pi);
    altitude(i,1) = (0.0524*distance)*6076; %tan(3)*dist, in feet
    initial_point = [1531900,-4474400,4268800];
    dist_radius = 2000; %in feet
    theta_cone = 90;
    max_altitude(i,1)= search_algorithm_function(i,lat2,lon2,
    lat3,lon3,altitude2,altitude,initial_point(1,:),dist_radius,theta_cone) ;
    time_stamp(i,1) = i/2;
end

distance = 4.665;
for i=106:1:364 %121 samples
    distance = distance-0.018; %when travelling at 130knots
    R = 6371;
    d= distance*1.85200;% in km
    theta = runway_info*(pi/180);
    lat2(i,1)= asin(sin(lat1_radians)*cos(d/R) + cos(lat1_radians)*sin(d/R)*cos(theta));
    lon2(i,1)= lon1_radians + atan2(sin(theta)*sin(d/R)*cos(lat1_radians), cos(d/R)-sin(lat1_radians)*sin(lat2(i,1)));
    lat2(i,1)=lat2(i,1)*(180/pi);
    lon2(i,1)=lon2(i,1)*(180/pi);
    altitude(i,1) = (0.0524*distance)*6076; %tan(3)*dist, in feet
    lat2_radians = lat2*(pi/180);
    initial_point = [1533800,-4469700,4271500];
    dist_radius = 2000; %in feet
    theta_cone = 90;
    max_altitude(i,1)= search_algorithm_function(i,lat2,lon2,
    lat3,lon3,altitude2,altitude,initial_point(1,:),dist_radius,theta_cone) ;
    time_stamp(i,1) = i/2;
end

time_stampl = sort(time_stampl,'ascend');
time_stamp = sort(time_stamp,'ascend');
time_stamp = time_stamp+50;
figure,
plot3(lon2,lat2,altitude,lon3,lat3,altitude2,'r','LineWidth',2);
xlabel ('Longitude in degrees');
xlabel ('Longitude in degrees');
ylabel ('Latitude in degrees');
legend ('GA aircraft flight path', 'Heavy aircraft flight path');
title('Aircraft Arrival Profile 3 KBOS Airport 3 Dimensional');
grid on;

figure,
plot(lon2,lat2,lon3,lat3,'r','LineWidth',2);
xlabel ('Longitude in degrees');
ylabel ('Latitude in degrees');
legend ('GA aircraft flight path', 'Heavy aircraft flight path');
title('Aircraft Arrival Profile 3 KBOS Airport');
grid on;

figure,
plot(time_stamp,altitude,time_stamp,max_altitude,'r','LineWidth',2);
xlabel ('Time in Sec');
ylabel ('Altitude in feet');
legend ('GA aircraft flight path', 'Wake strip altitude');
title('Vertical plot for scenario 3 KBOS Airport');
grid on;

C.4 Scenario 4 - May 2\textsuperscript{nd}, 2011 at KUNI airport

%% Author - Nikhil Tej Gandhi
%% Ohio University
%% 11/15/2012
clc
close all
clear all

load('flight_data_parsed.mat');
i=1;
dist_radius = 3000; %In feet
theta_cone = 90;
for j = 1801:1:2719
    lat3(i,1) = lead_lat_deg_parsed(j,:);
    lon3(i,1) = lead_lon_deg_parsed(j,:);
    altitude2(i,1) = lead_alt_m_parsed(j,:);
lat2(i,1) = trail_lat_deg_parsed(j,:);
lon2(i,1) = trail_lon_deg_parsed(j,:);
altitude(i,1) = trail_alt_m_parsed(j,:);
time_stamp(i,1) = run_time_min(j,:);
i=i+1;
end

lat2_radians = lat2*(pi/180);
lon2_radians = lon2*(pi/180);
altitude = altitude*3.28084;
rand = [lat2_radians,lon2_radians,altitude];
for i=1:1:2719-1800
    ecef_trailing(i,:) = llh2ec(rand(i,:));  \ %%ECEF coordinates of the trailing aircraft
end

lat3_radians = lat3*(pi/180);
lon3_radians = lon3*(pi/180);
altitude2 = altitude2*3.28084;
rand1 = [lat3_radians,lon3_radians,altitude2];
for i=1:1:2719-1800
    ecef_heading(i,:) = llh2ec(rand1(i,:));  \ %%ECEF coordinates of the heading aircraft
end

for i=1:1:919
    alt = 0;
    %initial_point = ecef_trailing(i,:);
    apex = ecef_trailing(i,:);  \ %%Apex of the cone
    trailing_point2 = -apex + initial_point;
dist_trailpoints = sqrt((apex(1,1) - initial_point(1,1))^2+(apex(1,2) - initial_point(1,2))^2+(apex(1,3) - initial_point(1,3))^2);
    height_cone = 0.25*1.85200;
    cone_axis = -
apex+(trailing_point2/dist_trailpoints)*height_cone*1000;  \ %calculating cone axis using linear interpolation
    norm_coneaxis = cone_axis/norm(cone_axis);
    test_point = ecef_heading;
    %Searching for every location on the heading aircraft
    for j=1:1:719
        dist_testvector = sqrt((test_point(j,1) - apex(1,1))^2+(test_point(j,2) - apex(1,2))^2+(test_point(j,3) - apex(1,3))^2);
        if dist_testvector>dist_radius
            continue;
        else
            test_vector = test_point(j,:)-apex;
            norm_testvector = test_vector/norm(test_vector);
            dot_product = dot(norm_testvector,norm_coneaxis);
phi = acos(dot_product);  \% Angle computed between the cone axis vector and test vector
if phi<0
    continue;
else
    phi = phi*180/pi;
end
if phi>theta_cone  \% Points which fall outside the cone are eliminated
    continue;
else
    alt(j,1) = altitude2(j,1);  \% Altitude of all the points on heading aircraft which satisfy the distance and angle condition.
end
end
max_altitude(i,1) = max(alt);
end

figure,
plot(lon2,lat2,lon3,lat3,'r','LineWidth',2);
xlabel ('Longitude in degrees');
ylabel ('Latitude in degrees');
legend ('GA aircraft flight path', 'Heavy aircraft flight path');
title('Aircraft Arrival scenario - Real flight data, KUNI Airport');
grid on;

figure,
plot3(lon2,lat2,altitude,lon3,lat3,altitude2,'r','LineWidth',2);
xlabel ('Longitude in degrees');
ylabel ('Latitude in degrees');
zlabel ('Altitude in feet');
legend ('GA aircraft flight path', 'Heavy aircraft flight path');
title('Aircraft Arrival scenario - Real flight data, KUNI Airport');
grid on;

plot(time_stamp,altitude,time_stamp,max_altitude,'r','LineWidth',2);
xlabel ('Time in Sec');
ylabel ('Altitude in feet');
legend ('GA aircraft flight path', 'wake altitude');
title('Vertical plot for scenario - Real flight data, KUNI Airport');
grid on;

figure,
plot(time_stamp,altitude2,'g','LineWidth',2);
xlabel ('Time in Minutes');
ylabel ('Altitude in feet');
legend ('Heavy aircraft flight path');
title('Vertical plot for scenario - Real flight data, KUNI Airport');
grid on;
C.5 Search Algorithm Function

%% Search Algorithm Function
%% Author - Nikhil Tej Gandhi
%% Ohio University
%% 10/1/2012

%%%% Search Algorithm Function 1 %%%%%%%%%%%%%%

function [max_altitude] = search_algorithm_function(i, lat2, lon2, lat3, lon3, altitude2, altitude, initial_point, dist_radius, theta_cone)

lat2_radians = lat2*(pi/180);
lon2_radians = lon2*(pi/180);
rand = [lat2_radians, lon2_radians, altitude];
ecef_trailing(i,:) = llh2ec(rand(i,:));  % ECEF coordinates of the trailing aircraft

lat3_radians = lat3*(pi/180);
lon3_radians = lon3*(pi/180);
rand1 = [lat3_radians, lon3_radians, altitude2];
for j = 1:1:403
    ecef_heading(j,:) = llh2ec(rand1(j,:));  % ECEF coordinates of the heading aircraft
end

alt = 0;
apex = ecef_trailing(i,:);  % Apex of the cone
trailing_point2 = -apex + initial_point;
dist_trailpoints = sqrt((apex(1,1) - initial_point(1,1))^2+(apex(1,2) - initial_point(1,2))^2+(apex(1,3) - initial_point(1,3))^2);
height_cone = 0.25*1.85200;
cone_axis = -apex + (trailing_point2/dist_trailpoints)*height_cone*1000;
% Calculating cone axis using linear interpolation
norm_coneaxis = cone_axis/norm(cone_axis);  % Normalizing the vector

for j = 1:1:403  % Searching for every location on the heading aircraft
    dist_testvector = sqrt((test_point(j,1) - apex(1,1))^2+(test_point(j,2) - apex(1,2))^2+(test_point(j,3) - apex(1,3))^2);
    if dist_testvector > dist_radius
        continue;
    else
        test_vector = test_point(j,:) - apex;
        norm_testvector = test_vector/norm(test_vector);
        dot_product = dot(norm_testvector, norm_coneaxis);
        phi = acos(dot_product);  % Angle computed between the cone axis vector and test vector
        if phi < 0
            continue;
        else
            phi = phi*180/pi;
        end
    end
end

end
if phi>theta_cone  %Points which fall outside the cone are eliminated
    continue;
else
    alt(j,1) = altitude2(j,1);  %Altitude of all the points on heading aircraft which satisfy the distance and angle condition.
end
end
max_altitude = max(alt);
end
%end