Optimal Recovery Trajectories for Automatic Ground Collision Avoidance Systems (Auto GCAS)

DISSERTATION

Angela W. Suplisson, Colonel, USAF

AFIT-ENY-DS-15-M-002

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENY-DS-15-M-002

OPTIMAL RECOVERY TRAJECTORIES FOR AUTOMATIC GROUND COLLISION AVOIDANCE SYSTEMS (AUTO GCAS)

DISSERTATION

Presented to the Faculty

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

Angela W. Suplisson, B.S., M.E., M.A.

Colonel, USAF

March 2015

AFIT-ENY-DS-15-M-002

OPTIMAL RECOVERY TRAJECTORIES FOR AUTOMATIC GROUND COLLISION AVOIDANCE SYSTEMS (AUTO GCAS)

Angela W. Suplisson, B.S., M.E., M.A.
Colonel, USAF

Approved:

//signed//	5 January 2015
Dr. Richard G. Cobb Dissertation Advisor	Date
//signed//	5 January 2015
Dr. William P. Baker Committee Member	Date
//signed//	5 January 2015
Dr. David R. Jacques Committee Member	Date

Accepted:

//signed//	5 January 2015
Dr. Adedeji B. Badiru Dean, Graduate School of Engineering and Management	Date

Abstract

The US Air Force recently fielded the F-16 Automatic Ground Collision Avoidance System (Auto GCAS). This system meets the operational requirements of being both *aggressive* and *timely*, meaning that extremely agile avoidance maneuvers will be executed at the last second to avoid the ground. This small window of automatic operation maneuvering in close proximity to the ground makes the problem challenging. There currently exists no similar Auto GCAS for manned military 'heavy' aircraft with lower climb performance such as transport, tanker, or bomber aircraft. The F-16 Auto GCAS recovery is a single pre-planned roll to wings-level and 5-g pull-up which is very effective for fighters due to their high g and climb performance, but it is not suitable for military heavy aircraft. This research proposes a new optimal control approach to the ground collision avoidance problem for heavy aircraft by mapping the aggressive and timely requirements of the automatic recovery to the optimal control formulation which includes lateral maneuvers around terrain. This novel mapping creates two ways to pose the optimal control problem for Auto GCAS; one as a Max Distance with a Timely Trigger formulation and the other as a Min Control with an Aggressive Trigger formulation. Further, the optimal path and optimal control admitted by these two formulations are demonstrated to be equivalent at the point the automatic recovery is initiated for the simplified 2-D case. The Min Control formulation was demonstrated to have faster computational speed and was chosen for the 3-D case. Results are presented for representative heavy aircraft scenarios against 3-D digital terrain. The Min Control formulation was then compared to a Multi-Trajectory Auto GCAS with five pre-planned maneuvers. Metrics were developed to quantify the improvement from using an optimal approach versus the pre-planned

maneuvers. The proposed optimal Min Control method was demonstrated to require less control or trigger later for the same scenario, which leads to a system which satisfies the aggressive and timely operational requirements. The results of this research provide a basis to evaluate the expected performance of any future Auto GCAS for all aircraft.

This research is dedicated to my friends

Kim (Harmon) Weilhouwer,

Judson Brohmer,

Aaron 'Cdot' George,

Dave 'Cools' Cooley,

and all the brave men and women

who have lost their lives to ground collision accidents.

To my husband,

You are my best friend and my soul mate.

Thank you for sharing this great adventure with me.

To our sons,

You are my greatest pride and joy.

Follow your dreams, no matter how high –

the view from the top is worth the climb.

Acknowledgements

It is difficult to adequately thank everyone who contributed to the success of this research. Most importantly, I am immensely grateful to my advisor Dr. Richard Cobb for his leadership and friendship. Thank you for having faith in this research topic and in me. Dr. Cobb saw the clear road ahead when I could not. His extreme energy and enthusiasm for teaching and research is inspiring. I could not have accomplished this work without him. To my committee members, Dr. Dave Jacques and Dr. Bill Baker - thank you for your guidance, advice, and hours spent teaching me about optimal control and mathematics. I could not have asked for a better advisor and committee - your dedication to the students is remarkable.

I did not accomplish this work alone. I have been privileged to work with the most amazing people these last three years. My sincere thanks go to my sponsors, the AFRL Automatic Collision Avoidance Technology (ACAT) Team, who had the vision for this research; especially Don Swihart, Amy Burns, Dave Homan, Finley Barfield, Kerianne Gross, Michael Coy, Ba Nguyen, Matt Stoeckle, and Heather Stickey for the incredible support you provided me. A special thank you goes to Finley Barfield who spent countless hours teaching me about F-16 Auto GCAS and allowed me to pepper him with questions. I'd like to thank Dr. Anil Rao and Dr. Michael Patterson who developed GPOPS-II and were available anytime for advice and technical support. Mark Skoog, Paul Sorokowski, and Loyd Hook at NASA Armstrong were invaluable in teaching me about F-16 Auto GCAS, Small UAV Auto GCAS, and digital elevation models. I'd also like to thank Mark 'Tex' Wilkins and Jerry Aslinger at OSD Personnel & Readiness and Doug Egged at the F-16 SPO for their Auto GCAS insight, technical expertise, and their time spent reviewing my work. I'm also grateful for the time Tom Anderson spent teaching me about the Navy's Terrain Awareness Warning System. Thank you to the Lockheed Martin Fort Worth Advanced Development Programs

ACAT team led by Ed Griffin who taught me how the system works and about the history of the evolution of technology that was developed to combat CFIT in the F-16. I have the utmost respect for all those on the Auto GCAS team who have tirelessly worked towards combatting CFIT, preventing accidents, and saving lives. Many aircrew and families will have you to thank for future Auto GCAS saves.

Thank you to my friends at AFIT, especially all of my classmates in PhD Class 14S. To Ted Masternak, Jeff Morris, and Nate Smith, your friendship made this journey fun and worthwhile. I'll miss our time together. Ted was also my GPOPS-II guru and Cade Saie was my LaTeX guru - thanks to both of you for your patience in teaching the Colonel how to program. Kara Greene, Nidal Jodeh, Emily Knight, and Jeremy Agte kept me sane, fed, caffeinated, and laughing even when times were tough. And to Jack 'Cowboy' Trombetta, thank you for your friendship and fantastic work on the Multi-Trajectory Auto GCAS. I'm thankful you decided to research Auto GCAS with us - your work was essential to help me finish this research. And my thanks to Col Neal Barlow who allowed me to pursue this PhD so that I could come back and continue doing what I love, teaching cadets at the US Air Force Academy. I am indebted to all of you for making this research possible. Finally, thank you to my parents, husband, and family who supported me without fail and had the faith in me that I could finish this work. Boys, I'm sorry for all the times I had to study when you wanted to play - but Mommy is back now and we're going to have a blast!

Angela W. Suplisson

Table of Contents

Page

Page

List of Figures

Figure Page

List of Tables

Nomenclature

— Letters —

$\mathbf{A}$	admissible set of control inputs
$\mathbf{C}$	path constraint vector
$\mathbf{H}$	final state weighting matrix
$\mathbf{N}$	diagonal matrix for sphere calculations
$\mathbf{Q}$	state weighting matrix
$\mathbf{Q}$	vector of the distances between aircraft location and local DTED post locations
$\mathbf{R}$	control weighting matrix
$\mathbf{X_{ac}}$	vector of aircraft x, y, z position
$\mathbf{X_{Terrain}}$	vector of terrain x, y, z position
$\mathbf{X}$	admissible set of state values
C^0	curves are joined or connected
C^1	first derivatives are continuous
C^2	first and second derivatives are continuous
D	aircraft drag
F	interpolant
H_z	Hertz
J	cost functional (performance measure)

L Lagrangian that contains the integral cost (running cost)

L aircraft lift

N_z normal acceleration (load factor)

$N_{z_{max}}$ upper bounds on normal acceleration

$N_{z_{min}}$ lower bounds on normal acceleration

$N_{z_{Scaledmax}}$ upper bounds on scaled normal acceleration

$N_{z_{Scaled}}$ N_z that is scaled to be symmetric about zero and on the order of one

R aircraft turn radius

S normalized radius of a sphere

T aircraft thrust

U_i i^{th} control state

U_{max_i} max control state limit for the i^{th} control state

U_{min_i} min control state limit for the i^{th} control state

V aircraft velocity

$\dot{d}$ relative velocity of the aircraft to terrain

$\dot{d}_{t_f}$ relative velocity of the aircraft to terrain at the end of phase one

d distance of the aircraft to terrain

d_{climb} distance covered (during a 4,000 ft climb)

$d_{threshold}$ user-defined keep-out zone around terrain

g acceleration of gravity

$\mathbf{r}(t_f)$ vector of desired values of final states

r radius of a sphere (aircraft buffer)

r range (used as GPWS conflict detection threshold)

r_1 weighting on the μ term in the cost functional

r_2 weighting on the N_z term in the cost functional

s slack variable

t time

t_0 initial time (current time)

t_{90° path propagation time for lateral avoidance path for aircraft to turn 90°

t_{CPA} time at the closest point of approach

$t_{End\ Mission}$ mission landing time

$t_{fwd\ path}$ path propagation time for forward avoidance path for a 4,000 ft climb

t_f final time (time horizon)

$t_{Start\ Mission}$ mission takeoff time

$t_{trigger}$ trigger time of the Auto GCAS recovery

$\mathbf{u}$ control vector

u_i i^{th} control input

u_{max_i} max control limit for the i^{th} control input

u_{min_i} min control limit for the i^{th} control input

v_{w_x} wind velocity in the x-direction

v_{w_y} wind velocity in the y-direction

v_{w_z} wind velocity in the z-direction

$\dot{\mathbf{x}}$ equations of motion (dynamics)

$\mathbf{x}$ state vector

x aircraft x-location (East)

x_{ac} aircraft x-location

y aircraft y-location (North)

y_{ac} aircraft y-location

z aircraft z-location (Up)

z_{ac} aircraft z-location

z_{interp} interpolated terrain height

— Symbols —

α aircraft angle of attack

$\dot{\theta}$ aircraft turn rate for 2-D problem (rate of change of aircraft heading)

θ aircraft heading angle for 2-D problem

λ Lagrange multiplier for equality constraints

μ Lagrange multiplier for inequality constraints

μ aircraft bank angle

μ_{max} upper bounds on aircraft bank angle

μ_{min} lower bounds on aircraft bank angle

ω aircraft turn rate

τ predicted time to closest point of approach (used as GPWS conflict detection threshold)

τ time in the tau domain

ϕ Mayer term that contains the terminal penalty (final cost)

χ aircraft heading angle for 3-D problem

γ aircraft flight path angle

$\mathbf{\Psi}$ boundary constraint vector

— Acronyms —

2-D Two-Dimensional

3-D Three-Dimensional

3-DOF Three-Degrees of Freedom

3DEP Three-Dimensional Elevation Program

6-DOF Six-Degrees of Freedom

ACAT Automatic Collision Avoidance Technology

ACAT/FRRP ACAT/Fighter Risk Reduction Program

AFB Air Force Base

AFIT Air Force Institute of Technology

AFRL Air Force Research Laboratory

AFTI Advanced Fighter Technology Integration

AGL Above Ground Level

ALOW Altitude Low

AMAS Automated Maneuvering Attack System

APF Artificial Potential Function

ARS Automated Recovery System

ART Available Reaction Time

ASI TF Aviation Safety Improvements Task Force

AT GCAS All-Terrain Ground Collision Avoidance System

ATAWS Automatic Terrain Awareness Warning System

Auto ACAS Automatic Air Collision Avoidance System

Auto GCAS Automatic Ground Collision Avoidance System

Auto ICAS Automatic Integrated Collision Avoidance System

Auto-CA Automatic-Collision Avoidance

BFM Basic Fighter Maneuvers

CAF Combat Air Forces

CARA Combined Altitude Radar Altimeter

CAS Close Air Support

CDR Conflict Detection and Resolution

CDTM Compressed Digital Terrain Map

CFCC Commercial Fire Control Computer

CFIT Controlled Flight Into Terrain

CPA Closest Point of Approach

CRS Congressional Research Services

DBTC Data Base Terrain Cueing

DEM Digital Elevation Model

DFLCC Digital Flight Control Computer

DFLCS Digital Flight Control System

DOC Direct Orthogonal Collocation

DoD Department of Defense

DROID Dryden Remotely Operated Integrated Drone

DSOC Defense Safety Oversight Council

DTD Digital Terrain Data

DTED Digital Terrain Elevation Data

DTM Digital Terrain Model

DTS Digital Terrain System

DWAT Descent Warning After Takeoff

EGC Enhanced Ground Clobber

EGI Embedded GPS/INS

EGM-96 Earth Gravitational Model of 1996

EGPWS Enhanced Ground Proximity Warning System

ENU East-North-Up

EOM Equations of Motion

FAA Federal Aviation Administration

ft feet

FY Fiscal Year

g-LOC g-induced Loss of Consciousness

GAAF Ground Avoidance Advisory Function

GB Gigabyte

GCAS Ground Collision Avoidance System

GEDACS Global Elevation Data Adaptive Compression System

GPOPS-II General-Purpose Pseudospectral Optimal Control Software-II

GPS Global Positioning System

GPWS Ground Proximity Warning System

HAT Height Above Target

HBVP Hamilton Boundary Value Problem

HMD Helmet-Mounted Display

HUD Heads-Up Display

Hy-Tech Hybrid Technology

ICAO International Civil Aviation Organization

INS Inertial Navigation System

JPL Jet Propulsion Laboratory

KGS Knots Groundspeed

KKT Karush-Kuhn-Tucker

LCR Left-Center-Right

LG Legendre Gauss

LGL Legendre Gauss Lobatto

LGR Legendre Gauss Radau

LIDAR Light Detection and Ranging

LIMDIS Limited Distribution

LIS Line-in-the-Sky

m meters

MB Megabyte

MCD Minimum Clearance Distance

MFD Multi-Function Display

MMC Modular Mission Computer

MS20 Mission System 20

MS21 Mission System 21

MSL Mean Sea Level

Mt Mountain

MTC Minimum Terrain Clearance

NAD83 North American Datum of 1983

NASA National Aeronautics and Space Administration

NAVD88 North American Vertical Datum of 1988

NED National Elevation Dataset

NGA National Geospatial-Intelligence Agency

NLP Nonlinear Programming

NM Nautical Miles

ODE Ordinary Differential Equation

OFP Operational Flight Program

ORT Oblique Recovery Trajectory

OSD P&R Office of the Secretary of Defense, Personnel and Readiness

OW&C Obstacle Warning and Cueing

PARS Pilot Activated Recovery System

PGCAS Predictive Ground Collision Avoidance System

PMA Program Management Activity

PPC Path Propagation Criteria

PPL Path Propagation Length

PPT Path Propagation Time

PR Passive Ranging

PSCOL Gauss Pseudospectral Optimization Program

PSM Pseudospectral Method

RadAlt Radar Altimeter

RBF Radial Basis Function

REO Radar Electro Optical

RMSE Root Mean Squared Error

RPA Remotely Piloted Aircraft

RTO Responsible Test Organization

RTOC Real-Time Optimal Control

SAR Synthetic Aperture Radar

sec seconds

SecDef Secretary of Defense

SPO System Program Office

SRTM Shuttle Radar Topography Mission

STWG Safety Technology Working Group

SWIM System Wide Integrity Monitoring

TAF SON Tactical Air Force Statement of Operational Need

TAMMAC Tactical Aircraft Moving Map Capability

TAWS Terrain Awareness Warning System

TERPROM Terrain Profile Matching

TFR Terrain Following Radar

TM Telemetry

TPA Trajectory Prediction Algorithm

TPS Test Pilot School

TRN Terrain Referenced Navigation

UAV Unmanned Aerial Vehicle

US United States

USAF United States Air Force

USD(P&R) Undersecretary of Defense for Personnel and Readiness

USGS United States Geological Survey

VOD Vertical Obstruction Database

VRT Vertical Recovery Trajectory

VSS Variable Stability System

WGS-84 World Geodetic System of 1984

OPTIMAL RECOVERY TRAJECTORIES FOR AUTOMATIC GROUND COLLISION AVOIDANCE SYSTEMS (AUTO GCAS)

I. Introduction

1.1 Motivation

For the last three decades, the United States Air Force (USAF), National Aeronautics and Space Administration (NASA), and Lockheed Martin have been working together as a team on an Automatic Ground Collision Avoidance System (Auto GCAS) to reduce Controlled Flight Into Terrain (CFIT) mishaps, the #1 cause of fighter pilot fatalities in the USAF. According to the Federal Aviation Administration (FAA), "A CFIT accident occurs when an airworthy aircraft, under the control of a pilot, is flown into terrain (water or obstacles) with inadequate awareness on the part of the pilot of the impending disaster" [33]. Not every aviation organization defines CFIT the same way however, and therefore may, or may not, include preventable mishaps resulting from physiological incidents such as spatial disorientation, hypoxia, and/or g-induced Loss of Consciousness (g-LOC). Unlike manual warning systems, Auto GCAS can prevent mishaps even if the pilot is incapacitated and for that reason, the Auto GCAS team created its own mishap category called "Auto GCAS-Preventable." An Auto GCAS-preventable mishap is defined as a mishap where a fully functioning aircraft, with the gear up, impacts the ground due to pilot spatial disorientation, loss of situational awareness, target fixation, or physiological incapacity such as g-LOC or hypoxia [98]. Initial Development Auto GCAS testing started in 1984 with Follow-on Development testing from 1991-1993. Further, a Joint

US-Sweden Auto GCAS flight test program was accomplished from January 1997 to November 1998 on the Advanced Fighter Technology Integration (AFTI) F-16 at Edwards Air Force Base (AFB) CA. The Joint AFTI Auto GCAS program successfully defined the 80% solution of ground collision protection for fighter/attack aircraft. Some additional work remained but there was no additional funding to accomplish more flight test.

On 19 May 2003, the Secretary of Defense (SecDef) Donald Rumsfeld published a memorandum titled "Reducing Preventable Accidents" which mandated a 50% reduction in the number of mishaps and accident rates across the Department of Defense (DoD) [79, 87]. There were two major actions leading to this SecDef memo. First, Secretary Rumsfeld was interested in implementing a safety campaign to reduce workplace injury rate just as then Treasury Secretary Paul O'Neill had done while President and Chief Executive Officer at Alcoa from 1987 to 2000 and at the Department of Treasury from 2001 to 2002 [6]. The second event was the publishing of the Congressional Research Service (CRS) study "Military Aviation Safety" by Christopher Bolkcom dated September 16, 2002 [6]. Bolkcom's CRS study described an increase in Air Force aviation accident rates (Class A mishap rate of 1.76 accidents per 100,000 flight hours) after years of decline [13].

In his 2003 SecDef memorandum, Secretary Rumsfeld appointed Dr. Chu, the Undersecretary of Defense for Personnel and Readiness USD(P&R), to lead the effort to reduce DoD mishaps by 50% [24, 79]. Dr. Chu created the Defense Safety Oversight Council (DSOC) to manage the DoD-wide implementation [24]. This safety mandate was later increased to a 75% reduction of DoD mishaps on 15 Mar 2004 by Deputy Secretary of Defense Paul Wolfowitz in the Strategic Planning Guidance (classified document) for fiscal years (FY) 2006-2011 [6] and later endorsed by Secretary of Defense Robert Gates in his 30 May 2007 memorandum titled "Zero Preventable

Accidents" [36]. This set the goal year to be FY 2008 which was later extended to FY 2012 in the Guidance for the Development of the Force FY 2010-2015 [6].

In 2006 across the USAF, CFIT mishaps were the #1 cause of fighter pilot fatalities and the #2 cause of fighter mishaps [97]. From 2000 to 2013, the DoD fighter losses that were from Auto GCAS-preventable mishaps totaled 23 pilots and 29 aircraft for the F-16, F/A-18, and F-22 (see Table 1) [98]. Note that there were other DoD fighter accidents that occurred during this timeframe; however, this table only includes the fighters that were Auto GCAS compatible at the time (equipped with digital flight controls) [98].

Table 1. Auto GCAS-Preventable Pilot and Fighter Losses from 2000 to 2013 [98]

Aircraft Type	# Pilots Lost	# Aircraft Lost
F-16	14	17
F/A-18	7	10
F-22	2	2

One of the task forces established by the DSOC was the Aviation Safety Improvements Task Force (ASI TF) which was given the charter to reduce aviation mishaps across the DoD [24]. "The Safety Technology Working Group (STWG) was the ASI TF working group charged with identifying technological mitigation strategies for aviation mishap reduction" [24, 87]. The STWG published a report in February 2006 titled *Fighter/Attack Automatic Collision Avoidance Systems Business Case* which included a cost-benefit analysis for automatic ground collision avoidance systems. Table 2 shows a breakdown of the estimated savings from implementing Auto GCAS across some fighter/attack platforms in the DoD, which were calculated by applying the historical CFIT rate to the estimated remaining service life for each weapon system [24]. If Auto GCAS were to be implemented in these platforms, the totals would be 65 pilots and 100 aircraft saved ($4.8B in aircraft assets) [24]. "[This] thorough business case analysis showed that billions of dollars worth of aircraft could be saved

with a program costing less than a single new aircraft" [17]. Note that the data in the business case was based on the Combat Air Forces (CAF) in 2006, with the predicted service life of F-16s through 2025 and of F-22s and Air Force variant F-35As through 2035 [24]. The F/A-18 estimates are from 2008 to 2032 [24].

Table 2. DoD Estimated Pilots and Aircraft Saved due to Automatic Collision Avoidance Systems from 2008 to 2035 (data from [24])

Aircraft Type	# Pilots Saved	# Aircraft Saved	Aircraft Cost Savings
F-16[1]	11	17	$615M
F/A-18[2]	6	8	$665M
F-22[3]	5	7	$863M
F-35A[3]	43	68	$2.7B

[1]F-16 estimates are from 2011 to 2025 (predicted end of service life).

[2]F/A-18 estimates are from 2008 to 2032.

[3]F-22 and F-35A estimates are from 2011 to 2035.

All of the data in Table 2 was taken directly from the DSOC STWG's *Fighter/Attack Automatic Collision Avoidance Systems Business Case* from February 2006. The STWG calculated F-16 estimated future mishap rates based on F-16 historical mishap rates, estimated service life of current F-16s (in 2006) and their historical utilization hours [24]. The F-22 estimated mishap rates were calculated by using a blend of F-15 (80%) and F-16 (20%) historical mishap rates, while the Air Force variant F-35A estimated mishap rates were calculated by using a blend of F-16 (90%) and combined A-7 and A-10 (10%) historical mishap rates [24].

Historically, CFIT mitigation efforts have consisted of advanced training, increased pilot experience, and pilot activated (i.e. manual) safety systems such as the F-16 Predictive Ground Collision Avoidance System (PGCAS) in the 1990s and the F/A-18 Terrain Awareness Warning System (TAWS) in the early 2000s [24]. According to the DSOC in 2006, "Training has had some success in reducing CFIT and MIDAIR rates in the past, but reductions in the rates have long been stagnant and no large improvements from training are envisioned for the future" [24]. Despite these mitigation

efforts, the DoD fighter CFIT rate has not shown a statistically significant change in 40 years [24]. According to Swihart et al., "both the Navy TAWS and the AF's PGCAS are manual systems requiring the pilot to maneuver the aircraft to avoid the collision. These systems may have had some success in reducing CFIT mishaps, but the magnitude of their improvement is not enough to achieve statistical significance. The human being is now the limiting factor because he or she cannot always recognize a warning or respond appropriately to prevent a mishap" [87]. Because the human is now the limiting factor, the DSOC has concluded that the comprehensive 75% reduction in fighter mishaps cannot be achieved without the widespread implementation of an automatic system such as Auto GCAS [24]. Therefore, in 2007, with DSOC support, the Air Force Research Laboratory (AFRL) initiated the Automatic Collision Avoidance Technology/Fighter Risk Reduction Program (ACAT/FRRP). The ACAT/FRRP effort ultimately cost approximately $26M. The goal of ACAT/FRRP was to perform Auto GCAS risk reduction and enable a faster technology transition to DoD fighter platforms, specifically the F-16 Blocks 40/50, F-22, and F-35.

The conclusions of ACAT/FRRP testing were that Auto GCAS provided excellent mishap prevention and nuisance[4]-free operation. Testing was completed on the production Auto GCAS algorithm in March 2013 at the 416^{th} Flight Test Squadron at Edwards AFB CA as part of the Operational Flight Program (OFP) M6.2+ and was fielded on all operational F-16 Block 40 and 50 aircraft by Dec 2014.

The F-16 Auto GCAS automatic recovery maneuver is a roll to wings-level and 5-g pull-up (until terrain is cleared) which is extremely effective in saving the life of a pilot with g-LOC or spatial disorientation. However, it is a single pre-planned maneuver regardless of the flight conditions or terrain. Variations in available load factor from 5 g down to 2 g are taken into account by the system to provide protection

[4]A nuisance warning is defined as an undesired/early activation of Auto GCAS that could impede mission operations.

at low speeds. A roll to wings-level and pull-up is very effective for fighters due to their high performance capability which allows going over obstacles with relative ease. For other aircraft, alternative maneuvers may be more effective.

The Auto GCAS algorithm, as well as manual ground collision avoidance algorithms, focuses on when to execute the pre-planned recovery maneuver and not what the optimal recovery maneuver should be. If Auto GCAS technology were to be applied to other aircraft such as cargo, transport, or general aviation aircraft, going over a high terrain obstacle may not be viable depending on the aircraft's climb performance, or it would require the aircraft to start climbing miles out in order to go over the high obstacle, when a lateral escape[5] maneuver to go around the obstacle could be more easily accomplished. An optimal lateral recovery maneuver may also increase mission effectiveness. For example, a lateral escape maneuver that goes around a mountain obstacle instead of over it could allow the aircraft to get closer to the obstacle before turning in order to continue on a mission-critical low-level reconnaissance route.

The goal of this research is to formulate optimal[6] ground collision avoidance maneuvers which include lateral escapes and then apply that technique to aircraft with lower climb performance such as tanker, transport, and bomber aircraft. An optimal solution may require more computational resources onboard than a pre-planned solution to determine solutions. A natural question then may be to ask if real-time implementation of an optimal control solution onboard an aircraft for automatic ground collision avoidance is feasible. The idea of implementing real-time optimal control (RTOC) is not new. Advances are being made in available computing power as well as robust optimal control solution techniques. "Efficient pseudospectral methods

[5]The term escape refers to any trajectory from a set of safe path trajectories that avoid terrain.

[6]Any trajectory produced by the solution of an optimal control problem will be loosely referred to as an 'optimal' trajectory.

have made real-time solutions of the optimal control problem not only possible, but a topic of great interest" [78]. In recent years, the advent of tools such as SOCS [10], DIDO [77], and General-Purpose Pseudospectral Optimal Control Software [74] (hereafter referred to as GPOPS-II) makes it possible for real-time implementation and allows for evaluation of "best possible" maneuvers. Ongoing research in that area shows that it is possible for certain scenarios. Two recent examples of RTOC programs that have been demonstrated in flight test are the Air Force Institute of Technology (AFIT)'s quadrotor in 2011 [78] and NASA's Small Unmanned Aerial Vehicle (UAV) in 2013 [83]. The AFIT quadrotor research in 2011 solved a bearing-only trajectory planning problem, autonomously landing a quadrotor on a wire [78]. NASA's Small UAV Auto GCAS control algorithm consisted of three pre-planned escape maneuvers and an algorithm that chose the best one based on predicted intersections with digital terrain [83]. Small UAV Auto GCAS not only demonstrated RTOC by choosing the best of three escape maneuvers to avoid the ground but also hosted the entire Auto GCAS algorithm and digital map of the US on a smart phone onboard the UAV, demonstrating that even with the current limiting factors of throughput and capacity that there is enough throughput and capacity in existing systems to perform some types of RTOC [83].

For an autonomous collision avoidance solution to be acceptable to the pilot community, the system must operate only after an aware pilot would recover the aircraft [88]. Even if the Auto GCAS prevented a ground collision, it would be considered an impedance to mission performance if at the time of actuation, an aware pilot still had time to recover with another trajectory [88]. In such cases, the pilot may consider this early activation a nuisance and decide to simply turn the system off [88]. Therefore, before developing new algorithms it is important to understand the current Auto GCAS Design Criteria, which communicate the Auto GCAS philosophy. The

Design Criteria (in order of priority) are [88]:

1. Do No Harm
2. Do Not Interfere
3. Avoid Ground Collisions

The top priority, "Do No Harm," means that Auto GCAS was designed to ensure that the aircraft would not be auto-recovered into terrain. In other words, only maneuver the aircraft if there is high certainty that the maneuver will not make the situation worse. This is ensured through Integrity Management. "Do Not Interfere" means do not impede the pilot from performing mission operations which is accomplished by eliminating nuisance warnings (as previously defined). Nuisance warnings are eliminated by starting an avoidance maneuver after the time an aware pilot would have recovered. "Avoid Ground Collisions" is defined as automatically preventing collisions with the ground. Avoiding ground collisions is achieved by projecting the future aircraft avoidance trajectory onto digital terrain, and if the trajectory intersects the terrain plus a safety buffer, initiating the maneuver at the last moment.

Note that while it might seem like the primary design criteria should be Avoid Ground Collisions, it is third. Why? This is due to the fact that preventing ground collisions cannot be successfully accomplished unless there is high certainty that the maneuver makes the situation better and the algorithm is nuisance free so that the pilot can perform the mission. Thus, there are two potentially competing objectives when it comes to Auto GCAS performance; preventing ground collisions while not impeding normal operations (nuisance potential). In other words, the Avoid Ground Collisions and Do Not Interfere design criteria are interconnected, and the relationship is very important. The threshold between the two is *when* the automatic recovery will be initiated. "It can also be reasoned that the dividing line between a valid warning and a nuisance warning is the point where an aware pilot feels an aggressive recovery

must be initiated to avoid collision" [87]. Through detailed flight test and analysis at Edwards AFB CA from January to October 1997, this threshold has been determined to be 1.5 seconds (sec) of 'Time Available' for an F-16. A Time Available of zero sec is defined as the point in time after which the pre-planned recovery maneuver will no longer provide a safe path away from terrain. If the automatic recovery was allowed to initiate earlier than 1.5 sec Time Available, the early recovery could be perceived by the F-16 pilot as a nuisance and impede mission performance.

The 1.5 sec threshold was determined through flight test as follows. For the purpose of Auto GCAS testing, the F-16 was equipped with a flight test aid system called Pilot Activated Recovery System (PARS) [7]. PARS was designed for the situation where a pilot realizes he is disoriented. PARS can be initiated by the pilot at any time and it automatically returns the aircraft to straight and level flight with a slight nose-high climb [7]. For nose-low conditions, the PARS recovery is identical to the Auto GCAS recovery of an automatic roll to wings-level and pull to 5 gs [88]. However, it should be noted that PARS has no knowledge of terrain, and therefore cannot be used in place of Auto GCAS [89].

For the 1997 nuisance criteria flight testing, PARS recoveries were used for repeatability and consistency across all pilots [88]. The PARS recoveries were initiated by several test pilots at the altitude where they felt comfortable recovering. The buildup approach used was to test at progressively lower altitudes as long as the pilot felt comfortable with the recovery from the previous buildup run at the same initial conditions[84]. After each PARS recovery, the test pilot gave an anxiety rating as described in Figure 1. As shown in Figure 1, the dividing line between a nuisance recovery and an acceptable recovery to prevent ground collision was determined to be 1.5 sec of Time Available [87]. Hence 1.5 sec became the Design Criteria Limit. The F-16 Auto GCAS recovery was designed to be commanded only after the 1.5 sec

nuisance boundary is crossed [87].

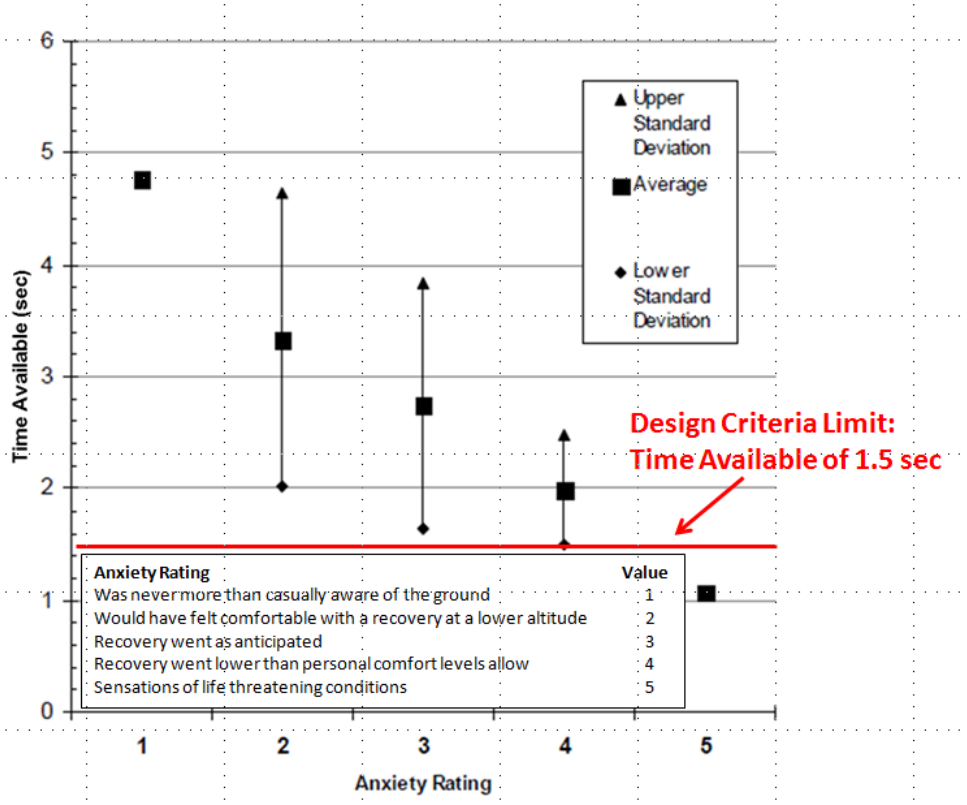


Figure 1. Time Available Design Criteria from Jan-Oct 1997 Flight Test [87].

Using this nuisance criteria, flight tests have proven Auto GCAS to be very effective and acceptable to F-16 pilots. Although this criteria was developed for fighters, the same approach could be applied when extending this to other platforms, including military heavy aircraft. At present the Design Criteria Limit (nuisance boundary) is not known for any military heavy aircraft. If Auto GCAS is to be integrated into a heavy aircraft, a similar series of flight tests must be done to determine the nuisance boundary for that specific aircraft. The fact that the automatic recovery must be implemented after the nuisance boundary and while close to the ground makes the Auto GCAS problem challenging.

1.2 Heavy Auto GCAS

Auto GCAS has been developed and tested on the F-16, but no similar automatic system has been developed for military heavy aircraft such as transport, tanker, or bomber aircraft. Two of the reasons are the fact that heavy aircraft have at least two pilots onboard and that there currently is not a business case for a heavy Auto GCAS. One reason that an automatic approach to ground collision avoidance has yet to be fielded on crew aircraft is the philosophy that although one pilot may be unaware of the impending collision and incapable of performing a recovery maneuver, the other pilot should be aware and able to recover. This mutual support aspect of flying does in fact limit the instances of CFIT, but it by no means eliminates it. Further, because heavy aircraft do not fly at a high g loadings, g-LOC is not an issue as it is for fighter aircraft, thus, eliminating one of the major causes of fighter CFITs. However, despite two pilots being onboard, ground collisions with fatalities still occur in crew aircraft, typically in high task saturation environments.

The second reason stems from the lack of a business case for heavy Auto GCAS. With fighters, much of the research and testing of F-16 Auto GCAS had been accomplished before the business case was written. One of the catalysts for finally obtaining an operational requirement and funding for the final testing and fielding of Auto GCAS for fighters was the Fighter/Attack Automatic Collision Avoidance Systems Business Case published in 2006 by the DSOC STWG [24]. In a similar way, a business case for a heavy Auto GCAS is necessary to justify an operational requirement and funding to enable eventual fielding of the system on candidate aircraft. Until then, research should still be conducted to further the development of the technology. Military heavy aircraft continue to operate both tactically and in training at low-level altitudes, therefore the potential for ground collision mishaps still exists.

The goal for this research is to determine efficient and effective Auto GCAS meth-

ods for heavy aircraft so that the foundation of the system is in place when the need or business case arises. Ultimately, Auto GCAS increases safety, mitigates risk, and protects aircrew and assets. For these reasons, it is a logical and necessary approach for long-term aviation safety needs. This research seeks to show the technology exists to implement a heavy Auto GCAS so that future mishaps can be prevented.

1.3 Research Objectives

The F-16 Auto GCAS algorithm can command the flight control system to execute a single recovery maneuver, a roll-to-wings-level and 5-g[7] pull-up until terrain is cleared. When a high performance jet flies over smooth terrain, a wings-level pull-up is the best recovery and is acceptable over rough terrain as well. However, for aircraft with lower climb performance that fly slow in rough terrain, a lateral escape maneuver is preferred in many cases. A versatile Auto GCAS should include one or more lateral escape options and is the focus of this research.

The primary research objective is to synthesize optimal control with Auto GCAS for lower climb performance aircraft, determining optimal recovery trajectories. As defined previously, an optimal trajectory is any trajectory produced by the solution of an optimal control problem. In finding the optimal recovery trajectories, the goal of this research is also to identify appropriate applications to different military heavy platforms. With these goals in mind, the specific research objectives are:

1. Optimal Recovery Trajectory for Aircraft with Lower Climb Performance

   Develop and evaluate an optimal control technique to calculate an Auto GCAS recovery trajectory (to include lateral avoidance maneuvers) for aircraft with lower climb performance. Candidate aircraft considered are low, medium, and high-speed military heavy aircraft.

[7]During slower speed operations, the F-16 Auto GCAS algorithm will provide protection down to 2 gs available [89].

2. Comparison of Optimal Auto GCAS to a Pre-Planned Trajectory Algorithm

   Conduct a limited comparison of the Optimal Auto GCAS algorithm and an algorithm with pre-planned trajectories. For this objective, the Optimal Auto GCAS will be compared to the Multi-Trajectory Auto GCAS algorithm with five pre-planned automatic recovery maneuvers [91].

The Multi-Trajectory Auto GCAS is follow-on research that is currently being conducted by Trombetta at AFIT, but is not yet published [91]. Comparison between the simulated Multi-Trajectory Auto GCAS and the Optimal Auto GCAS will be done as part of this research.

If successful, the results of the research herein will provide a validated approach for an Optimal Auto GCAS that can be applied to a variety of platforms for both simulation and flight test purposes. Further, metrics will be created that allow comparison between optimal recovery trajectories and pre-planned recovery trajectories for Auto GCAS design and analysis purposes.

1.4 Assumptions and Limitations

The optimal control technique developed for this research is limited to a three-degrees of freedom (3-DOF) simplified point mass aircraft model. Appropriate limits on the states and controls of this model allow the model to have performance characteristics similar to low, medium, and high-speed military heavy aircraft. However, the aircraft model does not account for the onset rate of controls, bur rather assumes that instantaneous control is possible. The model also assumes coordinated turns. Constant aircraft velocity is assumed based on the fact that the Auto GCAS activation duration is relatively short (less than 10 seconds).

For this research, Digital Terrain Elevation Data (DTED) Level 1 which has approximately 90 meter (m) post[8] spacing is used. In some cases, research has shown

[8] A post is an x, y, z data point representing longitude, latitude, and elevation.

that compressing the digital terrain data (i.e. thinning out the posts) is desirable [83]. The assumption is that the method developed for this research will work whether the digital terrain data used is compressed or not. All scenarios for this research were run against the digital terrain for one mountain on the Edwards AFB range; the assumption is that this terrain is representative of mountainous terrain.

If successful, the algorithms developed in this research will require real-time implementation. Proving the feasibility of real-time implementation is not a goal of this research, however, necessary steps were taken to make the method as computationally efficient as possible.

1.5 Document Outline

Chapter II provides necessary background on collision avoidance and optimal control. First a literature review of existing methods used for collision avoidance is presented followed by a review of manual and automatic ground collision avoidance systems that have been implemented on aircraft. Next, the relevant theory in regards to the optimal control problem formulation is covered to include appropriate cost functionals. The optimal control approach to Auto GCAS and the proposed experimental approach are then presented in Chapter III, followed by a 2-D example. Chapter IV presents how to formulate the path constraint and cost functional of the 3-D optimal control formulation for Auto GCAS, and Chapter V presents the full methodology for the 3-D optimal control formulation along with results for a low-speed heavy aircraft. Conclusions and future work are discussed in Chapter VI.

The appendices provide additional details to complement the material in the main body. Appendix A presents how to build the DTED matrices and the mappings between reference frames, Appendix B describes the methodology to interpolate terrain heights, Appendix C provides the methodology for applying Trombetta's Path Prop-

agation Criteria [91], and Appendix D includes the settings for the GPOPS-II and nonlinear programming (NLP) solver IPOPT software used in this research.

II. Literature Review

This chapter begins with a survey of existing methods used for both air and ground collision avoidance. Next a literature review of both manual and automatic ground collision avoidance systems (to include both proposed and in-service systems) is presented, followed by the presentation of the general formulation of the optimal control problem. Finally, an overview of the solution technique via direct orthogonal collocation, the aircraft model, and the digital terrain model used for this research are explained.

2.1 Air and Ground Collision Avoidance Modeling Methods

The preponderance of the literature on aircraft collision avoidance deals with air collision avoidance. Since many of the techniques for avoiding other aircraft can be applicable to avoiding terrain, a review of air collision modeling techniques is presented here alongside that of ground collision modeling techniques.

One methodology that includes aspects of both is that of conflict detection and resolution (CDR). An influential framework categorizing CDR methods and models was created by Kuchar and Yang [54, 55], who reviewed 68 CDR modeling methods for air and ground collision avoidance as well as for naval applications between 1997 and 2000. "A conflict is an event in which two or more aircraft experience a loss of minimum separation" and thus could be on a collision course [55] or in which an aircraft's future path is predicted to hit the ground. As defined by Kuchar and Yang, "The goal for the CDR system is to *predict* that a conflict is going to occur in the future, *communicate* the detected conflict to a human operator and, in some cases, assist in the *resolution* of the conflict situation" [55].

For air collision avoidance scenarios, the human operator could be the pilot, remote

pilot, or an air traffic controller. Since the majority of the air collision scenarios are strategic deconflictions such that the collision is averted miles out, *predicting* that a conflict will occur and *communicating* the detected conflict to these human operators early is vital. None of the systems that Kuchar and Yang reviewed had automatic recoveries to *resolve* the conflict situation, but rather had either manual resolution cues for techniques that can be implemented by a human operator or only warnings with no resolution cues at all.

For the research herein, the human operator is the pilot. *Predicting* that a conflict is going to occur and *resolving* the conflict with an automatic recovery are paramount. Although very important for manual ground collision avoidance systems, *communicating* the detected conflict to the pilot is not as important for automatic systems which assume that the pilot is unaware and unable to avoid the impending disaster. For a manual system, an example is Predictive GCAS (PGCAS) on the F-16, *communicating* a terrain conflict is absolutely essential since PGCAS provides an early warning that assumes an aware pilot.

Kuchar and Yang categorized the CDR methods by surveying 68 different algorithms (63 of which were for air collision avoidance). From this, they developed a taxonomy that includes 1) dimensions of state information, 2) method of dynamic state propagation, 3) conflict detection threshold, 4) conflict resolution method, 5) maneuvering dimensions, and 6) management of multiple aircraft conflicts [55]. The sixth characteristic, management of multiple aircraft, addresses air-to-air conflicts and is not directly applicable to a stand-alone ground collision avoidance system. Therefore, only the first five characteristics of Kuchar and Yang's taxonomy are applicable to ground collision avoidance and are discussed here. The majority (63 out of 68) of the methods they reviewed dealt with air collision avoidance, however some of the solution techniques can also be applied to ground collision avoidance. Three

of the models reviewed by Kuchar and Yang were developed for maritime collision avoidance [22, 50, 90, 54]. Two of the 68 systems specifically address the problem of ground collision, namely Ground Proximity Warning System (GPWS) and Enhanced GPWS (EGPWS). Therefore, the CDR attributes of GPWS and EGPWS are examined at the end of each the following five sections.

2.1.1 Dimensions of State Information.

The dimensions of state[9] information refer to whether the states measured by the algorithm include miss distances (closest approach distances) purely in the vertical plane, purely in the horizonal plane, or as three-dimensional (3-D) data [55]. Out of the 68 systems reviewed by Kuchar and Yang, only the GPWS algorithm measures state information purely in the vertical plane and therefore its state information includes altitude as well as climb (or descent) rate. Of the remaining 67 systems, 28 algorithms use states with dimensions only in the horizontal plane and 39 use 3-dimensional states. The EGPWS algorithm utilizes 3-D state information [55]. While it is clear for ground collision avoidance that as a minimum vertical state information must be used in the model, 3-D state information should be used for an automatic ground collision avoidance system such that not only terrain below the aircraft but also terrain in front of the aircraft can be avoided.

2.1.2 Method of Dynamic State Propagation.

In order to determine the future trajectory's point of closest approach to the terrain, propagated states and propagated controls are needed. Once propagated, the trajectory can be compared to the terrain and a point of closest approach is

[9]In general, state refers to the aircraft position vector as well as the aircraft velocity vector, but can also include aircraft information such as bank angle and heading angle. If the model includes only the horizontal plane, the aircraft position and velocity vectors will be two dimensional in the x and y axes and for the models that incorporate the horizontal and vertical planes, the position and velocity vectors will be three dimensional in the x, y, and z axes [54].

calculated to determine if a collision will occur. The three fundamental methods to propagate the aircraft dynamics forward can be characterized as nominal, worst-case, and probabilistic [55]. For the nominal method, the algorithm takes whatever the current state of the aircraft is and propagates that forward over time. In essence, it is assuming no change in pilot inputs. "An example would be extrapolating the aircraft's position based on its current velocity vector," [55] as shown in Figure 2 (a).

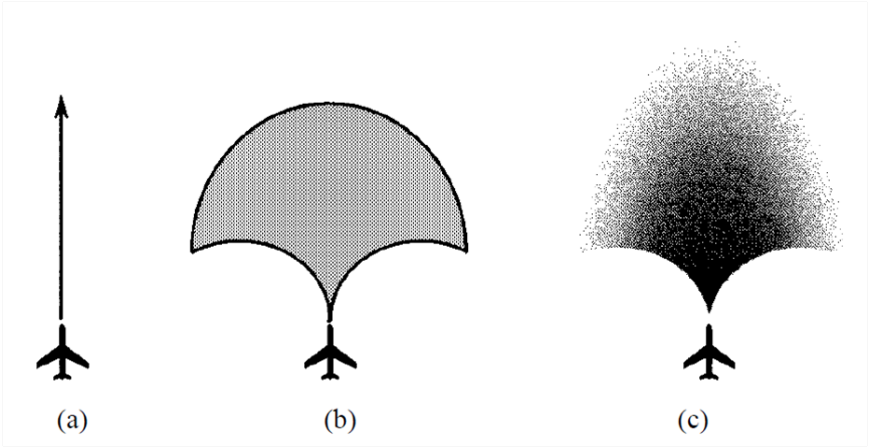


Figure 2. State Propagation Methods [54, 55]

In contrast, the worst-case method answers the question, "What is the most extreme (worst possible) maneuver the aircraft could perform based on its current state?" The benefit to including worst-case propagations is that they map out all possible trajectories the aircraft could fly as shown in Figure 2 (b) which can help predict all possible 'conflicts' that could arise [55].

Between the nominal and worst-case methods lies the probabilistic method as shown in Figure 2 (c). Note that "the probabilistic method is also the most general; the nominal and worst-case methods are subsets of probabilistic trajectories" [55]. Both the GPWS and EGPWS algorithms use the nominal method to propagate the aircraft dynamics forward [54, 55].

2.1.3 Conflict Detection Threshold.

The 68 algorithms were surveyed to see if conflict detection thresholds were used to determine when alerts are issued. Kuchar and Yang found that 19 of the algorithms do not have an explicit conflict detection threshold of some type while 47 algorithms do include explicit conflict detection thresholds [55]. These thresholds are metrics created from state information that are necessary to make decisions [54] and/or to trigger a resolution method. Some examples of conflict detection thresholds which the research herein will also consider are current range to the point of closest approach (to terrain or to other aircraft) as well as time to point of closest approach, miss distance if no escape trajectory is implemented, maneuvering cost, and probability of conflict [54]. Both GPWS and EGPWS have explicit conflict detection thresholds. GPWS uses τ, predicted time to closest point of approach, and r, range as thresholds [54], while EGPWS uses caution and warning ribbons as thresholds. Figure 3 depicts the EGPWS caution ribbon in yellow and the warning ribbon in red [43].

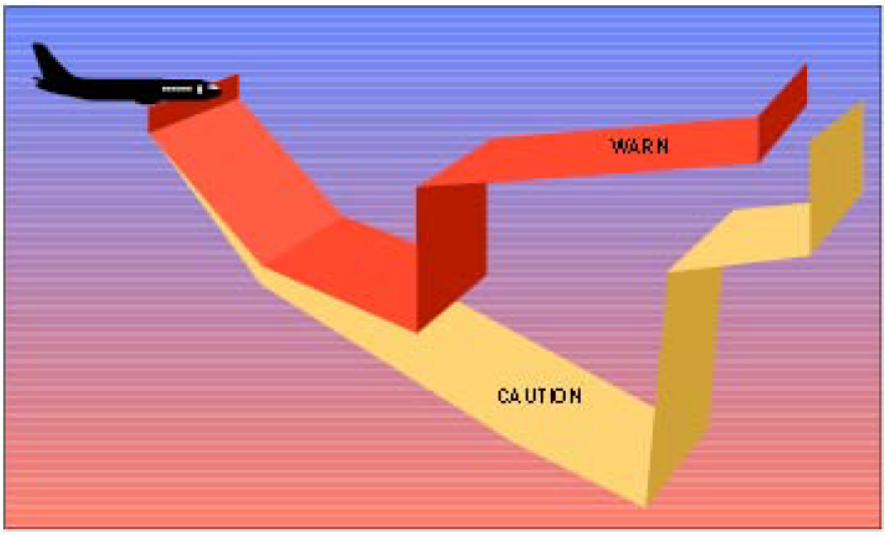


Figure 3. EGPWS Caution and Warning Ribbons [43]

"These ribbons project down, forward, then up from the aircraft with a width starting at 1/4 NM and extending out at ±3° laterally, more if turning. The look-down

and up angles are a function of the aircraft flight path angle, and the look-down distance a function of the aircraft's altitude with respect to the nearest or destination runway. This relationship prevents undesired alerts when taking off or landing. The look-ahead distance is a function of the aircraft's speed, and distance to the nearest runway" [43]. If either terrain or an obstacle intersect with these ribbons, then the appropriate caution or warning alerts and messages are generated. These EGPWS alerts and messages are provided at the end of Section 2.1.4. GPWS and EGPWS functions are discussed further in Section 2.2.4.

2.1.4 Conflict Resolution Method.

Kuchar and Yang describe five methods used for resolving conflicts which include 1) prescribed, 2) optimized, 3) force field, 4) manual, and 5) no method [55]. These five methods are briefly described in the following paragraphs.

"Prescribed resolution maneuvers are fixed during system design based on a set of predefined procedures. For example, GPWS issues a standard 'PULL UP' warning when a conflict with terrain exists. GPWS does not perform additional computation to determine an optimal escape maneuver" [55]. GPWS depends on an aware pilot to respond to the aural warning and to pull up to perform a manual escape maneuver.

The second conflict resolution method is an optimized approach which uses aircraft dynamics, path constraints and a cost functional specific to the problem. The solution to the problem is an optimal control input (that in this research is commanded by the autopilot) that minimizes the cost functional while at the same time satisfies the dynamic constraints (aircraft equations of motion) and path constraints.

Thirdly, a force field type of approach can be used. Force field approaches "treat each aircraft as a charged particle...[and] the repulsive forces between aircraft are used to define the maneuver each performs to avoid a collision" [55]. Artificial potential

functions (APFs), although not synonymous with force field approaches, are similar and can also be used for collision avoidance. The Air Force Institute of Technology (AFIT) has performed some research on adapting APFs to aircraft ground and air collision avoidance methods [1, 34, 85]. In AFIT's APF approach, all components of the problem are treated as potential fields, where obstacles (terrain or intruder aircraft) are modeled as sources indicating something to be avoided and goals are modeled as sinks indicating a waypoint, target, runway, or other goal to be achieved.

It is worthwhile to note that APFs and optimal control solutions demonstrate two extremes of real-time approaches that are both within the feasible solution space [1]. Using the APF methodology to model the terrain obstacles as sources and the goals as sinks guarantees the aircraft will not hit the terrain at a relatively low computational cost [1]. Once an APF is chosen to model the obstacle space, the aircraft is commanded to follow the contours of that function by incorporating a simple feedback control based on the negative of the gradient of the APF. However, the APF solution is sub-optimal and therefore the control effort required will be higher than the optimal control solution which minimized control. On the other extreme, using an optimal control approach guarantees that a minimum control effort solution is found (assuming global convergence) because of the cost functional being minimized. However, the aircraft may be arbitrarily close to the constraint boundaries and the ability to compute an optimal solution at the desired 10-12 Hz rate may result in a high computational burden.

The fourth conflict resolution method is a manual method. A manual conflict resolution technique allows "the user to generate potential conflict resolution solutions and obtain feedback as to whether the trial solution is acceptable" [55]. The advantage provided by a manual solution is that the human operator (air traffic controller or pilot) can take into account other information, such as weather, that the CDR

algorithm can not 'see' [55].

Finally, a system can have no method of conflict resolution. Having no resolution method simply means that while the algorithm can compute conflict detection, it is not designed to calculate an avoidance action [55]. For example, EGPWS provides cautions and warnings to the pilot but does not provide the pilot with an avoidance maneuver.

The GPWS conflict resolution method is further described here. According to the 1995 Industry/International Civil Aviation Organization (ICAO) CFIT Task Force [49], the prescribed GPWS warning escape maneuver recommended is:

1. React immediately to a GPWS warning.
2. Positively apply maximum thrust and rotate to the appropriate pitch attitude for your airplane.
3. Pull up with wings level to ensure maximum airplane performance.
4. Always respect the stick shaker.

In the earliest versions of GPWS, the available time between initial warning and aircraft impact was as little as five seconds. According to the Industry/ICAO CFIT Task Force, "Incidents and accidents have occurred because flight crews have failed to make timely and correct responses to the GPWS warnings. The available time has increased between initial warning and airplane impact since the first version of the GPWS; however, this time should not be used to analyze the situation. React immediately." [49]. Today's GPWS and EGPWS systems provide up to 60 seconds of time from the warning to airplane impact [43]. For EGPWS, the caution alert is typically provided 60 seconds ahead of the terrain conflict (caution lights come on in addition to the aural message "CAUTION TERRAIN, CAUTION TERRAIN"), while the warning alert is typically provided 30 seconds ahead (warning lights come on along with the aural warning "TERRAIN, TERRAIN PULL UP") [43].

2.1.5 Maneuvering Dimensions.

If a resolution maneuver (aka trajectory) is computed by the algorithm, it is important to classify those recovery maneuvers to provide a framework for comparison. Kuchar and Yang catalog the maneuver dimensions by 1) lateral/turning maneuvers only (Turn), 2) vertical maneuvers only (Vert), 3) speed change only (Sp), or 4) combined maneuvers C(*), where * denotes the maneuver type [54, 55]. Sp/Turn would indicate that either speed changes or turning maneuvers can be performed but not both simultaneously [54, 55]. If an algorithm computes a combined climb and turn maneuver, for example, the notation C(Turn/Vert) would be used to indicate that a "simultaneous climbing or descending turn may be performed" [54, 55].

Table 3 lists a summary of the CDR approaches used for current operational systems: including GPWS and EGPWS based on Kuchar and Yang's 1997 and 2000 survey papers [54, 55] as well as the F-16 Auto GCAS method [87, 88].

Table 3. Conflict Detection and Resolution Approaches for Current Operational Systems (GPWS and EGPWS data from [54, 55])

Method Name	State Dimension	Propagation Methods	Detection Thresholds	Resolution Maneuver	Maneuver Dimensions
GPWS[10]	Vertical	Nominal	τ, r	Prescribed	Vert
EGPWS[10]	3-D	Nominal	Ribbons	None	N/A
F-16 AGCAS	3-D	Worst-case[11]	d	Prescribed	C(Turn/Vert)

[10]GPWS and EGPWS data derived from [54] [55].

[11]Worst-case as defined by the physiological constraints on the pilot, not necessarily the limits of the airframe.

The threshold τ in this context is defined as the predicted time to the closest point of approach to the terrain, r is range and d is predicted miss distance [54].

In the current F-16 Auto GCAS approach which was fielded in the latter part of 2014, a combined roll and pull maneuver is automatically commanded once the predicted miss distance d in the propagated path (known as the Trajectory Prediction Algorithm or TPA) is less than or equal to the threshold value. If the current roll angle

of the aircraft is less than or equal to 90° (left or right), then the g-onset and the roll start at the same time as a combined maneuver [7]. If the current roll angle is greater than 90°, the autopilot first rolls the aircraft towards wings-level, and then once the aircraft rolls through 90° the autopilot starts to command the pull/g-onset [7].

By comparing the three approaches listed in Table 3, it can be seen that none of these operational ground collision avoidance system use anything other than a prescribed resolution maneuver or no resolution maneuver. After a review of the literature, the optimal control method which provides the resolution maneuver developed in this research is explained in detail in Chapter III.

2.2 Current Ground Collision Avoidance Systems

The system of systems involved in ground collision avoidance includes 1) sensors, 2) terrain databases, 3) the ground collision avoidance algorithm, and 4) displays/annunciators. In the F-16 for example, the Global Positioning System (GPS), and Inertial Navigation System (INS) are sensors which provide aircraft state information to the Auto GCAS recovery trajectory prediction algorithm which in turn produces a predicted trajectory. The terrain map scanning algorithm converts the 3-D digital terrain to a 2-D terrain profile, and the predicted trajectory is compared to this reduced set of data pulled from the Digital Terrain Elevation Data (DTED) terrain database. These processes are depicted in the Auto GCAS architecture in Figure 4 [87, 88]. If the collision estimation algorithm calculates there is an intersection between the predicted trajectory and the 2-D terrain profile, the digital flight control system (DFLCS) automatically recovers the aircraft with a last-second pre-planned trajectory. A loud aural tone will sound and 'break X' symbology on the heads-up display (HUD) will be projected indicating an auto recovery. A break X denotes two chevrons that come in, one from each side of the HUD, as a warning. When the two chevrons touch and

form an X, that is the signal that a recovery will be initiated immediately.

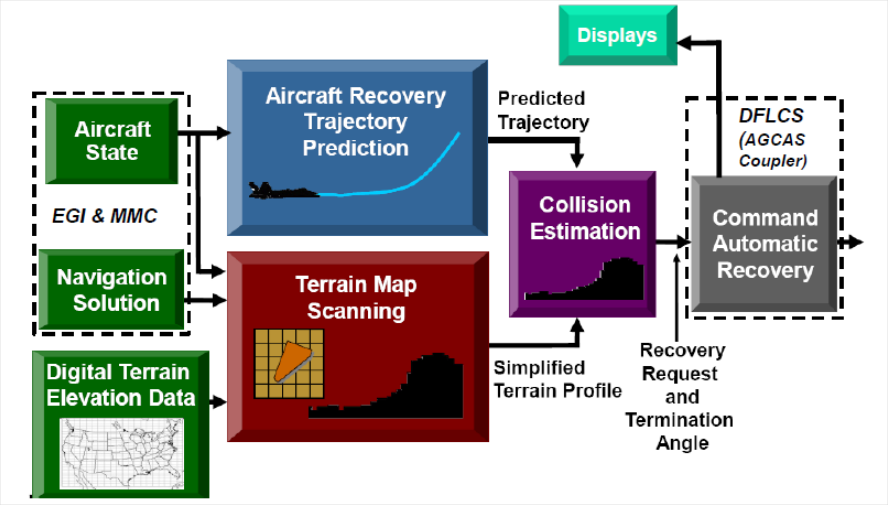


Figure 4. Auto GCAS Architecture [87, 88]

Annunciators are important for manual systems which are typically warning systems that assume an aware pilot. An automatic system operates as a last-second emergency system and assumes the pilot is unaware. Therefore the annunciators are not as important for automatic systems and are not discussed here. A review of sensors, terrain databases, and both manual and automatic algorithms is given here to provide proper context to the main focus of this research which is algorithm development for determining optimal recovery trajectories.

2.2.1 Sensors.

In order to guarantee the pilot continuous protection from terrain, Auto GCAS must be continuous and precise. For continuous protection, the F-16 Auto GCAS algorithm computes (at 12 Hz) a predicted trajectory and then compares it to the digital terrain. For precision location information, the F-16 Auto GCAS algorithm utilizes GPS and INS information via the Embedded GPS/INS (EGI). If the EGI information is degraded past acceptable tolerances, the System Wide Integrity Moni-

toring (SWIM) system puts Auto GCAS in standby mode. EGI is the primary sensor not only for Auto GCAS, but also for PGCAS and F/A-18 TAWS.

The F-16 is a combat fighter that was designed to minimize radar emissions that could alert the enemy to its location. Hence care was taken in the design of the Auto GCAS system to offer continuous protection and be precise but at the same time not make the F-16 vulnerable to the enemy. Using an active sensor such as radar to map the terrain would make the aircraft vulnerable to enemy detection. One distinguishing characteristic of the F-16 Auto GCAS is that its primary sensors are passive to ensure that the continuous protection requirement of Auto GCAS does not create a vulnerability. This characteristic is in contrast to aircraft systems which employ terrain following radar (TFR) that typically utilize a forward-looking radar or light detection and ranging (LIDAR) with active emissions.

2.2.2 Terrain Databases.

Once the sensors calculate the position and velocity of the aircraft, the Auto GCAS algorithm compares the aircraft location to its pre-loaded digital terrain database to determine where the aircraft is in relation to terrain. Currently, there are multiple choices for terrain databases which are generically known as Digital Elevation Models (DEMs), Digital Terrain Data (DTD), or Digital Terrain Models (DTMs) [95]. This research uses the term DEM defined as "any digital representation of the continuous variation of relief over space" [19]. A typical DEM is a regular gridded matrix of terrain elevation data, where the elevation z is measured in meters and represents the terrain elevation at specific longitude, latitude points. DEMs can be built from many raw data sources, for example geological surveys, quadrangle maps, and/or aerial surveys like aerial radar interferometry. The three types of database formats important to this research are Legacy Digital Terrain Elevation Data (DTED), Shuttle Radar

Topography Mission (SRTM) DTED, and the National Elevation Dataset (NED).

2.2.2.1 Legacy DTED.

DTED (also known as Legacy DTED to distinguish it from the SRTM DTED) is a 3-D database format produced by the National Geospatial-Intelligence Agency (NGA) and consists of latitude, longitude and terrain elevation data. Each lat/lon/elev triplet is referred to as a 'post'. Legacy DTED is available for most of the world from approximately 84° North to 84° South latitude [95].

Legacy DTED is a collage of all sorts of different data sources taken across different time periods. The data was taken over decades, so it is possible that the best source in one area of the country came from 1940 and in another area from 1970, each with different accuracies. "Conventional topgraphic mapping technologies have produced maps of uneven quality – some with astounding accuracy, some far less adequate" [31]. When Legacy DTED was investigated for use with F-16 Auto GCAS and National Aeronautics and Space Administration (NASA) Small UAV Auto GCAS, it was found that it was not suitable for an Auto GCAS system because of its discontinuities. According to the NASA Small UAV Auto GCAS technical report, "Legacy DTED has been a common source for terrain data prior to SRTM. Although Legacy DTED was a useful source in its time it has many issues with discontinuities across longitude-latitude boundaries and localized artifacts that can results in vertical errors of hundreds of feet (in some cases over a thousand feet of vertical error)" [83]. There are several different levels of fidelity of the DTED data. DTED Level 0 has 30 arc second (∼900 m) post spacing, DTED Level 1 has 3 arc second (∼90 m) post spacing, DTED Level 2 has 1 arc second (∼30 m) post spacing, and DTED Level 3 has $\frac{1}{3}$ arc second (∼10 m) post spacing [62]. DTED Level 0 is available to the public through the NGA website, and DTED Levels 1, 2, and 3 (regardless of geographic area of

coverage) are considered Limited Distribution (LIMDIS) and only available to the DoD [35, 63].

Many agencies had a need for a "globally consistent" and more accurate set of digital terrain for scientific studies of plants, animals, and soil, geological studies, recreational use, military use, and flight planning and navigation [62]. With the advent of synthetic aperture radar (SAR) interferometry in the 1990s, collecting this more accurate data efficiently from space was now possible [31]. The Space Shuttle was chosen as the vehicle to carry two SARs (X-band and C-band) to capture this high fidelity terrain data because it could access most of the world's terrain and capture the data in a short time period [31, 62].

2.2.2.2 Space Shuttle Radar Topography Mission DTED.

The DTED was updated through the Shuttle Radar Topography Mission (SRTM) which flew from 11 to 22 February 2000 on Space Shuttle Endeavor during STS-99 and mapped 80% of the Earth's surface (from 60° North to 56° South) [31, 62]. The northern coverage is higher due to the fact that the side-looking radars were pointed toward the north side of the Space Shuttle [93]. The SRTM was a joint venture between NASA and NGA. The SRTM DTED is pre-processed by NASA's Jet Propulsion Laboratory (JPL) and then forwarded to the NGA for final processing and production [62]. The SRTM DTED horizontal datum is the WGS-84 reference ellipsoid, and the vertical datum is the geoid (MSL) as determined by the WGS-84 EGM-96 geoid [64]. The advantage that SRTM DTED has is that it came from one source, using one method and was done over a very short time period of 10 days (and a total of 159 orbits) [31]. This leads to a more homogeneous SRTM DTED database as opposed to Legacy DTED which is known to have discontinuities. "The rows at the north and south edges as well as the columns at the east and west edges of each

cell overlap and are identical to the edge rows and columns in the adjacent cell" [31].

It should be noted that Legacy DTED can have very high levels of accuracy in some areas of the world, however there is no guarantee that this high accuracy exists everywhere the aircraft needs to fly. Therefore, SRTM DTED was found to be better suited for an Auto GCAS for military aircraft that fly worldwide [4, 83]. The original version of SRTM DTED had some voids as well as spikes and wells [31] and was not acceptable for use in Auto GCAS [7]. NGA was aware that "many applications require a continuous DEM with no voids", and developed methods to fill these voids [31]. An updated version called void-filled SRTM DTED fixed these problems and is discussed in Farr et al. [31].

One key difference between Legacy DTED and SRTM DTED is the fact that Legacy DTED represents the bare-earth terrain elevation while SRTM DTED represents "first return" elevation; i.e. radar returns that can capture vegetation and man-made buildings and towers [45]. In dense vegetation, the SRTM radars captured the elevation of the tops of the canopies, while in less dense vegetation, the radars may have captured partial canopy heights or may have been able to penetrate to the bare earth [31]. Since an aircraft colliding with a man-made feature or dense forest is just as catastrophic as an aircraft colliding with the ground, SRTM DTED was again found to be most appropriate for an Auto GCAS [45]. SRTM DTED Level 1 and 2 data were both examined for this research. SRTM-1 is publicly available, but SRTM-2 was only available to the DoD until recently. In Sep 2014, the White House announced that SRTM-2 will now be made available to the public over the next 12 months [62, 35]. The SRTM-2 data is the full resolution of the original measurements from the Space Shuttle mission [62]. SRTM-2 for the US and Africa is currently released to the public, and the next planned releases this year are for South America and North America south of the US [62]. Table 4 shows the post spacing

in arc-seconds, post spacing in meters, cells per degree, and matrix size for some of the DTED and SRTM DTED levels. Note that for longitude, these nominal post-spacing measurements are at the equator. The terms resolution and post spacing are synonymous and will be used interchangeably.

Table 4. Digital Terrain Elevation Data Types (data from [31, 62])

DTED Level	Post Spacing (arc-seconds)	Post Spacing (ground dist)	Cells per Degree	Matrix Size (1° x 1° tile[12])
DTED-0	30 arc-sec	∼900 m	120	121 x 121
DTED-1/SRTM-1	3 arc-sec	∼90 m	1,200	1,201 x 1,201
DTED-2/SRTM-2	1 arc-sec	∼30 m	3,600	3,601 x 3,601
DTED-3	$\frac{1}{3}$ arc-sec	∼10 m	10,800	10,801 x 10,801

[12]A tile is defined as a 1° x 1° grid (60 NM x 60 NM at the equator) of digital terrain data [83]. The terms tile and facet are used interchangeably [83].

2.2.2.3 National Elevation Dataset.

A third type of DEM is the NED which is produced by the United States Geological Survey (USGS). The NED is available in 2, 1, $\frac{1}{3}$ arc-second (∼10 m), and $\frac{1}{9}$ arc-second (∼3.33 m) resolutions [94]. All but the $\frac{1}{9}$ arc-second resolution contain seamless elevation layers from tile to tile [94]. Although the $\frac{1}{3}$ arc-second NED is only available for the continental United States and parts of Alaska, its advantages are that it is publicly available and has higher resolution than SRTM DTED Level 2 [94]. The 1 arc-second NED is available for the contiguous United States, Hawaii, Puerto Rico, US territorial islands, Mexico, and Canada, while the 2 arc-second resolution also includes all of Alaska [94]. In addition to its high horizontal accuracy, the $\frac{1}{3}$ arc-second NED also has high vertical accuracy (eight feet Root Mean Square Error (RMSE)) [83]. The horizontal datum is the North American Datum of 1983 (NAD83), and the vertical datum for the contiguous United States (it varies for other areas) is the North American Vertical Datum of 1988 (NAVD88) [94]

The USGS is currently designing another DEM called the 3D Elevation Program (3DEP) to eventually replace the NED [82]. "The primary goal of 3DEP is to systematically collect enhanced elevation data in the form of high-quality light detection and ranging (lidar) data over the conterminous United States, Hawaii, and the U.S. territories" [82]. The 3DEP will be another database alternative for aircraft wanting to integrate Auto GCAS, however, the fact that the database is currently limited to the US means limited military application [91].

The two Auto GCAS programs most relevant to this research are the F-16 Auto GCAS and the NASA Small UAV Auto GCAS programs. The F-16 Auto GCAS program utilizes the void-filled SRTM DTED Level 1 for the world data while the NASA Small UAV program utilizes the $\frac{1}{3}$ arc-second NED for the US, the void-filled SRTM DTED Level 1 for the non-US landmass, and other sources for any landmass not captured by the NED or SRTM DTED [83]. Because NED is the same source used by Google-Earth, an additional advantage was that the NASA Small UAV test team could use Google-Earth extensively for test planning and data analysis [83].

Now that sensors and terrain databases have been reviewed, two different ways of working with the discrete digital terrain are reviewed next.

2.2.3 Terrain Modeling/Approximation.

The two methods that have been flight tested and proven, the F-16 Auto GCAS method and NASA's CDTM method for Small UAV Auto GCAS, are discussed here.

2.2.3.1 F-16 Auto GCAS Terrain Scanning Method.

The F-16 DTED scan is a software scan of digital terrain vice an active scan of actual terrain such as with radar [88]. Based on the aircraft's current location, the F-16 algorithm scans the DTED terrain elevation with a fan-shape scan and

converts the 3-D digital terrain into a 2-D terrain profile data (elevation vs. range) for quicker computational speed. Comparisons with simplified 2-D terrain data are possible because the fan-shape scan includes all terrain within a reasonable distance around the predicted trajectory. The height of a 2-D range bin is determined by the highest post within each bin [88]. Figure 5 shows the Terrain Scanning and Binning Process for the F-16 Auto GCAS.

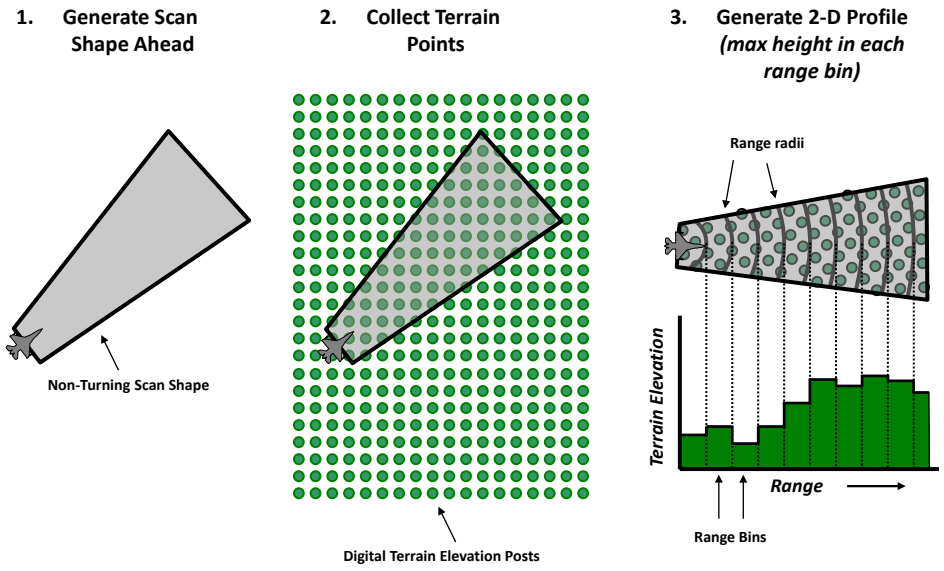


Figure 5. F-16 Terrain Scanning and Binning Process [88]

2.2.3.2 Advantages and Disadvantages.

The primary advantage of this method is that it is computationally fast, however it could sacrifice some accuracy. For example, in a low-level flight in a canyon, the algorithm can grab a post on the canyon wall and assign that height as the corresponding range bin height. In early flight tests, this would prohibit the pilot from flying as low as desired in the canyon. As a remedy, a check was added to the algorithm to automatically detect flight conditions where the pilot is purposely flying

low [88]. In these cases, the system reduced the scan shape to a narrower polygon which is the absolute minimum size possible [88]. This smaller scan shape allows the pilot to fly through the canyon with a reduction in nuisances, however there is also a reduction in CFIT protection as well.

2.2.3.3 NASA's Compressed Digital Terrain Map Method.

NASA Dryden developed the Compressed Digital Terrain Map (CDTM) specifically for Auto GCAS applications. CDTM is the map output by the Global Elevation Data Adaptive Compression System (GEDACS), a powerful tool that NASA Dryden created for aircraft which do not have the required amount of onboard digital storage available for the high resolution digital terrain map. The full SRTM DTED worldmap (60° North to 56° South) is 400 Gigabytes (GB) in size. However, not all current platforms have this much digital storage available, therefore NASA created a unique 'Tip-Tilt' compression method where digital terrain data are represented as planes that are tipped or tilted to fit terrain vice terrain being represented as discrete posts in DTED. For the entire world, the digital terrain was reduced from ∼400 GB to ∼170 Megabytes (MB), for an impressive compression ratio of more than 2,400:1 while maintaining elevation accuracy sufficient for an Auto GCAS [83]. With CDTM, even aircraft with very limited storage "can carry the whole world" [71]. If only a local map was needed (for example a 1° x 1° tile), the average size of an SRTM DTED 1° x 1° tile is ∼2.76 GB, and using the Tip-Tilt method the CDTM would be only ∼1.2 MB.

Many mathematical compression models exist. The original DTED is in raster format where each gridded square shown in Figure 6 represents the height of the DTED post which is located at the center of the square. The regular methods of compression are basic thinning (low resolution raster), and tip-tilt (rectangular sloped

facets (or tiles) tilted to approximate the terrain) [58]. Two semi-regular methods are the binary-tree (a single tile is bisected if errors exceed tolerances) and the quad-tree (a single tile is quartered if errors exceed tolerances) [58]. Figure 6 illustrates different types of digital terrain compression methods.

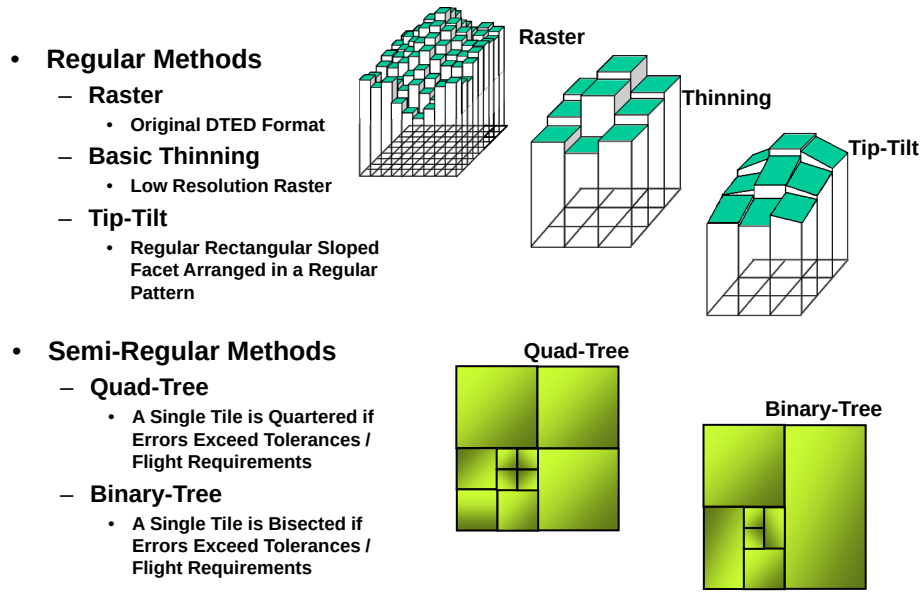


Figure 6. Digital Terrain Compression Methods [58]

NASA's CDTM implements a binary-tree tip-tilt compression method where digital terrain data is represented as planes that are tipped or tilted to fit terrain vice terrain being represented as posts in DTED. The source data for CDTM is a combination of NGA DTED and USGD NED which is stored in raster format. "The source raster data is fed into algorithms which take the discrete raster values and create a continuous representation based on rectangular tiles" [46]. A tile is vertically biased such that the tile is above all of the source raster data; in other words, the compression algorithm only allows errors in the conservative direction [58].

GEDACS allows the user to customize digital terrain data for any application.

For example, GEDACS can combine the best available sources to provide the most useful aspects of Legacy DTED, SRTM DTED and/or NED in different geographical areas. Thus GEDACS creates a CDTM product that provides higher accuracy where needed, such as within Special Use Airspace, and lower accuracy where not needed, such as areas where the aircraft would only be en-route at high altitudes [84]. NASA Dryden has successfully flight tested the CDTM map on an Android phone on their Small UAV Auto GCAS tests in 2012 [71, 83].

2.2.3.4 Advantages and Disadvantages.

The F-16 terrain scan method and the CDTM both have advantages and disadvantages.

For a real-time implementation of Auto GCAS, one advantage to compressing the data is that it can be searched more quickly. Another advantage is that no interpolation is required between terrain posts because a tile has been fit between each post. The CDTM is a continuous realization but is only class C^0, and therefore not differentiable. If the solution technique employed in Auto GCAS requires terrain gradient information, it is not available.

Now that two possible methods of terrain modeling have been presented, specific ground collision avoidance algorithms are discussed next. Manual ground collision avoidance algorithms are covered first, followed by automatic ground collision avoidance algorithms. Section 2.2.4 is provided to put the research and algorithms developed and documented herein in context with other systems.

2.2.4 Manual Ground Collision Avoidance Algorithms.

In order to provide an understanding of the history that laid the foundation for automatic collision avoidance systems, manual systems are discussed to provide a

complete background. A great amount of research and development has taken place over the last three decades in the area of ground collision avoidance, and the knowledge gained from development of the manual systems also enabled the development of Auto GCAS. Manual ground collision avoidance systems can be grouped in two categories, those that utilize DTED and those that do not [28]. The initial focus will be on the history of the F-16 manual altitude warnings systems. First, the manual systems that do not use DTED are reviewed, which include many F-16 systems as well as Ground Proximity Warning System (GPWS). Second, the manual systems that utilize DTED are reviewed, including Enhanced GPWS, US Navy Terrain Advisory Warning System (TAWS), TAWS II, and the F-16 systems that utilize DTED.

2.2.4.1 Manual Systems that Do Not Use DTED.

F-16 Manual Ground Collision Avoidance Systems

The F-16 includes many manual collision avoidance or warnings systems which have been added over the years to increase CFIT protection. These manual systems merit mentioning because they show the cumulative learning process and technology advances that were current at the time. The intent is to show the evolution of technology that was developed to combat CFIT on the F-16 over the years, ultimately resulting in the F-16 Auto GCAS technology. This section details the history of predictive (manual) F-16 ground collision avoidance systems which include Line-in-the-Sky with the Modular Mission Computer and barometric altimeter as the sensors, Altitude Low warning with the radar altimeter as the sensor, Descent Warning After Takeoff with multiple sensors, and ground clobber (known as Ground Avoidance Advisory Function) with the air-to-ground radar or the radar altimeter as the sensor.

– *Line-in-the-Sky (LIS)*

All F-16 A/B aircraft were originally equipped with Line-in-the-Sky (LIS) mecha-

nization, and all F-16 C/D aircraft are equipped with LIS as well. With LIS, the pilot can select a mean sea level (MSL) altitude 'floor'. The Modular Mission Computer (MMC) in the F-16 Block 40/50 (and the commercial fire control computer (CFCC) in the F-16 Block 25/30/32) aircraft uses inputs from the barometric altimeter to provide a reference signal to compute altitude [28]. The MSL altitude floor set by the pilot is constantly compared to the MMC's estimate of actual system MSL altitude. If the MMC estimate of aircraft MSL altitude goes below or penetrates the pilot's MSL floor set in LIS, then the system provides an aural warning of 'ALTITUDE! ALTITUDE!' as well as a visual warning [56]. The MSL floor warning provided by LIS is useful for basic fighter maneuvers (BFM), i.e. dogfighting.

– *Altitude Low (ALOW)*

All blocks of the F-16 aircraft currently have Combined Altitude Radar Altimeter (CARA) capability which includes an Altitude Low (ALOW) warning mechanization. CARA ALOW is mechanized with a pilot-selectable above ground level (AGL) altitude floor. The calculations for triggering the visual and aural warnings are accomplished by the MMC in the F-16 Block 40/50 (and by the CFCC in the F-16 Block 25/30/32) [28]. The MMC compares the AGL altitude floor to the CARA's own AGL altitude. When the aircraft penetrates the pilot-selected AGL floor, the MMC provides two types of visual warnings: the MMC flashes the radar altitude on the HUD and illuminates the 'RDR ALT LOW' warning light, in conjunction with an aural warning of 'ALTITUDE! ALTITUDE!' [56]. The AGL floor warning is beneficial for low-level operations over relatively smooth terrain [56, 84].

– *Descent Warning After Takeoff (DWAT)*

All F-16 blocks currently have the Descent Warning After Takeoff (DWAT) functionality to warn of an impending collision of terrain if the aircraft has insufficient climb rate after takeoff. DWAT requires no pilot action and is automatically active

for the first three minutes after takeoff (including touch and go takeoffs)[57]. If the landing gear is up and the time since takeoff is under three minutes, the DWAT system is armed at 300 feet above the runway MSL elevation and is de-armed at 10,000 feet above the runway MSL elevation or three minutes after takeoff, whichever comes first [57]. DWAT uses a combination of barometric altitude and CARA altitude. DWAT "has no pilot selectable options" [57]. If the current aircraft descent rate is such that it will cause the aircraft to penetrate the runway altitude in 30 seconds, then DWAT provides an aural 'ALTITUDE! ALTITUDE!' warning (only issued once per takeoff) [57].

– *"Ground Clobber" or Ground Avoidance Advisory Function (GAAF)*

Early F-16 Blocks had another capability called 'ground clobber' which used the air-to-ground radar and radar altimeter as sensors. The ground clobber mechanization was "based on air-to-ground radar ranging data or system altitude height above steering point and is available only in air-to-ground weapon delivery modes" [56]. Height above target (HAT) (computed using air-to-ground ranging data) is the primary altitude input to the system for calculating aircraft AGL [56]. When air-to-ground ranging is not available, aircraft AGL is based on the system altitude height above steering point [56].

The original ground clobber provided only visual warnings, which include a 'break X' on the HUD as well as a matrix/line of Xs on the radar electro optical (REO) display. Enhanced ground clobber (EGC) mechanization was integrated on the F-16 C/D Block 40 in the mid-1980s [56]. The EGC software takes into account weight and drag and a terrain clearance buffer to improve predictions and pullout maneuvers. For example, the EGC calculations predict total altitude loss during the pull-up to include altitude lost due to: pilot reaction time, roll response time, altitude lost during the dive recovery (based on weight and drag, and if applicable, lack of available

gs)[56] [57]. EGC warnings are triggered by a simple predictive algorithm based on this predicted altitude loss and when the aircraft's current projected velocity vector will penetrate the set buffer above terrain [28]. EGC added an aural warning message of "PULL UP! PULL UP!" and a third visual warning was included: a heads-down break X on the Multi-Function Display (MFD) appears two seconds prior to the break X on the HUD and the new aural message [56].

Because of its possibly negative connotation, ground clobber was renamed Ground Avoidance Advisory Function (GAAF). All blocks of the F-16 are currently equipped with GAAF.

In 1982 the USAF recognized the need to develop a predictive ground proximity warning system (GPWS) for fighter and attack aircraft with the goal of reducing CFIT [56]. "The intent of TAF SON 311-82 [the 1982 Tactical Air Force Statement of Operational Need study] was to take advantage of current or near-term technology to provide a wider margin of safety, i.e., predictive ground proximity warning, for the low-level, high performance fighter and attack aircraft" [56]. In 1985, the USAF directed General Dynamics to conduct a GPWS alternatives and comparison study on the F-16. In March of 1987, General Dynamics published a report entitled "F-16 Ground Proximity Warning System Study" which fulfilled the USAF requirement for a GPWS study. The conclusion of this study was that "The most effective systems for preventing CFITs were the Digital Terrain System or the electronically scanned phase array radar." [56] This conclusion led to the advent of using the digital terrain based reference systems Predictive GCAS (PGCAS) and Automatic GCAS (Auto GCAS) on the F-16 [56].

Ground Proximity Warning System (GPWS)

Honeywell's Ground Proximity Warning System (GPWS) provides six types of alerting modes to aircrew to signal that the aircraft is on a collision course with the

ground, and GPWS is mandated to be installed on all US jet transport aircraft [42, 43]. The six alerting modes (which are also incorporated into Enhanced GPWS as well) are [42, 43]:

- Mode 1: Excessive Descent Rate
- Mode 2: Excessive Terrain Closure Rate
- Mode 3: Altitude Loss After Takeoff
- Mode 4: Unsafe Terrain Clearance
- Mode 5: Excessive Deviation Below Glideslope
- Mode 6: Advisory Callouts

In US FAA Circular AC23-18 published in 1974, the FAA mandated that all large turbine and turbojet commercial aircraft install GPWS [32]. According to Snow and Moroze in 1999, "The GPWS mandate reduced CFIT accidents from about 9 per year in the seven years immediately preceding the mandate to about 4 per year after. This rate has remained fairly constant" [61].

Because GPWS uses the radar altimeter as a primary sensor, GPWS has no forward-looking capability and can only protect the aircraft from collisions with terrain directly beneath the aircraft.

2.2.4.2 Manual Ground Collision Avoidance Systems that Use DTED.

This section covers Enhanced GPWS, the US Navy Terrain Awareness Warning System (TAWS) and TAWS II, and the five F-16 manual systems that are part of the Digital Terrain System.

Enhanced GPWS (EGPWS) or Terrain Awareness Warning System (TAWS)

Enhanced GPWS was a term coined by Honeywell who designed the first Enhanced GPWS (EGPWS) in 1980. According to Honeywell, "Honeywell pioneered the first Terrain Awareness and Warning System (TAWS) over 30 years ago. Today, we offer

Enhanced TAWS protection in our 'EGPWS' for thousands of aircraft of all types around the world. EGPWS is already credited with 30 documented aircraft saves, including the first helicopter/obstacle save with our MK XXII" [44].

Since 2000, EGPWS has become commonly known as Terrain Awareness Warning System (TAWS) due to the fact that the FAA used this more generic name in its US FAA Circular AC 25-23 published in March 2000. The FAA anticipated there would be multiple manufacturers of TAWS devices and did not want to use a brand name or give preference to Honeywell, the sole supplier of EGPWS at the time. "EGPWS includes all the same features as the current GPWS but also includes a predictive component. This predictive component would enable the EGPWS to provide more warning time – up to 60 seconds – in cases where impact into precipitous terrain is imminent" [61]. This forward-looking capability is made possible through EGPWS' internal databases, including a terrain database, an obstacle database, and airport runway data. In addition to the six GPWS alerting modes described in the previous section, EGPWS is able to combine this predictive component with its own internal databases to provide additional alerts and displays that provide "enhanced situational awareness and safety", hence the term *enhanced* GPWS [42].

In Honeywell's Aug 2011 EGPWS Pilot's Guide, Honeywell states, "The EGPWS uses aircraft inputs including geographic position, attitude, altitude, airspeed, and glideslope deviation. These are used with respect to internal terrain, obstacle, and airport databases to predict a potential conflict between the aircraft flight path and terrain or an obstacle. A conflict will result in the EGPWS providing a visual and audio caution or warning alert" [43]. While cautions and alerts are provided, EGPWS does *not* provide the pilot cues for how to perform a manual recovery to avoid the terrain.

US Navy Terrain Awareness Warning System (TAWS)

The US Navy has been developing TAWS since 1997 and testing TAWS on its aircraft since 2001. TAWS is currently installed on the Navy/Marine Corps' F/A-18 C/D/E/F Hornet and Super Hornet, EA-18G Growler, AV-8B Harrier, T-45C Goshawk trainer, and VH-60N Presidential Helicopter [4, 5, 65]. TAWS is the natural progressive improvement from the Navy-developed GPWS. The limitation of GPWS technology has always been that while it provides adequate look-down safety protection over level or descending terrain, it cannot provide forward-looking protection which is needed for rising terrain. GPWS has been in the Navy/Marine Corps fleet since 1997, however these early GPWS algorithms had nuisance warnings in certain flight regimes [16]. "These nuisance cues eroded pilot confidence and led to a general technique of disabling the system prior to takeoff" [16].

TAWS uses SRTM Level 1 DTED for the high-speed fighter and attack aircraft to provide the forward-looking capability not possible with radar altimeter alone. SRTM Level 2 DTED is used for TAWS fielded on helicopters such as the VH-60N. Even with the forward-looking capability that TAWS provides, the Navy Air Combat Electronics Program Office (PMA-209) who manages both GPWS and TAWS stresses that "GPWS/TAWS are safety-backup systems, NOT performance aids" [65]. In order to reduce nuisance warnings which troubled GPWS, TAWS computes two predicted trajectories, the Vertical Recovery Trajectory (VRT) and the Oblique Recovery Trajectory (ORT). The VRT is a roll to wings-level followed by a 5-g pull[13], while the ORT is a 5-g pull while maintaining the current bank angle [16]. If only one of two trajectories is predicted to intersect terrain, no visual or aural recovery warning will be given to the pilot since the second recovery is still available. However, once one trajectory (for example, the VRT) intersects for 4 frames or more (at a 10 Hz rate)

[13] 5-g pull or 80% of the maximum sustainable g for slower speed flight regimes [4].

and the second trajectory (in this example, the ORT) intersects for at least 1 frame, the TAWS algorithm will warn the pilot to execute the ORT.

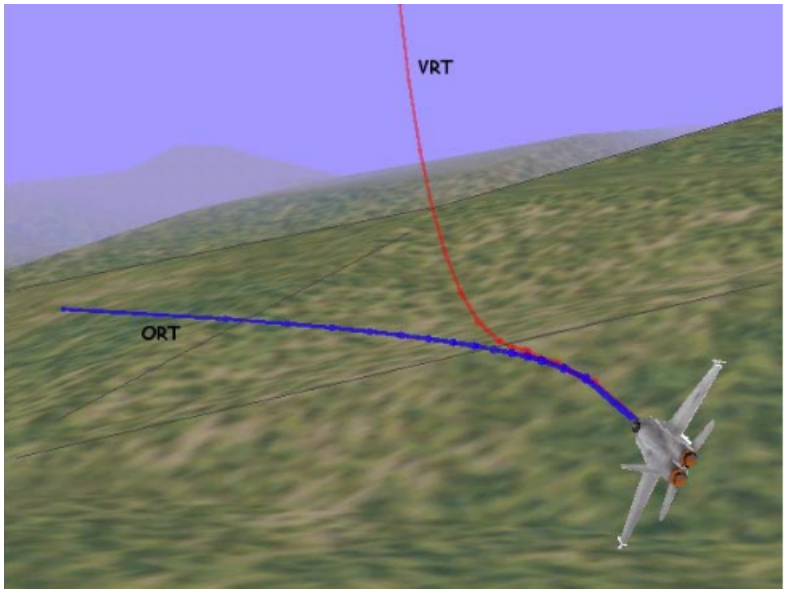


Figure 7. F-18 TAWS Vertical and Oblique Recovery Trajectories [16]

In Figure 7, the red line represents the VRT and the blue line represents the ORT. The VRT and ORT are each divided up into 'slices'. The VRT slices are illustrated as red dots and the ORT slices as blue dots in Figure 7. A slice is a point at which the SRTM DTED terrain elevation is compared to the predicted aircraft altitude from the VRT or ORT [4]. The VRT is always divided into 15 slices, while the number of slices for the ORT varies between 10 and 15 depending on bank angle when used with Level 1 DTED. Note that TAWS does not operate with Level 0 DTED because it is too coarse for any effective protection without creating nuisance warnings [4]. The VRT prediction ends once the aircraft reaches 90° nose up and the ORT prediction ends once a 90° heading change is achieved. The number of ORT slice points is bank angle dependent because the endpoint of the ORT is a 90° heading change. At low bank angles and roll rates, the ORT is longer; therefore closer to 15 slices would be used. At high bank angles and roll rates, the ORT is shorter; therefore closer to 10

slices would be used so that the slices do not bunch up on top of each other. In addition to providing directive aural warnings to the pilot, TAWS also provides the pilot directive visual cues for how to perform a manual recovery to avoid the terrain. "The most common combination would be a 'ROLL-RIGHT [or LEFT]!' followed by a 'PULL-UP!' voice message. The combination of the directive aural warning and HUD arrow is designed to provide the pilot with unambiguous information that allows for timely and appropriate responses to the warnings to avoid ground collision" [16].

As of October 2014, Navy TAWS has been credited with six documented saves in the F-18 community with anecdotal/undocumented evidence of saves in the AV-8B [4]. In addition, the Naval Safety Center has declared that the introduction of TAWS into the F-18 fleet has made a statistically significant reduction in CFIT mishaps in the F-18 community [4].

US Navy TAWS II

Figure 8 shows the build-up of protection from GPWS to TAWS to TAWS II.

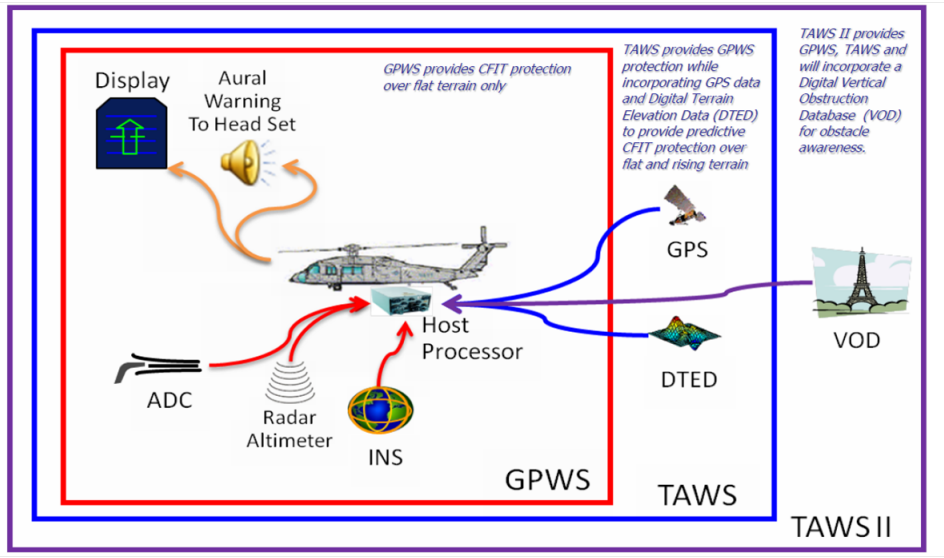


Figure 8. The US Navy's CFIT Avoidance Capability [65]

As explained previously, GPWS only provides CFIT protection over flat and descending terrain, while TAWS adds forward looking capability through the use of GPS and

DTED. TAWS II includes prior GPWS and TAWS capabilities and introduces the use of higher fidelity SRTM DTED Level 2 for terrain protection and a digital Vertical Obstruction Database (VOD) for obstacle protection [4]. NGA's VOD is the digital database for TAWS II obstacle protection [4]. TAWS II became a program of record in FY13 and simulator testing started in 2013 [4]. Additionally, the Navy has already incorporated SRTM DTED Level 2 for the rotary-wing TAWS effort currently under development [4, 5].

F-16 Manual Ground Collision Avoidance Systems that use DTED

– *Digital Terrain System (DTS)*

Digital Terrain System (DTS) was the first operational F-16 system to utilize DTED [23]. DTS was developed primarily to combat CFIT [23]. The five DTS functions are [23]:

1. Terrain Referenced Navigation (TRN)
2. Predictive Ground Collision Avoidance System (PGCAS)
3. Obstacle Warning and Cueing (OW&C)
4. Data Base Terrain Cueing (DBTC)
5. Passive Ranging (PR)

Briefly, TRN provides enhanced navigation accuracy using a Kalman filter, PGCAS provides all-attitude predictive ground collision avoidance advisories and warnings, OW&C provides advisories and warnings for man-made obstacles, DBTC provides a vertical steering cue to assist the pilot in clearing terrain at or near a preselected clearance height, and PR provides passive ranging inputs for weapon delivery computations [23]. Table 5 shows which DTS functions utilize DTED, the vertical obstruction database (VOD), or both [23].

USAF Block 40/50 F-16s have all of these DTS functions as do the USAF Block 25/30/32 F-16 aircraft. TRN, PGCAS, and OW&C are the functions that have

Table 5. Digital Terrain System Functions and Databases [23]

DTS Function	DTED	VOD
TRN	X	
PGCAS	X	X
OW&C		X
DBTC	X	X
PR	X	

demonstrated the most tactical utility [28]. The primary goal of DTS was to reduce the number of CFIT mishaps, and it is through PGCAS that it was designed to achieve that goal [23]. PGCAS is the most applicable to this research and is covered more in-depth below.

– *Predictive Ground Collision Avoidance System (PGCAS)*

PGCAS was developed in parallel with F-16 Auto GCAS during the mid-1980s. The most current collision avoidance software at the time on the F-16 was GAAF, however GAAF could only take into account terrain below the aircraft [23]. A demonstration system using TERPROM[14] digital terrain software was first flown on the General Dynamics F-16 #B2 aircraft in 1985 [23]. As a result of the 1987 F-16 GPWS study and the success of the TERPROM demo flights, DTS with PGCAS was recommended in order to take into account terrain in front of the aircraft [56].

PGCAS development started officially in 1989 as a combined USAF-General Dynamics (later Lockheed Martin) effort through the F-16 System Program Office (SPO). It should be noted that PGCAS and Auto GCAS were originally independent but parallel efforts. AFRL research on Auto GCAS started before the F-16 SPO started working on DTS/PGCAS. However, PGCAS was fielded first in the late 1990s. The original Auto GCAS algorithms came out of AFTI and matured over time to use similar functionality from the DTS/PGCAS implementation since Lockheed

[14]The DTS software is based on Terrain Profile Matching (TERPROM) software, a British Aerospace Systems and Equipment (BASE) product [23].

Martin was the contractor for both the DTS/PGCAS and Auto GCAS programs. Today the two programs complement each other to provide the best CFIT protection for the F-16. There were two phases of DTS flight tests at Edwards AFB, CA. 55 developmental test (DT) sorties were flown as part of DTS Phase 1 from 1995-1996, and 149 DT sorties were flown as part of DTS Phase 2 from 1996-1998 [28]. Of these sorties, approximately 60 were PGCAS flight tests [28].

The predicted recovery trajectory for PGCAS is a roll to wings level and 4-g pull [23]. When this predicted trajectory penetrates the pilot-selectable minimum terrain clearance (MTC) height, the PGCAS algorithm provides both visual and auditory advisories for the pilot to perform a manual recovery [57]. 1.1 seconds prior to the HUD break X, a flashing break X is displayed on both MFDs in case the pilot is heads-down in the cockpit [57]. A flashing break X and 'GND PROX' are then displayed in the HUD concurrent with a voice message warning of 'PULL UP! PULL UP!', indicating that the pilot must perform the PGCAS recovery maneuver immediately [57].

– *Pilot Activated Recovery System (PARS)*

PARS was developed concurrently with F-16 Auto GCAS as a flight test aid for a potentially disoriented test pilot. PARS is a manual initiation of the Auto GCAS recovery that was made available for safety reasons during Auto GCAS flight test, during which a PARS recovery could be activated by squeezing the trigger switch on the control stick [87]. Note that PARS is not terrain-aware like Auto GCAS. As previously discussed in Section 1.1, PARS was the system used for the 1997 F-16 Auto GCAS flight tests that determined the Time Available threshold of 1.5 seconds. For nose low conditions at the time of activation, the PARS recovery is identical to the Auto GCAS recovery of an automatic roll to wings-level and pull to 5gs. "PARS was initially conceived as a flight test aid to help address concerns that the distractions

of low-level night flying with the off-axis [Helmet-Mounted Display] HMD might lead to pilot disorientation. In fact, PARS has shown the potential to overcome the same problem in any operational situation where the pilot may come to doubt which way is up. With a single hands-on switch action, the pilot commanded PARS to perform an attitude recovery maneuver. The maneuver would take the aircraft from any arbitrary initial attitude to a wings level, slightly nose high attitude" [41].

The conclusions from Wright Laboratory's 1992 AFTI F-16 flight test report were that "The PARS was universally accepted by the AFTI/F-16 pilots as a useful aid for recovering the aircraft from unusual attitudes, and was in fact used in this capacity on several occasions. The algorithm is simple and straightforward and should be strongly considered for any application where pilot disorientation might occur" [41].

PARS is being fielded in 2014 as part of the Auto GCAS software M6.2+ combined with the modified Digital Flight Control Computer (DFLCC) software in all operational F-16 Block 40 and 50 aircraft to aid a pilot to manually recover from disorientation. PARS is triggered via the Advanced Modes Switch (as a press and latch to engage switch) on the left front glare shield avionics panel.

2.2.5 Automatic Ground Collision Avoidance Algorithms.

One very important difference between manual and automatic ground collision avoidance systems is the assumed state of consciousness of the pilot. GPWS, TAWS, and PGCAS and all manual variants assume the pilot is still aware and in the loop. Even though the pilot may be experiencing target-fixation or spatial-disorientation, these manual algorithms assume that the pilot is still aware, can hear aural warnings and see recovery cues and consequently respond to them with proper stick inputs. Auto GCAS does not assume the pilot is in the loop nor that the pilot can respond. Although Auto GCAS will safely recover a pilot who is target-fixated or

spatially-disoriented, Auto GCAS will also recover a pilot experiencing g-induced loss of consciousness (g-LOC) or a pilot otherwise incapacitated. The following conclusion from the 1993 AFTI/F-16 Automated Recovery System test report reveals another advantage of an automatic recovery system: "Also during this phase of flight test, the advantages of an automated flyup maneuver over a manual flyup became apparent. Since there was no need to compensate for pilot reaction time, the automated flyup maneuver was delayed which helped prevent nuisance flyups" [3]. This is due to the fact that an automated recovery such as Auto GCAS, (because it is a last-ditch rescue maneuver) is commanded after an aware pilot would normally recover the aircraft.

There are four automatic ground collision avoidance algorithms that have been developed, three of which have been flown. These four systems will be reviewed in the next section: 1) the USAF F-16 Auto GCAS, 2) the Swedish Gripen Auto GCAS, 3) the US Navy Automatic TAWS, and 4) NASA Small UAV Auto GCAS. The F-16 Auto GCAS, the Swedish Gripen Auto GCAS, and the NASA Small UAV Auto GCAS have been flight tested. As stated previously, the F-16 Auto GCAS was fielded in 2014. Note that none of these four algorithms use an optimal control solution technique; instead they all use some type of pre-planned maneuver or maneuvers.

2.2.5.1 USAF F-16 Auto GCAS.

The development of the current Auto GCAS algorithm can be traced back to the early flight tests of the Advanced Fighter Technology Integration (AFTI)/F-16 program. In 1984 the AFTI/F-16 program developed an automatic ground collision avoidance maneuver as a test safety aid for the advanced weapon delivery program called Automated Maneuvering Attack System (AMAS). "The [AMAS] system was used for automatic gun tracking of airborne targets and maneuvering bomb deliveries. The aircraft demonstrated the ability to accurately deliver unguided bombs using 5-g

curvilinear toss delivery at 200 feet above the ground" [30].

In order to ensure the test pilots would be safe during these low altitude bomb deliveries, an all-attitude automatic ground collision avoidance system was developed called the Automated Recovery System [30]. The AMAS testing was only accomplished for a partial F-16 envelope and assumed a flat Earth, hence, all the deliveries were over level terrain such as dry lakebeds. Thus, the genesis of the current Auto GCAS algorithm was as a test safety aid for these automated curvilinear toss bomb deliveries.

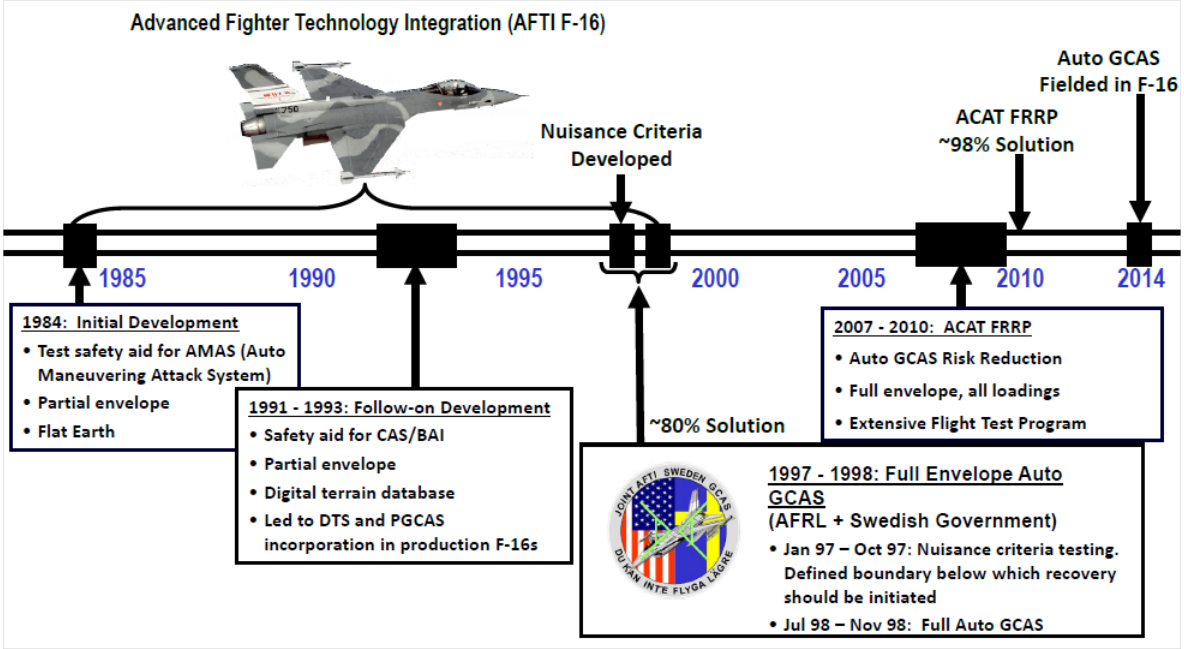


Figure 9. Auto GCAS Development Timeline [7, 45]

Figure 9 shows the Auto GCAS development timeline which spans the last three decades. The Automated Recovery System (ARS) flight testing for the Close Air Support (CAS) Block II/III phase on the AFTI/F-16 consisted of 118 flights that were conducted from 8 June 1991 to 14 January 1993 [3]. The motivation for the ground collision avoidance maneuver was as a safety aid for Close Air Support (CAS) and night attack operations testing. "The ARS consisted of three systems: the All-

Terrain Ground Collision Avoidance System (AT GCAS), the flat-earth GCAS, and the Pilot Activated Recovery System (PARS)" [3]. The ARS testing again included only a partial envelope, but now a digital terrain database was used instead of only the flat Earth assumption. "Automated weapon deliveries using the AFTI/F-16 AMAS bombing algorithm were completed at a variety of conditions. In this case, the bombing impact altitude was set so that upon weapons delivery, the aircraft would require a flyup to prevent a penetration of the [Minimum Clearance Distance] MCD. For all test runs the AT GCAS prevented the aircraft from penetrating the MCD during weapon release" [3]. Additional conclusions of the ARS testing were that "Both all-terrain and flat-earth GCAS prevented penetration at floor altitudes as low as 50 feet while the PARS recovered the aircraft from recognized instances of spatial disorientation. Because it was automated and integrated with a Terrain Referenced Navigation (TRN) system, the AT GCAS was highly complex and demonstrated excellent traits such as consistent recoveries and minimization of nuisance flyups" [3]. This testing of a digital terrain database integrated with a ground collision avoidance maneuver led to the incorporation of Digital Terrain System (DTS) and PGCAS into production F-16s in 2002.

Although there was no additional development of the Auto GCAS system from 1993-1997, the AFTI/F-16 project team continued to fly Auto GCAS sorties by supporting various mishap investigation boards and by supporting subsequent avionics testing that utilized automatic attack systems [87]. It was during this period that Auto GCAS was credited with its first save. "All of the [Automatic-Collision Avoidance] Auto-CA systems functioned reliably as designed even in the face of numerous unplanned failures. The most significant event during this period occurred in 1995 when an undetected failure caused the automated bombing system to initiate a 5-g barrel roll towards the ground from 500 feet above ground level. The Auto-GCAS

initiated a recovery from roughly 460 knots at 420 feet above ground level in a 13 degree dive at 115 degrees of bank while the bombing system was attempting to roll through the inverted at 120 degrees per second while pulling 5-gs of normal load factor. The Auto-GCAS bottomed the aircraft out at 290 feet above the ground. The pilot reacted to recover the aircraft 0.5 seconds after the Auto-CA system initiated the recovery. Post flight analysis based on flight test data determined that the 0.5 second delay in the pilots reaction would have resulted in an additional 475 feet of altitude loss or his recovery would have passed 185 feet below ground level. The Auto-GCAS had prevented the loss of the AFTI/F-16 pilot and aircraft" [87].

The Joint AFTI Sweden GCAS program was a cooperative effort between the Air Force Research Labs, NASA, and the Swedish government from 1997-1998. The Joint AFTI program had two phases. The first phase, January to October 1997, focused on nuisance criteria testing, and the second phase, July to November 1998, expanded to the full envelope Auto GCAS testing.

In 2007 the Defense Safety Oversight Council (DSOC) in the Office of the Secretary of Defense (OSD) funded the Automatic Collision Avoidance Technology/Fighter Risk Reduction Program (ACAT/FRRP) in response to the 2003 and 2007 SecDef mandates to reduce mishaps by 50% and 75%, respectively. ACAT/FRRP testing took place from 2007 to 2010 on a production F-16D Block 50 Model, tail number 90-840. 103 tests flights were conducted which spanned 141 total flight hours and included 1,670 automatic recoveries. The goal was to perform Auto GCAS risk reduction to see if it was viable for DoD fighter platforms, specifically the F-16 Blocks 40 and 50 and the F/A-18 E/F. Specifically, all F-16 CFIT accidents were put into nine categories [87]. As stated in Chapter I, a total of 48 flight tests were flown across the nine categories, and the team concluded that the current Auto GCAS algorithm would have saved the pilot and aircraft in 98% of those accidents [87]. The conclu-

sions of the ACAT/FRRP testing were that Auto GCAS provided excellent mishap prevention and nuisance-free operation [80]. The production Auto GCAS algorithm completed flight test in Mar 2013 at the 416^{th} Flight Test Squadron at Edwards AFB CA as part of Operational Flight Program (OFP) M6.2+ and was fielded on all US operational F-16 Block 40 and 50 aircraft in late 2014.

F-16 Auto GCAS is not limited to aircraft with digital flight control systems. Lockheed Martin and AFRL have developed an innovative and cost effective method to adapt Auto GCAS to F-16s with analog flight control systems by "adding redundant digital processor modules to the existing analog flight control computers of pre-block 40 F-16s" [17, 29]. The technology results in a hybrid analog/digital flight control architecture which is called Hy-Tech (Hybrid Technology) [40]. "This will not only allow Auto GCAS to be added to F-16s with analog flight control computers but demonstrates a technical path forward for incorporating Auto GCAS onto other older aircraft with analog flight control computers" [17]. The Hy-Tech program has successfully completed safety of flight testing in 2013 and initial demonstration flight testing in 2014 [29].

In addition to the F-16, Auto GCAS technology is also planned for the F-22 and the F-35. The Line-in-the-Sky (LIS) variant of Auto GCAS is currently being flight tested on the F-22 and is planned to be fielded in calendar year 2015 as part of Update 5. F-22 LIS is mechanized such that when a pilot-selected MSL floor is predicted to be penetrated, a roll to wings-level and a nominal 5-g pull recovery is executed by the autopilot. Auto GCAS is also planned to be fielded on the F-35 in Block 4 (otherwise known as Follow-On Development) no later than 2024. F-35 Auto GCAS shares the basic functionality of F-16 Auto GCAS where there is a single pre-planned roll to wings-level and 5-g pull automatic recovery.

2.2.5.2 Swedish Gripen Auto GCAS.

The Swedish Air Force shares the common history with the US for the 1997-1998 Joint AFTI Sweden GCAS testing conducted at Edwards AFB on the AFTI F-16. During this Joint testing (conducted with the USAF, NASA Dryden, and Lockheed Martin), nuisance criteria were developed and tested and full-envelope Auto GCAS testing was accomplished.

Saab has successfully integrated Auto GCAS on the JAS-39 C/D Gripen, and the final 25 Gripen Auto GCAS verifying flight tests were completed in Sweden from May to December 2013 [70]. The Auto GCAS functionality will be fielded to the Swedish Air Force JAS-39 C/D fleet in 2015 as part of the Mission System 20 (MS20) software block upgrade [51].

The JAS-39 C/D Gripen has the same basic Auto GCAS functionality as the F-16, where a predicted trajectory is constantly computed using a pre-planned roll to wings-level and 5-g pull recovery maneuver [70]. When this predicted trajectory intersects digital terrain, the autopilot commands this predicted maneuver as the automatic recovery. There is a small difference in the warning implementation between the Gripen and the F-16 [70]. In the Gripen Auto GCAS implementation, the pilot will get visual and aural warnings (arrows showing the direction to roll and pull are provided on both the heads-down and heads-up displays simultaneous with an aural 'PULL-UP! PULL-UP!' message followed by a tone) $\sim$1.5-2.5 seconds prior to the automatic recovery and is expected to manually command a wings-level roll and 5-g pull up [70]. If the pilot does not execute this recovery, then the Auto GCAS logic commands the automatic recovery [70]. In the F-16, the warning is not built into the Auto GCAS but rather in the Predictive GCAS which is a separate system. The Predictive GCAS provides the warnings to the pilot, and if no action is taken by the pilot, then the Auto GCAS commands the automatic recovery.

Auto GCAS will also be included in the new Mission System 21 (MS21) software for the next generation Gripen E/F which is currently under development and expected to be fielded to the Swedish Air Force in 2018 [70]. In December 2013, the Swedish government awarded a contract to Saab to convert 60 Gripen C to Gripen E aircraft [51, 86]. Saab is looking into the possibility of implementing different logic for very low-level flying in order to provide better CFIT protection for some specific cases [70]. The desire is to come up with a method for low-level flying that keeps the nuisance-free operation without sacrificing CFIT protection [70].

2.2.5.3 US Navy Automatic Terrain Advisory Warning System.

While the TAWS team understood it was first and foremost a passive system, from the inception of the TAWS planning and design they made sure to put hooks in the algorithm for an active TAWS system in the future if one was desired. In FY10, OSD gave money to the Navy and to the Air Force to put an automatic system on the F/A-18 E/F Super Hornet and F-16 C/D Block 40 and 50, respectively [98]. The verbiage from OSD stipulated that either implementing Auto GCAS or automating TAWS was a viable solution. The Navy Air Combat Electronics Program Office (PMA 209) went to work on automating their TAWS algorithm, called Automatic TAWS (ATAWS). However, the F-18 program office (PMA 265) analyzed the situation and determined that although ATAWS was a good idea they could not justify the return on investment of the significant cost and time that would need to be invested to validate and verify the software [4]. Allowing a flight control system to take control of an aircraft away from the pilot requires substantial verification. Their analysis was based on the fact that there have been no CFIT accidents in the F/A-18 E/F Super Hornet since 2004 when they finished Operational Test [4]. This can be attributed to the fact that the Super Hornet was born with TAWS as well as display improvements and anti-spin

control [4]. The F/A-18 E/F came off of the production line with the Tactical Aircraft Moving Map Capability (TAMMAC) which had the TAWS algorithm installed, while the F/A-18 C/D models had to be retrofitted with the TAMMAC in 2005 which is when they obtained TAWS. In addition, Hornet and Super Hornet aircrew have reported five documented CFIT saves that were attributed to GPWS and TAWS [4]. Based on the superior safety record of the Super Hornet, the Navy just could not justify the business case for ATAWS on the Super Hornet. While ATAWS made it to the planning phase, neither simulation nor flight test on ATAWS ever occurred.

2.2.5.4 NASA Small UAV Auto GCAS.

The goal of NASA's Small UAV Auto GCAS test was to demonstrate that the Auto GCAS algorithm could be hosted on an unmanned aircraft for possible future implementation on NASA's Ikhana testbed (similar to an MQ-9 Reaper) [84]. NASA also wanted to take the proven successes of the F-16 Auto GCAS algorithm and tackle the challenges of adapting it to another airframe.

The innovations that build on previous Auto GCAS testing include: 1) using a remotely piloted aircraft as the testbed (NASA chose to use the Dryden Remotely Operated Integrated Drone (DROID) as the testbed), 2) hosting the Auto GCAS algorithm on a smart phone which could either command the aircraft from the ground via telemetry (TM) or ride onboard the aircraft, 3) a new scan pattern called the 'cylindrical' (or 'beer can') scan pattern, 4) the inclusion of two lateral escape maneuvers in addition to the straight ahead maneuver, and hence an algorithm to choose the best maneuver out of the three possibilities, and 5) the compression of the digital elevation model (DEM) [83]. In addition to the single pre-planned recovery maneuver that the F-16 Auto GCAS had (i.e. roll to wings-level and pull), NASA's Small UAV program added a 40° left and 40° right turning escape as two additional maneuvers.

From January to May 2012, NASA flew 21 Auto GCAS sorties on the DROID with 208 test runs. 15 of the sorties were with the Android phone on the ground, controlling the DROID through TM, then 6 were flown with the Android phone on board the DROID [83]. Many of the 208 test runs were unusable due to TM dropouts, however there were 52 Auto GCAS automatic recoveries that were considered valid. Of these 52, the majority were lateral escapes. The breakdown was 24 left, two straight (roll to wings-level and pull), and 26 right recoveries [83]. The test team designed the system so that they had the option of turning the lateral escapes On or Off. When flown with all three escape options, there were no nuisance activations [83]. When flown with the left and right escape options disabled, there were multiple, highly disruptive nuisance activations [83]. This supports the Navy's similar conclusion from TAWS testing that including the Oblique Recovery Trajectory in addition to the Vertical Recovery Trajectory reduced nuisance warnings.

Section 2.2 provided a description of the most important systems involved in a ground collision avoidance system. An overview of 1) sensors, 2) terrain databases, 3) terrain modeling/approximation, and 4) both manual and automatic ground collision avoidance algorithm was presented. Four automatic ground collision avoidance algorithms were reviewed, which included 1) the USAF F-16 Auto GCAS, 2) the Swedish Gripen Auto GCAS, 3) the US Navy Automatic TAWS, and 4) NASA Small UAV Auto GCAS. The NASA Small UAV Auto GCAS used a 'Best of 3' approach by choosing and implementing the best of three pre-planned recovery maneuvers. However, none of these methods used an optimal control solution technique to solve the problem. Optimal control approaches are reviewed in the next section.

2.2.6 Optimal Control Approaches to Obstacle Avoidance Problems.

Optimal control solutions have been applied to aircraft obstacle avoidance problems with much success. Bollino, Lewis, Sekhavat, and Ross used direct orthogonal collocation (also called pseudospectral methods) with Legendre Gauss Lobatto (LGL) collocation points to generate a time-optimal trajectory for simulation of a UAV flying through an obstacle-rich urban environment [15] and extended the work to optimal trajectories for simultaneous air collision avoidance between multiple UAVs and for ground collision avoidance from the static urban obstacles [14]. Patel and Goulart's anti-hijack override system for commercial aircraft utilized a direct multiple shooting optimal control method that was demonstrated in simulation and generated feasible obstacle avoidance trajectories with computation times sufficiently fast for real-time implementation [67, 68]. The cost functional used was a minimum deviation from a reference trajectory. The purpose of their system was to prevent an aircraft that was in the hands of a hostile pilot from being flown into "buildings and other critical infrastructure and landmarks [68]." Patel and Goulart's research defined three trigger mechanisms to calculate when an override should be commanded and calculated the optimal trajectory to be flown [67]. The proposed Optimal Auto GCAS research differs from previous work in these ways: 1) the military mission of the aircraft requires low-level flying in close proximity to terrain, 2) the Auto GCAS must be nuisance-free which requires the recovery to be timely and aggressive (denoting a 'last-second' recovery) [80], and 3) the optimal control technique used is Legendre Gauss Radau (LGR) direct orthogonal collocation using the pseudospectral method [74]. The military mission is an important distinction. In a majority of civilian scenarios, a last-second aggressive automatic recovery would be an unacceptable risk. However, for Auto GCAS on a military platform, a last-second aggressive recovery is essential in order to be nuisance-free and still accomplish the operational mission.

Applying optimal control to obstacle or terrain avoidance is not new, but mapping the last-second timely and aggressive recovery to an optimal control formulation is new. F-16 Auto GCAS does address the last-second issue through its aggressive and timely recovery but it is a pre-planned recovery vice an optimal recovery and was not designed for climb-limited or g-limited aircraft. Thus, there is no existing operational Auto GCAS system that uses optimal control for nuisance-free operation via a last-second recovery. This research seeks to address that gap.

2.2.7 Real-Time Optimal Control.

For an operational system, the optimal control calculations will have to be done onboard and in real-time. Current research demonstrates that direct optimal control methods show promise for real-time implementation. Patel and Goulart conducted online optimization simulations using a Boeing 767 simulator and the nonlinear programming (NLP) solver IPOPT [12] to generate their obstacle avoidance trajectories in 2010-2011 [67, 68]. In 2008, Bollino and Lewis used DIDO [77] and the NLP solver SNOPT [39] to generate min-time trajectories for multiple UAVs flying through an urban environment and even incorporated real-time automatic route re-planning to account for a new pop-up target to intercept [15, 14]. Two other recent examples of real-time optimal control programs that have been demonstrated in flight test are Ross, Cobb, and Baker's quadrotor optimal control research at AFIT in 2011 [78] and Geiger et al.'s real-time UAV direct collocation path planner at the Pennsylvania State University in 2007 [37, 38]. AFIT's quadrotor research solved a bearing-only trajectory planning problem to autonomously land a quadrotor on a wire [78]. Penn State's UAV research planned a path to maximize surveillance time of a target by a camera onboard a UAV [37, 38].

Although the real-time implementation aspect is not the focus of this research, it

is worthwhile to note that it is feasible that the algorithms being formulated for this research could be employed onboard a manned aircraft in the near future.

This research proposes an optimal control solution technique applied to the Auto GCAS problem, and therefore, some background in the optimal control problem formulation is provided in the next section.

2.3 Optimal Control

Now that a survey of collision avoidance methods and a literature review of operational manual and automatic ground collision avoidance systems has been presented along with research in optimal control solutions for collision avoidance, it is necessary to provide background information on optimal control which this research uses as the solution technique.

2.3.1 General Form of the Optimal Control Problem.

The goal of an optimal control problem is to find a control $\mathbf{u}^* \in \mathbf{A}$ "which causes the system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t)$ to follow a trajectory $\mathbf{x}^* \in \mathbf{X}$ that minimizes the performance measure J" where $\mathbf{A}$ is the admissible set of control inputs and $\mathbf{X}$ is the admissible set of state values [53]. The general optimal control problem is stated as [53] [69]

$$\underset{\mathbf{u}\in\mathbf{A}}{\text{Minimize}}\ J = \phi(\mathbf{x}(t_0), t_0, \mathbf{x}(t_f), t_f) + \int_{t_0}^{t_f} L(\mathbf{x}(t), \mathbf{u}(t), t)dt \tag{1}$$

Subject to:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t) \quad \text{(EOM/dynamic constraints)} \tag{2}$$

$$\psi(\mathbf{x}(t_0), t_0, \mathbf{x}(t_f), t_f) = 0 \quad \text{(boundary constraints)} \tag{3}$$

$$\mathbf{C}(\mathbf{x}(t), \mathbf{u}(t), t) \leq 0 \quad \text{(algebraic path constraints)} \tag{4}$$

where ϕ is the Mayer term that contains the terminal penalty (also known as the final cost), L is the Lagrangian term that contains the integral cost (also known as the running cost or along the way cost), $\dot{\mathbf{x}}$ represents the equations of motion or dynamics, $\mathbf{x}$ represents the states, $\mathbf{u}$ is the control, ψ represents the boundary constraints, and $\mathbf{C}$ represents the algebraic path constraints.

2.3.2 Performance Measures.

A performance measure, or performance index, is what is used to evaluate the performance of the system of interest [53]. The goal is to choose an optimal control, $\mathbf{u}^*$ that will minimize (or maximize) the performance measure. In a military scenario with ground-to-air threats, an optimal control problem could be to fly an aircraft from the home base, take reconnaissance photos and videos of a certain point in enemy territory and then return home. The performance index for the aircraft's flight path could be to maximize the distance away from enemy ground radars and surface-to-air missile sites. A performance measure is represented by the cost functional, J, in Equation (1). Five performance measures for some common optimal control problems are described here. Constraints exist for each that are specific to the user's goal, but for now only the different possible forms of the cost functional are being discussed in this chapter. The next sections will describe the following five potential performance measures: 1) minimize time, 2) minimize terminal state, 3) minimize control effort, 4) minimize deviation from a path with minimum control effort, and 5) minimize deviation from the origin with minimum control effort [53]. Chapter III will present the cost functionals employed in this research.

2.3.2.1 Minimize Time.

Minimum time solutions are useful for a variety of applications. For example, a min time solution makes sense for navigational applications where the vehicle needs to move to another location or state in the quickest time. Using our scenario of an aircraft flying through a high-threat area, minimum time could be a viable performance measure such that the time the aircraft is vulnerable to threats is minimized. The minimum time performance measure is

$$J = \int_{t_0}^{t_f} 1dt \quad \text{where } [\phi = 0, L = 1]$$

or equivalently,

$$J = t_f - t_0 \quad \text{where } [\phi = t_f - t_0, L = 0] .$$

Note that in the first formulation of the min time problem, there is only a Lagrangian term, while in the second formulation there is only a Mayer term. The min time formulation is a simple example to show that Lagrangian terms, once integrated, can become Mayer terms.

2.3.2.2 Minimize Terminal State.

The goal of this performance measure is to "minimize the deviation of the final state of a system from its desired value $\mathbf{r}(t_f)$ [53]." Kirk also refers to this as a Terminal Control problem. An example is a weapon that must arrive and detonate at a specific final (x, y, z) location in space. The deviation from this final location is the 'final cost,' but where the weapon goes along the way is not important. The

performance measure to minimize the terminal state is

$$J = [\mathbf{x}(t_f) - \mathbf{r}(t_f)]^T \mathbf{H} [\mathbf{x}(t_f) - \mathbf{r}(t_f)] \quad \text{where } \mathbf{H} \geq 0$$

where $\mathbf{H}$ is called the final state weighting matrix. $\mathbf{H}$ is a real symmetric positive semi-definite $n \times n$ weighting matrix, where n is the number of states, i.e. $\mathbf{x}, \mathbf{r} \in \mathbb{R}^n$. The semi-definite requirement on $\mathbf{H}$ permits penalizing some but not all elements in $[\mathbf{x}(t_f) - \mathbf{r}(t_f)]$. Even though some states can have a zero weighting, at least one of the states must be weighted or else the cost functional is pointless. In the cost functional above, there is no running cost, but a running cost can be added if desired. For example, this performance index can also be combined with a running cost of min control effort as described next. This performance index can also be combined with a running cost of min control effort as described next.

2.3.2.3 Minimize Control Effort.

The goal of min control is "to transfer a system from an arbitrary initial state $\mathbf{x}(t_o) = \mathbf{x}_0$ to a specified target set with a minimum expenditure of control effort" [53]. If there are many controls, such as in this research where aircraft bank angle ϕ and load factor N_z are all possible controls, then $\mathbf{u}$ is a vector. The min control effort performance measure is

$$J = \int_{t_0}^{t_f} \left(\mathbf{u}^T(t) \mathbf{R} \mathbf{u}(t) \right) dt \quad \text{where } \mathbf{R} > 0$$

where $\mathbf{R}$ is called the control weighting matrix and is chosen as a real symmetric positive definite $m \times m$ weighting matrix, where m is the number of controls. $\mathbf{R}$ is positive definite because if any one of the controls had a zero weighting, then that would imply infinite control could be used which is not physically realizable. Thus, the

positive-definiteness of $\mathbf{R}$ ensures that any control used in the performance measure of a system's optimal control problem must have a positive penalty associated with it. Technically, $\mathbf{R}$ could be equal to zero but then $\mathbf{u}$ must be bounded in the constraint function.

2.3.2.4 Minimize Deviation from a Path with Minimum Control Effort.

Combining the above with a desire to minimize path deviations, creates Minimum Deviation from a Path problems (or what Kirk also refers to as Tracking problems), where the goal is "to maintain the system state $\mathbf{x}(t)$ as close as possible to the desired state $\mathbf{r}(t)$ in the interval $[t_0, t_f]$" [53]. The minimum deviation from a path with minimum control effort performance measure is

$$J = \int_{t_0}^{t_f} \Big((\mathbf{x}(t) - \mathbf{r}(t))^T \mathbf{Q} (\mathbf{x}(t) - \mathbf{r}(t)) + \mathbf{u}^T(t) \mathbf{R} \mathbf{u}(t) \Big) \, \mathrm{d}t, \, \mathbf{Q} \geq 0, \mathbf{R} > 0$$

where $\mathbf{Q}$ is called the state weighting matrix and is a real symmetric positive semi-definite $n \times n$ weighting matrix and with $\mathbf{R}$ as previously defined. The semi-definite requirement on $\mathbf{Q}$ means not all elements in $[\mathbf{x}(t) - \mathbf{r}(t)]$ must be penalized.

This is by no means an exhaustive list of performance measures, but it does represent the most commonly used ones. This research desires an optimal control solution, therefore the question of which J is 'best' for Auto GCAS needs to be answered. Chapter III will develop the use and/or modification of these cost functionals as appropriate for Auto GCAS.

2.3.3 Solution Methods.

2.3.3.1 Direct vs. Indirect Approaches.

With the indirect method, the optimality conditions are first applied in order to arrive at a Hamilton Boundary Value Problem (HBVP) which is more difficult to solve. The constrained problem is then turned into an unconstrained problem and discretized. In the direct method, first the problem is discretized using pseudospectral methods. Thus, the problem is now transcribed to a parameter optimization problem. The constrained problem is then turned into an unconstrained problem, and handed off to an NLP solver for optimization. It is named direct because the method is to directly solve for the optimal control u^* first then to solve for the co-states λ, while with the indirect method, one solves for the co-states λ first and then solves for the optimal control u^*. Hence, the control u^* is solved for indirectly.

According to the Gauss Pseudospectral Costate Mapping Theorem, "The Karush-Kuhn-Tucker (KKT) conditions of the NLP are exactly equivalent to the discretized form of the continuous first-order necessary conditions of the continuous Bolza problem when using the Gauss psuedospectral discretization" [75]. The solution path for both the direct and indirect methods is shown in Figure 10 [75] with the Costate Mapping depicted in the bottom right-hand corner.

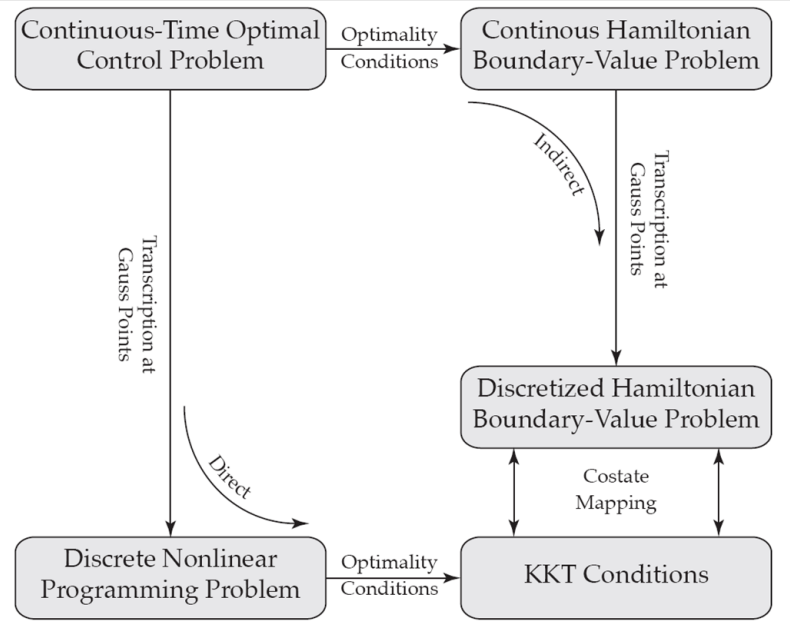


Figure 10. Equivalence of Indirect and Direct Forms Using the Gauss Pseudospectral Discretization [75]

According to Rao et al., "One of the key features of the Gauss pseudospectral method is the ability to map the KKT multipliers of the NLP to the costates of the continuous-time optimal control problem" [75].

Benson states that "The discretization of the optimal control problem at Gauss points has no defects, so that the direct solution is exactly equivalent to the indirect solution. This result means that the operations of discretization and optimization commute, and the solution is the same whether you discretize then optimize, or optimize then discretize" [9]. In the direct method we discretize then optimize, while with the indirect method, we optimize then discretize. Detailed discussions and proofs regarding the direct and indirect methods can be found in Benson's and Huntington's research [9, 47] as well as the Gauss Pseudospectral Optimization Program (PSCOL) manual [75].

2.3.3.2 Pseudospectral Method.

In recent years direct transcription methods have become very popular in solving complex optimal control problems in many areas such as aerospace, chemical, and medical fields. According to Huntington, these direct orthogonal collocation (DOC) methods are more commonly known as pseudospectral methods in the aerospace community [47]. The solution method used for this research is the pseudospectral method (PSM), also referred to as the Variable-Order Gaussian Quadrature Collocation Method by Rao [69]. The PSM and the accompanying Legendre-Gauss-Radau collocation rule which will be used for this research are summarized here.

The PSM is a direct method for solving optimal control problems using Gaussian quadrature. The general formulation of the optimal control problem was given in Equations (1) to (4) in Section 2.3.1. An optimal control problem has two main tasks that must be solved. One task is to evaluate the integral in the cost functional, and the second is to solve the differential state equations, i.e. dynamic constraints. The PSM is a numerical method that solves partial differential equations. PSM uses Gaussian quadrature to perform the integration and enforces the differential constraints to be satisfied at the collocation points.

Collocation methods are chosen because they are computationally efficient and robust to initial guesses [73]. As shown by Rao, using collocation leads to a large sparse nonlinear programming problem (NLP) [73]. The Lagrange interpolating polynomials are chosen as the basis functions used to approximate the states $\mathbf{x}$ and controls $\mathbf{u}$. Lagrange polynomials also have a special quality called the Isolation Property, leading to sparse matrix equations. This sparsity pattern makes the iteration method very efficient. Current NLP solvers were designed to take advantage of this sparsity which is why using collocation methods leads to computationally efficient solutions [73].

In order to perform the first task of numerically evaluating the integral in the

cost functional J, Legendre-Gaussian quadrature is used where quadrature points are chosen as the roots of the Legendre polynomials. In order to use the roots of the Legendre polynomials (which are defined on the interval $[-1, 1]$), an affine transformation must be done from the time domain, where t $\in [t_0, t_f]$ to the tau domain, where $\tau \in [-1, 1]$ [48]

$$\tau = \frac{2t}{t_f - t_0} - \frac{t_f + t_0}{t_f - t_0}. \tag{5}$$

Taking the derivative of Equation (5),

$$\frac{\mathrm{d}\tau}{\mathrm{d}t} = \frac{2}{t_f - t_0} \tag{6}$$

which can be rearranged

$$\frac{\mathrm{d}t}{\mathrm{d}\tau} = \frac{t_f - t_0}{2} \tag{7}$$

or equivalently

$$\mathrm{d}t = \frac{1}{2}(t_f - t_0)\mathrm{d}\tau. \tag{8}$$

There are three main collocation methods which are the Legendre-Gauss (LG) method, the Legendre-Gauss-Radau (LGR) method, and the Legendre-Gauss-Lobatto (LGL) method. Figure 11 shows a graphical example of how collocation points and endpoints (if applicable) for each of the three methods are placed [73]. Note that in all methods the Gauss collocation points are interior to $\tau \in [-1, 1]$ and dense near the endpoints.

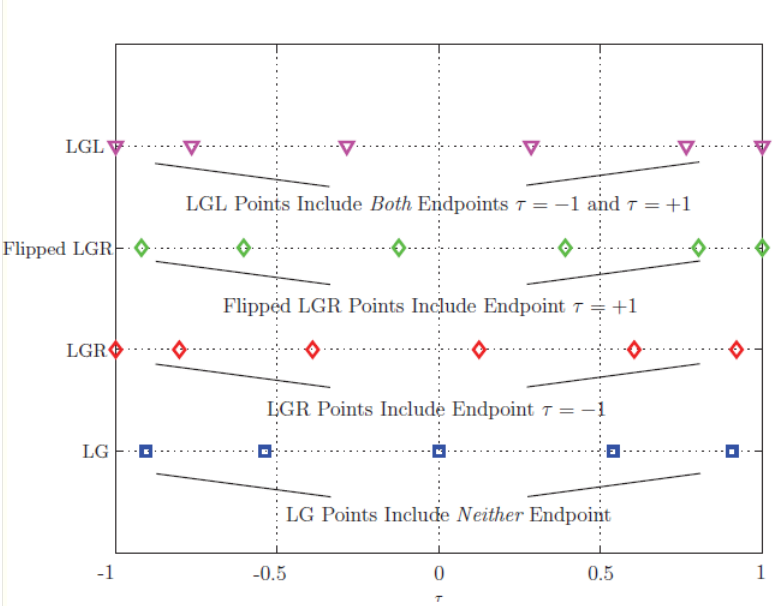


Figure 11. Schematic of Gaussian Quadrature Points [73]

In Figure 11, the blue squares indicate the LG method spacing which includes the collocation points strictly on the open interval $\tau \in (-1, 1)$, and therefore does not include either endpoint. The red diamonds show the LGR method which is on the half-open interval $\tau \in [-1, 1)$ which includes the initial point plus the interior collocation points, while the green diamonds are simply the 'Flipped LGR' which is on the half-open interval $\tau \in (-1, 1]$ where the final point is included vice the initial point. Finally, the purple triangles represent the LGL method which is on the closed interval $\tau \in [-1, 1]$ and hence includes the interior collocation points plus both endpoints [73].

All three of these Gaussian quadrature methods are extremely accurate [48, 73]. Accuracy is defined as the highest degree polynomial that can be integrated exactly. LG quadrature can exactly integrate a polynomial of degree $2n$-1, where n is the number of collocation points. For example, for $n = 2$, LG quadrature will be accurate for cubics, and for $n = 4$ it would be accurate for polynomials of degree 7 or less. LGR quadrature can exactly integrate a polynomial of degree $2n$-2, and LGL quadrature

can exactly integrate a polynomial of degree $2n$-3.

While numerous methods exist in the literature for numerical integration, the LGR integration rule is used for this research because it is considered the best way to incorporate an initial condition. For problems where the initial state $\mathbf{x}_0$ is specified such as in the Auto GCAS problem where the initial aircraft condition (states) is known, the LGR method works well. To accurately evaluate the integral using the LGR rule, only the values of the states $\mathbf{x}$ and the control $\mathbf{u}$ are needed at the quadrature points in τ. These states $\mathbf{x}$ and the controls $\mathbf{u}$ at the τ quadrature points represent the unknowns in the transcribed NLP problem.

The second task in solving the optimal control problem is solving the continuous differential state equations. Due to the computational burden of solving a continuous-time problem numerically, the continuous problem is transformed into a discrete approximation. Therefore, the control $\mathbf{u}$ and states $\mathbf{x}$ need to be discretized. The unknown functions are approximated as linear combinations of orthogonal polynomials. The functions are approximated using orthogonal polynomials because polynomials are easy to numerically integrate and differentiate. Taking advantage of the Isolation Property of Lagrange polynomials, a Lagrange Interpolating Polynomial is used as the approximating function and the state equations (dynamics) are integrated to get discretized values of the states. The state vector $\mathbf{x}$ is now a discretized equation, that is, $\mathbf{x}$ is a function of the control vector $\mathbf{u}$ sampled at our collocation points.

The cost functional is evaluated using discrete values for states $\mathbf{x}$ and control $\mathbf{u}$, and the constraints are static instead of dynamic ones. The PSM produces exact results at special points, called the nodes or collocation points. Since the discrete solution at the collocation points is known to be exact, a continuous solution can be obtained by interpolating between them. The final goal is still to solve for the optimal control $\mathbf{u}^*(t)$ that will minimize the cost functional J, but the problem has

been transformed into a static optimization problem that can be solved with a nonlinear programming (NLP) solver like SNOPT [39] or IPOPT [12]. For this research, a general-purpose MatLab software program called GPOPS-II from the University of Florida [74] is used to solve the optimal control problems. GPOPS-II can utilize multiple NLP solvers, including quasi-Newton (first derivative) solvers such as SNOPT [39] and full Newton (second derivative) solvers such as IPOPT [12] and KNITRO [20, 69]. IPOPT is the NLP solver used for the research herein.

2.4 3-D Aircraft Model

To achieve the research objectives, a full 3-D environment which includes an aircraft model and digital terrain elevation data (DTED) are needed.

A nonlinear three degree-of-freedom (3-DOF) point mass model is used in this research similar to that of Ragunthan [72], Bicchi [11], and Smith [81]. This is a common dynamics model used in the literature for aircraft simulation when both computational speed and realistic flight dynamics are important, but when a particular airframe is not specified/needed. The 3-DOF point mass model EOM are shown in Equations (9) to (14) where x, y, and z are aircraft position, V is the airspeed of the aircraft, γ is flight path angle, χ is heading angle, and v_w is wind speed in the x, y, and z directions. Winds are defined as the direction they are coming from, for example, $v_{w_z} > 0$ for an updraft. In addition, T is thrust, L is lift, D is drag, M is

mass, g is the acceleration of gravity, α is angle of attack, and μ is bank angle.

$$\dot{x} = V\cos\gamma\cos\chi + v_{w_x} \tag{9}$$

$$\dot{y} = V\cos\gamma\sin\chi + v_{w_y} \tag{10}$$

$$\dot{z} = V\sin\gamma + v_{w_z} \tag{11}$$

$$\dot{V} = \frac{T\cos\alpha - D - Mg\sin\gamma}{M} \tag{12}$$

$$\dot{\gamma} = \frac{(T\sin\alpha + L)\cos\mu - Mg\cos\gamma}{MV} \tag{13}$$

$$\dot{\chi} = \frac{(T\sin\alpha + L)\sin\mu}{MV\cos\gamma}. \tag{14}$$

A further simplification is made using 3-DOF approximations to eliminate T, D, M, g, and α. Velocity is assumed to be constant due to the relatively short duration of an Auto GCAS recovery (a few seconds), resulting in

$$T\cos\alpha - D = Mg\sin\gamma. \tag{15}$$

Normal acceleration (or load factor) N_z is defined in the velocity-axis frame, resulting in

$$MN_z = T\sin\alpha + L. \tag{16}$$

Substituting Equations (15) and (16) into Equations (12)-(14), eliminates Equation (12) and simplifies Equations (13) and (14). Hence, the simplified 3-DOF point

mass model equations of motion used for this research are

$$\dot{x} = V \cos\gamma \cos\chi + v_{w_x} \tag{17}$$

$$\dot{y} = V \cos\gamma \sin\chi + v_{w_y} \tag{18}$$

$$\dot{z} = V \sin\gamma + v_{w_z} \tag{19}$$

$$\dot{\gamma} = \frac{N_z \cos\mu - g\cos\gamma}{V} \tag{20}$$

$$\dot{\chi} = \frac{N_z \sin\mu}{V\cos\gamma}. \tag{21}$$

where the five states $\mathbf{x}$ are defined as aircraft x-position, y-position, z-position, flight path angle γ, and heading angle χ. The two controls $\mathbf{u}$ are μ (bank angle) and N_z (normal acceleration or load factor).

$$\mathbf{x} = \begin{bmatrix} x \\ y \\ z \\ \gamma \\ \chi \end{bmatrix} \quad \text{and} \quad \mathbf{u} = \begin{bmatrix} \mu \\ N_z \end{bmatrix} \tag{22}$$

2.5 3-D Digital Terrain Model

SRTM DTED provides the terrain height at each lat/lon point in the specified grid and an accompanying reference vector that contains three descriptive elements: cells/degree, the Northern latitude corner, and the Western longitude corner. A cell is defined as the width between two SRTM DTED posts. Please refer to Appendix B on how to work with SRTM DTED data for this research.

2.6 East-North-Up Reference Frame

The aircraft states must be propagated in a reference frame that is compatible with the aircraft equations of motion. A common reference frame in the literature used for aircraft simulation is the East-North-Up (ENU) frame, where East is the x-axis, North is the y-axis, and Up is the z-axis. For navigation purposes, "(east, north, up) coordinates are essential in determining the line of sight for terrain data given as latitude, longitude and height, such as digital terrain elevation data (DTED)" [27, 91]. The ENU reference frame overlay is shown in Figure 12.

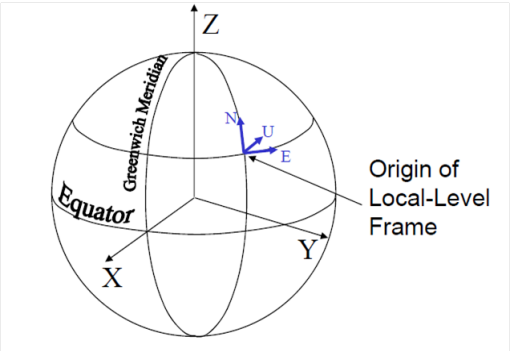


Figure 12. East-North-Up Reference Frame Overlay [76]

In the z-axis (Up) of the ENU frame, aircraft altitude and terrain heights are shown in positive numbers which is desirable for an Auto GCAS system. Therefore, the ENU frame is most appropriate for this research.

2.7 Summary

This chapter presented a survey of existing CDR methods for collision avoidance, a literature review of manual and automatic ground collision avoidance systems, the general form of the optimal control problem, the 3-D aircraft model, the 3-D digital terrain model, and the reference frame used for this research. Next, Chapter III lays out how the optimal control approach will be applied to the Auto GCAS problem.

III. Methodology

Chapter I motivated the need to solve the problem of Controlled Flight Into Terrain (CFIT) and showed that the best way to accomplish any further reduction in CFIT mishaps is to implement an automatic solution such as the Automatic Ground Collision Avoidance System (Auto GCAS). The F-16 Auto GCAS and requirements were introduced, and then the research objectives were outlined: 1) develop an optimal control technique to calculate an Auto GCAS recovery trajectory for aircraft with lower climb performance, and 2) conduct a limited comparison of the Optimal Auto GCAS to an Auto GCAS algorithm with pre-planned trajectories.

Chapter II described a taxonomy with which to classify air and ground collision avoidance methods and provided a literature review of pertinent manual and automatic ground collision avoidance systems. The literature review identified a gap and showed that optimal control solution methods have not been applied to the Auto GCAS problem with a last-second recovery. This chapter develops an optimal control method as a solution technique for implementing Auto GCAS on aircraft. As a precursor to the research methodology, a description of the general form of the optimal control problem formulation was presented in Chapter II along with example performance measures. Chapter II concluded with an explanation of the aircraft model, digital terrain model, and reference frame.

In this chapter, the specific methodology of applying optimal control theory to the Auto GCAS problem is explained. First, the Auto GCAS design requirements on the automatic recovery maneuver are explained. Next, two different optimal control formulations of the Auto GCAS problem proposed for this research are discussed. Finally, results for both optimal control formulations are presented for a 2-D example problem, enabling a direct comparison between the two formulations.

3.1 Definition of *Aggressive* and *Timely* Requirements in Relation to Nuisance Warnings

Conceptually, the *aggressive* and *timely* requirements are described below, followed by mathematical definitions in Section 3.2. In order to reduce nuisances and not interfere with the military mission, an Auto GCAS recovery maneuver must be both aggressive and timely [80, 87]. In Figure 13 below, the red dotted line shows a user-defined keep-out zone (or buffer) that is added to the digital terrain. The blue dashed line represents an aggressive but not timely recovery maneuver, while the green dashed line represents a timely but not aggressive recovery maneuver [80]. The solid orange line represents a recovery that is both aggressive and timely and hence meets both Auto GCAS requirements [80].

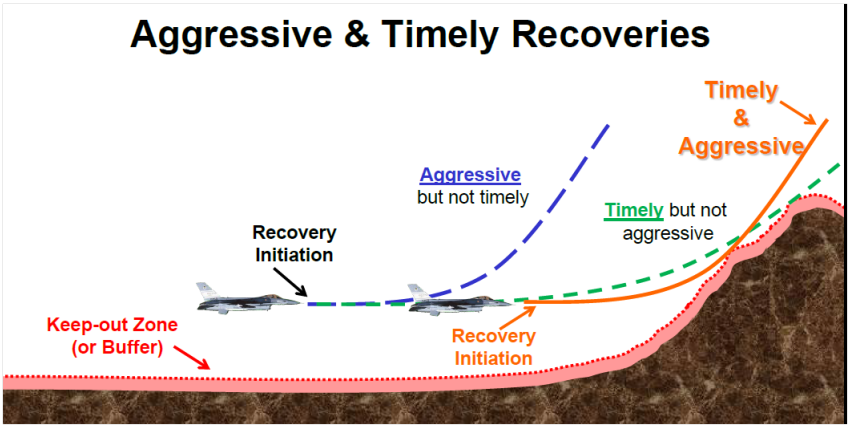


Figure 13. Conceptual Diagram of Aggressive, Timely, and Aggressive and Timely Recoveries [80]

If an automatic recovery is aggressive but commanded too early, it can be judged as a nuisance by a situationally aware pilot. If an automatic recovery is timely but is not aggressive, it can likewise be judged as a nuisance by a situationally aware pilot [80, 87]. Hence, an Auto GCAS recovery maneuver must be both aggressive and timely [80, 87].

The current F-16 Auto GCAS recovery was designed to meet both requirements.

F-16 Auto GCAS constantly calculates a single predictive trajectory with a wings-level 5-g pull-up (making it aggressive like the blue dashed line) with emphasis on executing the maneuver after the 1.5 sec Time Available threshold when an aware pilot would recover (making it timely and aggressive like the solid orange line). For this research, two optimal control solution techniques have been formulated to create an aggressive and timely Auto GCAS recovery for climb- and g-limited aircraft.

While Figure 13 shows a spatial view of the problem, the times related to each event in any recovery must be defined. Figure 14 provides a generic timeline which is applicable to both Auto GCAS optimal control solution techniques. The timeline shown assumes there is terrain that must be avoided within the time horizon used by the optimizer. The optimizer runs continuously from takeoff time $t_{Start\ Mission}$ to landing time $t_{End\ Mission}$. The beginning of the time horizon is t_0 and the end of the time horizon is t_f, while $t_{trigger}$ is defined as the start of the auto recovery and t_{CPA} is the time at the closest point of approach (CPA) to terrain. The trigger time $t_{trigger}$ and t_{CPA} both lie in the interval $[t_0, t_f]$. The Auto GCAS recovery shown in red is commanded by the autopilot for $t \in [t_{trigger}, t_{CPA})$, and control is given back to the pilot just after t_{CPA}, once the algorithm assures that the aircraft will clear terrain. A timely recovery is one in which $t_{trigger}$ is as close to t_{CPA} as possible while still allowing the aircraft to clear the terrain.

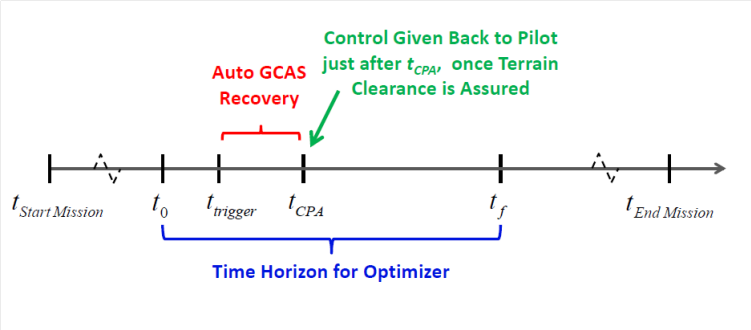


Figure 14. Auto GCAS Optimal Control Maneuver Timeline

3.2 Mathematical Definition of *Aggressive* and *Timely* Requirements

The Auto GCAS requirements of *aggressive* and *timely* are defined mathematically here so that these requirements can be captured in an optimal control formulation. For aircraft, it seems natural to define aggressive as being mapped to control input. We first define a set of functions based on aircraft state and control variables that represent a measure of the aircraft's 'control state'. These functions map the physical control inputs to the aircraft control state. For either safety or physical reasons, each of these functions are bounded such that

$$U_{\min_i} \leq \ U_i(\mathbf{x}(t), \mathbf{u}(t)) \ \leq U_{\max_i} \tag{23}$$

where $U_{\min_i}$ and $U_{\max_i}$ are user-specified limits for the i^{th} control state constraint (where $i = 1,\ 2, ...n$ is the number of control states). Examples of a control state constraint might be a max structural g limit on the aircraft or a min or max g physiological limit on the pilot. Each of these limits would be a functional of the state of the aircraft and the current aircraft control inputs. First, define aggressive to be

$$\left.\begin{aligned} U_i(\mathbf{x}(t), \mathbf{u}(t)) \ &= U_{\max_i} \\ \text{or} \qquad & \\ U_i(\mathbf{x}(t), \mathbf{u}(t)) \ &= U_{\min_i} \end{aligned}\right\} \text{for at least one } U_i,\ \forall\ t \in [t_0, t_{CPA}) \tag{24}$$

Here the control state $U_i(\mathbf{x}(t), \mathbf{u}(t))$ can be a functional of a combination of states and controls. For simplicity, in this research we consider only the case where the control state is directly mapped to a single control input. For example, a min or max control surface deflection limit that is directly mapped to a control stick input. Thus,

$$U_i(\mathbf{x}(t), \mathbf{u}(t)) = u_i(t) \tag{25}$$

Using Equation (25) and aggressive as defined in Equation (24), Equations (23) and (24) reduce to Equation (26)

$$\left.\begin{array}{rl} u_i(t) & = u_{\max_i} \\ \text{or} & \\ u_i(t) & = u_{\min_i} \end{array}\right\} \text{for at least one control } u_i,\ \forall\ t \in [t_0, t_{CPA}). \tag{26}$$

The time at the closest point of approach t_{CPA} lies in the interval $[t_0, t_f]$ and represents the point in time where the aircraft is closest (in the 2-norm sense) to the terrain[15] during the optimal recovery maneuver. The control space can be thought of as a hyper-cube, where each face represents the u_{max} or u_{min} for the corresponding control $u(t)$. Hence, an aggressive control would be one that lies on one or more of the faces of the hyper-cube. For an aircraft scenario with two controls, the hyper-cube is reduced to a square, where each edge represents an aggressive control.

Note that i may change during the interval $t \in [t_0, t_{CPA})$. For example, the optimal control solution may result in two controls at their max limits everywhere in the interval or may result in one control at its max limit for the first part of the time interval and another control at its max limit for the second part of the interval.

Next, the timely requirement must be defined. Over a short period of time $[t_0, t_f]$ it is reasonable to assume that the aircraft is flying at a relatively constant velocity. Therefore, time maps to distance for a fixed velocity. Define a distance threshold (terrain buffer or keep-out zone) $d_{threshold}$ which is the min allowable distance to terrain. Based on this, for a given control $u(t)$, a timely aircraft trajectory is a 'last-second' recovery maneuver $\mathbf{X}_{ac}(t, \mathbf{u}(t))$ that reaches but does not violate the distance

[15]In the 2-D case, terrain is defined as the closest digital terrain post, while in the 3-D case terrain means a continuous surface approximation of digital terrain which will be explained in detail in Chapter IV.

threshold. Define a timely trajectory $\mathbf{X}_{ac}$ as

$$\min_{t\in[t_0,t_f]} ||\mathbf{X}_{ac}(t,\mathbf{u}(t)) - \mathbf{X}_{Terrain}||_2 = d_{threshold} \tag{27}$$

where the trajectory $\mathbf{X}_{ac}$ is defined as the 3-D vector position of the aircraft, $\mathbf{X}_{Terrain}$ is defined as the 3-D location of the terrain which is reachable in the timespan $t \in [t_0, t_f]$, and $d_{threshold}$ is a scalar user-defined terrain buffer. For the work developed herein, $d_{threshold}$ was set to 350 feet (as explained in Chapter V Section 5.5) representing a buffer around terrain that cannot be encroached. The minimum of the 2-norm $||\mathbf{X}_{ac}(t) - \mathbf{X}_{Terrain}||_2$ is the closest point of approach (CPA) between the aircraft and the terrain in the *local map*[16]. It is assumed that there exists terrain that must be avoided within the time horizon of the optimizer. Hence, it is necessary to perform an avoidance maneuver. Thus, a timely maneuver is one that is executed such that there exists a point in time, denoted t_{CPA}, along the trajectory $\mathbf{X}_{ac}$ where equality is reached in Equation (27). It follows that $t_0 \leq t_{CPA} \leq t_f$, and if the maneuver were executed before t_0, the min of Equation (27) would exceed $d_{threshold}$ and hence would not be considered timely.

The goal is to trigger the auto recovery when it meets both the aggressive and timely requirements. Note that not all aggressive controls result in timely trajectories. Figure 15 depicts what an aggressive control looks like for a trajectory that is not timely for an aircraft with one control in a single obstacle scenario.

[16]The *local map* is the immediate vicinity around the aircraft as defined by the user and based on the aircraft mission.

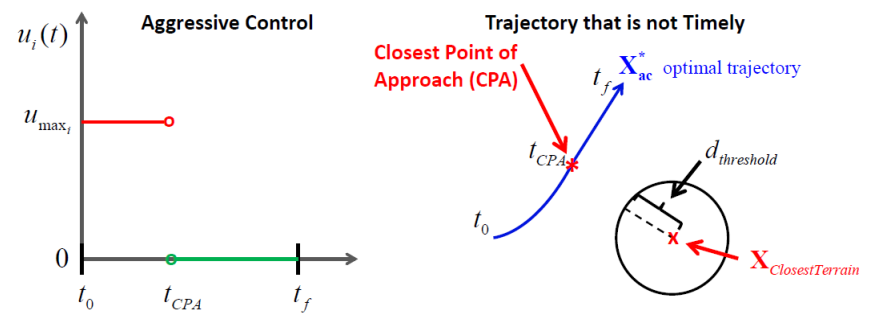


Figure 15. An Aggressive Control Resulting in a Trajectory that is Not Timely

Conversely, a control that is not aggressive can still result in a timely trajectory. Figure 16 depicts what a control that is not aggressive looks like for a timely trajectory for an aircraft with one control and a scenario with a single obstacle.

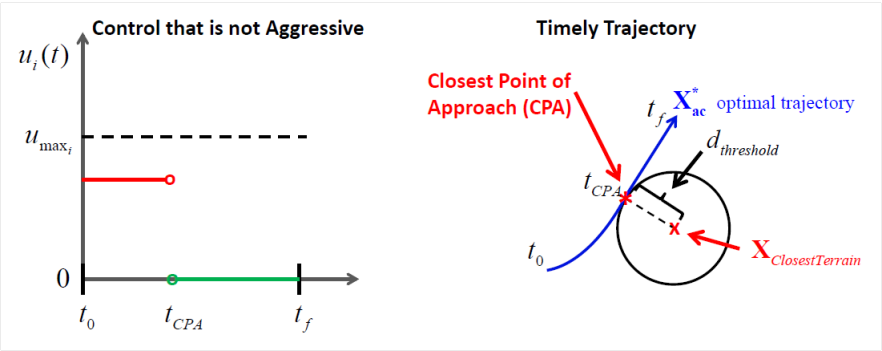


Figure 16. A Control that is Not Aggressive Resulting in a Timely Trajectory

The triggers for the auto recovery to be commanded are that both the aggressive and timely requirements are met simultaneously. The next section more fully illustrates a commanded auto recovery that is both aggressive and timely.

3.3 Mapping *Aggressive* and *Timely* Requirements to the Optimal Control Formulation

Now that the aggressive and timely requirements of the Auto GCAS recovery have been defined mathematically, these requirements need to be incorporated in an optimal control formulation. The question is how to map the aggressive and timely

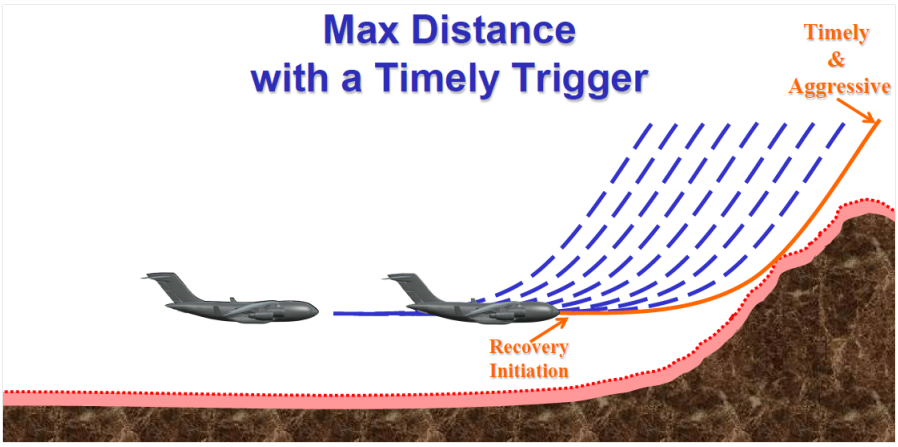
requirements to the optimal control formulation? For most problems, the engineer must accept the dynamics and boundary constraints that are inherent to the problem. However, the art of formulating an optimal control problem is in how the cost functional J and the path constraints $\mathbf{C}$ are designed. Further, for an iterative problem such as Auto GCAS, choosing the point at which the automatic recovery is triggered is crucial. For this optimal Auto GCAS research, aggressive and timely requirements have been purposefully mapped in two different ways to the cost functional J, the path constraints $\mathbf{C}$, and the trigger point for the automatic recovery.

Each of these requirements can be put either in the cost functional, in the path constraints, or (in the case of an autonomous algorithm) in the trigger logic for the automatic recovery via the autopilot. Table 6 shows how these two Auto GCAS requirements were mapped to develop two distinct optimal control formulations, one referred to as Max Distance and one as Min Control.

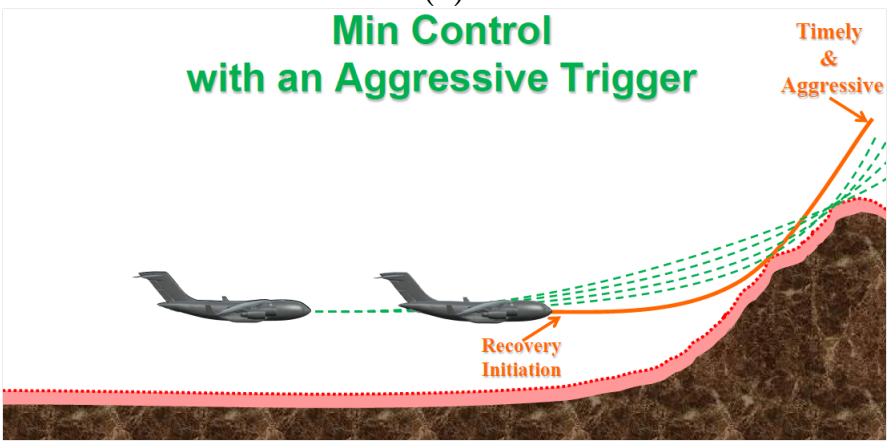
Table 6. Mapping of *Aggressive* and *Timely* requirements to J, C, and Trigger

Method Name	**Combination of Cost Functional J and Path Constraints C**	**Trigger for Auto Recovery**
Max Distance	*Aggressive*	*Timely*
Min Control	*Timely*	*Aggressive*

Each method is named based on the form of the cost functional J. It is important to remember the iterative nature of the Auto GCAS problem where a new optimal control solution is calculated at each time step but the optimal control is not commanded until the trigger criteria have been met. The two graphics in Figure 17 illustrate the iterative nature of Auto GCAS and depict what the Max Distance and Min Control optimal control formulations represent physically. As in Figure 13, the blue dashed lines represent an aggressive recovery, the green dashed lines represent a timely recovery, while the solid orange lines represent the commanded automatic recovery that is both aggressive and timely.



(a)



(b)

Figure 17. Two Optimal Control Approaches to Auto GCAS: a) Max Distance and b) Min Control

Now that we have defined the aggressive and timely requirements, the next section shows how to incorporate these requirements into an optimal control formulation.

3.4 Optimal Control Formulations for Auto GCAS

This section first develops the specifics of the two optimal control formulations from Table 6 for the general case, followed by a specific 2-D example.

3.4.1 Max Distance with a Timely Trigger.

In the Max Distance with a Timely Trigger formulation (also referred to as Max Distance for brevity), the Auto GCAS problem is posed as a maximization problem. Note that the Max Distance formulation always admits an aggressive maneuver. The goal is to find the aircraft trajectory that produces the maximum distance away (2-norm) from the terrain (represented as DTED posts) at the CPA with aggressive constraints on controls. The combination of the Max Distance performance measure J and the constraint on control

$$u_{min_i} \leq u_i(t) \leq u_{max_i} \tag{28}$$

admits an optimal solution that meets the aggressive requirement.

Define $\mathbf{X}_{ac}(t)$ as the 3-D aircraft position and $\mathbf{X}_{Terrain}$ as the 3-D location of the DTED posts in the local map. Originally, the intent was to formulate the Max Distance cost functional as a max min problem with the goal of maximizing distance away from the closest terrain to the aircraft such that

$$\underset{\mathbf{u}\in\mathbf{A}}{\text{Maximize: }} J = \min ||\mathbf{X}_{ac}(t) - \mathbf{X}_{Terrain}||_2^2 \tag{29}$$

where $\mathbf{A}$ is the set of admissible controls. However, max and min functions are not well-conditioned for a nonlinear programming (NLP) solver because they do not have smooth gradients. As can be seen in the right hand side of Equation (29), the minimum can occur at a different aircraft trajectory point and/or at a different DTED post at each iteration, which results in the non-smooth gradient.

To avoid this, a surrogate objective functional formulation was created which has continuous gradients which are required for use in the NLP solver. Mathematically, maximizing the integral of the distance is the same as maximizing the distance to

the closest terrain point as long as the function $\mathbf{X}_{ac}(t)$ is smooth. Hence, the Max Distance optimal control problem is formulated to find the optimal control $\mathbf{u}^*(t) \in \mathbf{A}$ that maximizes the cost functional

$$\underset{\mathbf{u}\in\mathbf{A}}{\text{Maximize: }} J = \int_{t_0}^{t_f} ||\mathbf{X}_{ac}(t) - \mathbf{X}_{Terrain}||_2^2 \, \mathrm{d}t \tag{30}$$

subject to the dynamic constraints which are the aircraft equations of motion,

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, t) \tag{31}$$

with control inputs $\mathbf{u}$(t) and with boundary conditions $\mathbf{\Psi}$,

$$X_0 = \text{specified} \quad \text{(initial conditions)} \tag{32a}$$

$$X_f = \text{free} \quad \text{(final conditions)} \tag{32b}$$

$$t_0 = \text{specified} \quad \text{(current time)} \tag{32c}$$

$$t_f = \text{specified} \quad \text{(fixed end time of predicted trajectory)} \tag{32d}$$

and the path constraint $\mathbf{C}$ on control (making it aggressive) is defined in Equation (28), where t_{CPA} lies in the interval $[t_0, t_f]$.

To ensure the timely requirement is met, the automatic recovery trigger occurs when the CPA of the optimal trajectory $\mathbf{X^*}_{ac}$ to the closest terrain in the local map is equal to the keep-out zone, $d_{threshold}$,

$$\min_{t\in[t_0,t_f]} ||\mathbf{X^*}_{ac}(t) - \mathbf{X}_{Terrain}||_2 = d_{threshold}. \tag{33}$$

Algorithm 1 shows the decision process for the Max Distance formulation. Note that this algorithm is an outer-loop method that sends a command to the inner loop

autopilot to execute the optimal $\mathbf{u}^*$ Auto GCAS recovery.

Algorithm 1 : Max Distance with a Timely Trigger Algorithm

Load digital terrain
Obtain current aircraft conditions
Provide initial guess of control, state, and time vectors $\mathbf{u}, \mathbf{x}$, and $\mathbf{t}$ (Cold start)
for $t_0 : \Delta t : t_f$ *for end of mission* **do**
  Obtain current aircraft conditions
  Optimizer:
    Find $\mathbf{u}^*$ that maximizes Max Distance cost functional J (Equation (30))
    Satisfy dynamic constraints $\dot{\mathbf{x}}$ (Equation (31))
    Satisfy boundary constraints $\mathbf{\Psi}$ (Equation (32))
    Satisfy algebraic path constraints $\mathbf{C}$ (Equation (28))
  Output: discretized $\mathbf{u}^*, \mathbf{x}^*$, and CPA
  if *CPA* $= d_{threshold}$ *(Equation (33))* **then**
    Command Auto GCAS Recovery (Autopilot) using previous $\mathbf{u}^*$
    Once clearance of terrain is assured, return control to pilot
  end
  Set sol'n of $\mathbf{u}^*$ and $\mathbf{x}^*$ as new initial guess of control & state vectors (Warm start)
end
Result: Optimal Auto GCAS Recovery triggered when CPA $= d_{threshold}$

3.4.2 Min Control with an Aggressive Trigger.

In the Min Control with an Aggressive Trigger formulation (also referred to as Min Control), the Auto GCAS problem is formulated to find the optimal control $\mathbf{u}^*(t)$ in the set of admissible controls $\mathbf{A}$ that minimizes the cost functional

$$\underset{\mathbf{u}\in\mathbf{A}}{\text{Minimize: }} J = \int_{t_0}^{t_f} \mathbf{u}^T(t)\mathbf{R}\mathbf{u}(t)\ \mathrm{d}t \tag{34}$$

subject to the same dynamic constraints $\dot{\mathbf{x}}$ as for Max Distance in Equation (31), with control inputs $\mathbf{u}(t)$ and where the boundary conditions $\mathbf{\Psi}$ are the same as shown in Equation (32), and the path constraint $\mathbf{C}$ which keeps the aircraft path clear of a

keep-out zone around the DTED posts

$$||\mathbf{X}_{ac}(t) - \mathbf{X}_{Terrain}||_2 \geq d_{threshold}. \tag{35}$$

To ensure the aggressive requirement is met, the automatic recovery is only triggered when

$$\left.\begin{array}{rl} u_i(t) & = u_{\max_i} \\ \text{or} & \\ u_i(t) & = u_{\min_i} \end{array}\right\} \text{for at least one control } u_i,\ \forall\ t \in [t_0, t_{CPA}) \tag{36}$$

is satisfied. The combination of the cost functional J and the path constraint $\mathbf{C}$ makes this formulation timely. In order to minimize the cost functional J and satisfy the constraint $\mathbf{C}$ on aircraft path, the optimal control $\mathbf{u}^*$ is the min control effort required to keep the aircraft on the boundary of the keep-out zone defined by $d_{threshold}$, thus ensuring timeliness. The Auto GCAS recovery is triggered when the min control solution equals the allowable control limit (either a max or a min), making the recovery aggressive. Thus, upon trigger, the Min Control formulation, like Max Distance, meets both the mission requirements of being aggressive and timely.

Algorithm 2 shows the Min Control formulation decision process.

Algorithm 2 : Min Control with an Aggressive Trigger Algorithm

Load digital terrain
Obtain current aircraft conditions
Provide initial guess of control, state, and time vectors $\mathbf{u}, \mathbf{x}$, and $\mathbf{t}$ (Cold start)
for $t_0 : \Delta t : t_f$ *for end of mission* **do**
  Obtain current aircraft conditions
  Optimizer:
    Find $\mathbf{u}^*$ that minimizes Min Control cost functional J (Equation (34))
    Satisfy dynamic constraints $\dot{\mathbf{x}}$ (Equation (31))
    Satisfy boundary constraints $\mathbf{\Psi}$ (Equation (32))
    Satisfy algebraic path constraints $\mathbf{C}$ (Equation (35))
  Output: discretized $\mathbf{u}^*, \mathbf{x}^*$, and CPA
  if $u_i^*(t) = u_{min_i}$ *or* $u_i^*(t) = u_{max_i}$ *(Equation (36))* **then**
    Command Auto GCAS Recovery (Autopilot) using previous $\mathbf{u}^*$
    Once clearance of terrain is assured, return control to pilot
  end
  Set sol'n of $\mathbf{u}^*$ and $\mathbf{x}^*$ as new initial guess of control & state vectors (Warm start)
end
Result: Optimal Auto GCAS Recovery triggered when Equation (36) is satisfied

Based on the discussion, we now clarify Table 6 as shown in Table 7. Table 7 is a more detailed summary of the mapping of the aggressive and timely mission requirements to the cost functional J, path constraints $\mathbf{C}$, and the trigger point of the Auto GCAS recovery for the general case.

Table 7. Mapping of *Aggressive* and *Timely* to J, C, and Trigger (Detailed)

Method Name	Combination of Cost Functional J and Path Constraints C	Trigger Automatic Recovery When
Max Distance	Max Distance and $U_{min_i} \leq U_i(t) \leq U_{max_i}$ $\Rightarrow$ *Aggressive*	CPA $= d_{threshold}$ $\Rightarrow$ *Timely*
Min Control	Min Control and CPA $\geq d_{threshold}$ $\Rightarrow$ *Timely*	$U_i(t) = U_{max_i}$ or $U_i(t) = U_{min_i}$[16] $\Rightarrow$ *Aggressive*

[16]for at least one control $U_i \; \forall \; t \in [t_0, t_{CPA})$, where CPA is the closest point of approach

Hence, the Max Distance formulation always admits aggressive solutions but triggering the Auto GCAS recovery when the closest point of approach of $\mathbf{X}_{ac}^*$ to the clos-

est DTED post is equal to $d_{threshold}$ makes the solution timely. Conversely, the Min Control formulation always admits timely solutions but triggering the Auto GCAS recovery when the optimal control vector satisfies Equation (36) makes the solution aggressive.

3.5 2-D Example

Initially a 2-D optimal control model was created to compare the two formulations in a simple scenario. The 2-D model considers a 3-state aircraft (x, y, and θ) with a single control $u = \dot{\theta}$ in a single obstacle scenario. Constant velocity and constant altitude are assumed for the 2-D scenario.

3.5.1 Max Distance with a Timely Trigger.

In the Max Distance with a Timely Trigger formulation, the combination of the Max Distance performance measure J and the constraint on control, $u_{min} \leq u(t) \leq u_{max}$, admits an optimal solution that meets the aggressive mission requirement. Define

$$\mathbf{X}_{ac}(t) = \begin{bmatrix} x(t) \\ y(t) \end{bmatrix} \quad \text{and} \quad \mathbf{X}_{Terrain} = \begin{bmatrix} x_{DTED} \\ y_{DTED} \end{bmatrix} \tag{37}$$

where $\mathbf{X}_{ac}(t)$ is the aircraft position and $\mathbf{X}_{Terrain}$ is the location of the closest DTED post amongst the DTED posts in the local map. As an illustrative example, only one obstacle (represented by one DTED post) is considered, therefore by default it is the closest DTED post (since it is the only post). This Auto GCAS problem formulation is posed as a Fixed Final Time, Free Final State optimal control formulation for each iteration.

The Auto GCAS optimal control problem is now formulated to find the optimal control $\mathbf{u}^*(t)$ in the set of admissible controls $\mathbf{A}$ that maximizes the cost functional,

$$\underset{\mathbf{u}\in\mathbf{A}}{\text{Maximize: }} J = \int_{t_0}^{t_f} ||\mathbf{X}_{ac}(t) - \mathbf{X}_{Terrain}||_2^2 \, \mathrm{d}t \tag{38}$$

subject to the dynamic constraints (equations of motion),

$$\dot{x}(t) = V\cos\theta(t) \tag{39a}$$

$$\dot{y}(t) = V\sin\theta(t) \tag{39b}$$

$$\dot{\theta}(t) = u(t) \tag{39c}$$

where the control input $\mathbf{u}$ is turn rate $\dot{\theta}$, with the boundary conditions $\mathbf{\Psi}$,

$$(x(t_0), y(t_0), \theta(t_0)) = (x_0, y_0, \theta_0) \tag{40a}$$

$$t_0 = \text{specified (current time)} \tag{40b}$$

$$t_f = \text{specified (fixed end time of predicted trajectory)} \tag{40c}$$

and the path constraint $\mathbf{C}$ on control,

$$u_{min} \leq u(t) \leq u_{max}. \tag{41}$$

Notice that this definition does admit $|u(t)| = u_{max}$ for $t \in [t_0, t_{CPA})$ when $u_{min} = -u_{max}$.

To ensure the timely requirement is met, trigger the automatic recovery when

$$\min_{t\in[t_0,t_f]} ||\mathbf{X}^*{}_{ac}(t) - \mathbf{X}_{Terrain}||_2 = d_{threshold} \tag{42}$$

is satisfied, which corresponds to $t = t_{CPA}$. Further, for all previous trajectories before

the trigger, the min distance is always greater than $d_{threshold}$. The combination of the cost functional J and the path constraint $\mathbf{C}$ makes this formulation aggressive. In order to maximize the cost functional J and satisfy the constraint $\mathbf{C}$ on control, the control $\mathbf{u}$ is driven to a limit until the flight path results in a point of closest approach, ${\mathbf{X}^*}_{ac}(t_{CPA})$, whose distance from the terrain is equal to the $d_{threshold}$ value. By triggering the Auto GCAS recovery at the point when the min point of closest approach of the optimal aircraft trajectory passes at the buffer around the DTED post, $d_{threshold}$, the timely requirement is met. Thus, at the trigger point the Auto GCAS recovery meets mission requirements by being both aggressive and timely.

3.5.2 Min Control with an Aggressive Trigger.

The 2-D Auto GCAS problem can also be posed as a minimization problem for the Min Control with an Aggressive Trigger formulation. Like the Max Distance formulation, Min Control is formulated as a Fixed Final Time, Free Final State optimal control problem for each iteration. Here the Auto GCAS problem is formulated to find the optimal control $\mathbf{u}^*(t)$ in the set of admissible controls $\mathbf{A}$ that minimizes the cost functional,

$$\underset{\mathbf{u}\in\mathbf{A}}{\text{Minimize: }} J = \int_{t_0}^{t_f} \mathbf{u}^T(t)\mathbf{u}(t)\ \mathrm{d}t \tag{43}$$

subject to the same dynamic constraints for Max Distance as shown in Equation (39), where the control input $\mathbf{u}$ is turn rate $\dot{\theta}$, and the boundary conditions $\mathbf{\Psi}$ are the same as Max Distance as shown in Equation (40), and the path constraint $\mathbf{C}$ which keeps the aircraft path outside of a keep-out zone/buffer around the DTED post is

$$||\mathbf{X}_{ac}(t) - \mathbf{X}_{Terrain}||_2 \geq d_{threshold}. \tag{44}$$

To ensure the aggressive requirement is met, trigger the automatic recovery when

$$\left.\begin{array}{rl} u(t) & = u_{\max} \\ \text{or} & \\ u(t) & = u_{\min} \end{array}\right\} \forall\ t \in [t_0, t_{CPA}) \tag{45}$$

is satisfied. The cost functional J and the path constraint $\mathbf{C}$ together make this formulation timely. When the min control solution admitted by the optimizer equals the allowable control limit (max or min), then Equation (45) is satisfied and the auto recovery (now aggressive and timely) is triggered.

3.6 2-D Analytical Equivalency

Both the Max Distance and Min Control optimal control formulations admit the same minima at trigger. To show this, the necessary conditions are derived here for both cases. The necessary conditions are then evaluated at the trigger point of the aggressive and timely Auto GCAS recovery to demonstrate equivalency.

3.6.1 Necessary Conditions for Optimality for Max Distance.

Using the Max Distance optimal control formulation from Section 3.5.1, the constraints are adjoined to the cost functional J to form the adjoined cost functional J_1 of the unconstrained optimization problem. For simplicity, a 2-D scenario with only one obstacle is considered, represented by the coordinates of the DTED post, (x_{DTED}, y_{DTED}). The cost functional of Equation (38) is changed to a minimization problem because the optimizer is set up for minimization problems. The scalars V, x_{DTED}, y_{DTED}, and u_{max} are constants defined by the user, and $s(t)$ is the slack variable for the inequality constraint $u^2(t) \leq u_{max}^2$. Note that a symmetric control is assumed for this 2-D example, where $u_{min} = -u_{max}$. The aircraft velocity, V, is assumed constant due to the relatively short duration of the automatic recovery. The three states x, y, and θ and the control u are time dependent as are the Lagrange multipliers λ_i and μ and slack variable s. For simplicity the explicit use of the variable t will be dropped for the remainder of this section.

$$\underset{\mathbf{u}\in\mathbf{A}}{\text{Minimize: }} J_1 = \int_{t_0}^{t_f} -(x - x_{DTED})^2 - (y_- y_{DTED})^2 + \lambda_1(Vcos\theta - \dot{x}) + \lambda_2(Vsin\theta - \dot{y}) + \lambda_3(u - \dot{\theta}) + \mu(u^2 - u_{max}^2 + s^2)\ \mathrm{d}t. \tag{46}$$

Deriving the necessary conditions for optimality results in nine equations for nine unknowns: $x,\ y,\ \theta,\ u,\ \lambda_1,\ \lambda_2,\ \lambda_3, \mu$, and s,

Co-state equations:

$$\frac{\partial J_1}{\partial x} = 0 \Rightarrow \dot{\lambda_1} = 2(x - x_{DTED}) \tag{47a}$$

$$\frac{\partial J_1}{\partial y} = 0 \Rightarrow \dot{\lambda_2} = 2(y - y_{DTED}) \tag{47b}$$

$$\frac{\partial J_1}{\partial \theta} = 0 \Rightarrow \dot{\lambda_3} = \lambda_1(V \sin\theta) - \lambda_2(V \cos\theta) \tag{47c}$$

Optimal control equation:

$$\frac{\partial J_1}{\partial u} = 0 \Rightarrow 2\mu u = -\lambda_3 \tag{47d}$$

State equations:

$$\frac{\partial J_1}{\partial \lambda_1} = 0 \Rightarrow \dot{x} = V \cos\theta \tag{47e}$$

$$\frac{\partial J_1}{\partial \lambda_2} = 0 \Rightarrow \dot{y} = V \sin\theta \tag{47f}$$

$$\frac{\partial J_1}{\partial \lambda_3} = 0 \Rightarrow \dot{\theta} = u \tag{47g}$$

Slackness equation:

$$\frac{\partial J_1}{\partial \mu} = 0 \Rightarrow u^2 - u_{max}^2 + s^2 = 0 \tag{47h}$$

Switching condition:

$$\frac{\partial J_1}{\partial s} = 0 \Rightarrow 2\mu s = 0. \tag{47i}$$

For the automatic recovery to be triggered, the recovery generated must meet the requirements of aggressive and timely. Every optimal recovery resulting from the Max Distance formulation is aggressive by design, therefore $|u(t)| = u_{max}$ (based on the assumption of a symmetric control). Upon trigger, the automatic recovery is timely because it is only triggered when the closest point of approach of the recovery equals $d_{threshold}$. Since $|u(t)| = u_{max}$ is assumed, then from the slackness equation $s = 0$,

meaning the control constraint is active across the interval $t \in [t_0, t_{CPA})$. We are only concerned with the time interval when the Auto GCAS recovery is activated, which is the initial interval $t \in [t_0, t_{CPA})$. Just after t_{CPA}, the control is returned back to the pilot, hence the autopilot is not commanding the control during the interval $t \in [t_{CPA}, t_f]$. With $s = 0$, the switching condition is also satisfied and the Lagrange multiplier μ can be arbitrary. Integrating the equations of motion yields the following optimal solution $\forall\ t \in [t_0, t_{CPA})$

$$|u(t)| = u_{max} \tag{48a}$$

$$x(t) = x_0 + \frac{V}{u_{max}}(\sin\theta(t) - \sin\theta_0) \tag{48b}$$

$$y(t) = y_0 - \frac{V}{u_{max}}(\cos\theta(t) - \cos\theta_0) \tag{48c}$$

$$\theta(t) = \theta_0 + (t - t_0)u_{max}. \tag{48d}$$

3.6.2 Necessary Conditions for Optimality for Min Control.

The Min Control optimal control formulation from Section 3.5.2 is used. The constraints are adjoined to the Min Control cost functional J of Equation (43) to form an unconstrained optimization problem with the adjoined cost functional J_2. The scalar $d_{threshold}$ represents the keep-out zone around the DTED post and is a constant defined by the user, and $\tilde{s}$ is the slack variable for the inequality constraint $d_{threshold} \leq \sqrt{(\mathbf{x}(t) - x_{DTED})^2 + (\mathbf{y}(t) - y_{DTED})^2}$. A tilde is used here over the Lagrange multipliers and slack variable of the Min Control formulation to show that they are not identical to those of the Max Distance formulation. The adjoined cost

functional J_2 is

$$\underset{\mathbf{u}\in\mathbf{A}}{\text{Minimize: }} J_2 = \int_{t_0}^{t_f} u^2 + \tilde{\lambda_1}(Vcos\theta - \dot{x}) + \tilde{\lambda_2}(Vsin\theta - \dot{y}) + \tilde{\lambda_3}(u - \dot{\theta}) \tag{49}$$
$$+ \tilde{\mu}(d^2_{threshold} - (x - x_{DTED})^2 - (y\_y_{DTED})^2 + \tilde{s}^2)\ \mathrm{d}t.$$

Deriving the necessary conditions for optimality results in nine equations for nine unknowns: $x,\ y,\ \theta,\ u,\ \tilde{\lambda_1},\ \tilde{\lambda_2},\ \tilde{\lambda_3}, \tilde{\mu}$, and $\tilde{s}$,

Co-state equations:

$$\frac{\partial J_2}{\partial x} = 0 \Rightarrow \dot{\tilde{\lambda}}_1 = 2\tilde{\mu}(x - x_{DTED}) \tag{50a}$$
$$\frac{\partial J_2}{\partial y} = 0 \Rightarrow \dot{\tilde{\lambda}}_2 = 2\tilde{\mu}(y - y_{DTED}) \tag{50b}$$
$$\frac{\partial J_2}{\partial \theta} = 0 \Rightarrow \dot{\tilde{\lambda}}_3 = \tilde{\lambda_1}(V\sin\theta) - \tilde{\lambda_2}(V\cos\theta) \tag{50c}$$

Optimal control equation:

$$\frac{\partial J_2}{\partial u} = 0 \Rightarrow 2u = -\tilde{\lambda_3} \tag{50d}$$

State equations:

$$\frac{\partial J_2}{\partial \tilde{\lambda_1}} = 0 \Rightarrow \dot{x} = V\cos\theta \tag{50e}$$
$$\frac{\partial J_2}{\partial \tilde{\lambda_2}} = 0 \Rightarrow \dot{y} = V\sin\theta \tag{50f}$$
$$\frac{\partial J_2}{\partial \tilde{\lambda_3}} = 0 \Rightarrow \dot{\theta} = u \tag{50g}$$

Slackness equation:

$$\frac{\partial J_2}{\partial \tilde{\mu}} = 0 \Rightarrow d^2_{threshold} - (x - x_{DTED})^2 - (y\_y_{DTED})^2 + \tilde{s}^2 = 0 \tag{50h}$$

Switching condition:

$$\frac{\partial J_2}{\partial \tilde{s}} = 0 \Rightarrow 2\tilde{\mu}\tilde{s} = 0. \tag{50i}$$

Realizing that the Lagrange multipliers may be different, note that the necessary conditions for Min Control above admit the same equations as for Max Distance with the exception of Equations (50d) and (50h).

The automatic recovery generated for Min Control must also meet the requirements of aggressive and timely. Every optimal recovery resulting from the Min Control formulation is timely by design. Upon trigger, the automatic recovery becomes aggressive because it is only triggered when $|u(t)| = u_{max}$ (where symmetric control is assumed) for the time interval from trigger time up through but not including the time at the closest point of approach, i.e. $\forall\ t \in [t_0, t_{CPA})$. Examination of the slackness equation shows that $\tilde{s} \neq 0$ due to the fact that the 2-norm distance threshold constraint is not active except at the closest point of approach, where $t = t_{CPA}$. In other words, the 2-norm of every other element of the optimal path is greater than the distance threshold, except at $t = t_{CPA}$ where the 2-norm equals $d_{threshold}$. Since $\tilde{s} \neq 0$, then $\tilde{\mu} = 0$ in order to satisfy the switching condition. Integrating the equations of motion yields the identical results as for Max Distance, and, therefore, results in the equivalent optimal solution $\forall\ t \in [t_0, t_{CPA})$

$$|u(t)| = u_{max} \tag{51a}$$

$$x(t) = x_0 + \frac{V}{u_{max}}(\sin\theta(t) - \sin\theta_0) \tag{51b}$$

$$y(t) = y_0 - \frac{V}{u_{max}}(\cos\theta(t) - \cos\theta_0) \tag{51c}$$

$$\theta(t) = \theta_0 + (t - t_0)u_{max}. \tag{51d}$$

The optimal Max Distance solution shown previously in Equation (48) is identical to the optimal Min Control solution shown in Equation (51) $\forall\ t \in [t_0, t_{CPA})$. The analytical solutions show the equivalence of the Max Distance and Min Control optimal control and optimal path solutions $\forall\ t \in [t_0, t_{CPA})$ at the instant at which

the auto recovery maneuver is triggered (i.e. commanded by the autopilot). In order to determine which method should be used for real-time onboard calculations, the computational results will be compared to determine which method has the fastest computational speed.

3.7 2-D Numerical Results

A generic heavy aircraft was used for the simulation. For the purposes of this research, a military transport flying a low-level route at $V = 210$ kts (or 108 m/sec) is assumed. An aggressive recovery at control limits u_{max} or u_{min} is defined here to be $\pm$ 9°/sec of turn rate. This is based on the assumption that aggressive maneuvers for this aircraft are $\pm$ 60° of bank flying at 108 m/sec. These values are representative of the performance of a slow to medium speed military transport. An aircraft standard turn rate is 3°/sec, so the Auto GCAS aggressive recovery maneuver is three times the standard turn rate. For the Min Control formulation, the automatic recovery is triggered when the timely Min Control solution requires 9°/sec of turn rate (aggressive). For the Max Distance formulation, the predicted recovery trajectory will always be at the aggressive 9°/sec turn rate, but the automatic recovery is only triggered when the min point of closest approach equals $d_{threshold}$ (i.e. the keep-out zone or buffer around the DTED post), thus being timely. In this example, $d_{threshold}$ is defined to be 5 miles.

Figure 18 shows the optimal path and optimal control results for the commanded auto recovery for both optimal control formulations. Figure 18 a) on the left is a bird's eye view of the optimal aircraft trajectory, and Figure 18 b) on the right is the optimal control input versus time. The aircraft is initially travelling Northeast, with the aircraft's x-position defined as East and the y-position defined as North. The aircraft initial conditions are $x_0 = 1.5$ miles, $y_0 = 1.0$ miles, and heading $\theta_0 = 5°$,

where 0° heading is defined as East.

Figure 18 shows a comparison of the results for Min Control Formulation (blue diamonds) and Max Distance (green circles) prior to the point of trigger of the automatic recovery. Note that neither the optimal aircraft trajectories nor the optimal control inputs are equivalent for Max Distance and Min Control *prior* to the triggering of the auto recovery. The red asterisks represent the CPAs for each the Max Distance and Min Control formulations. The black X marks the point of radial tangency for the Max Distance formulation. The point at which the aircraft finishes the max rate turn and transitions to a constant heading as seen in Figure 18 a) is the time point in the Max Distance solution where the aircraft transitions from a max control effort turn and rolls out on the radial ray to a zero control effort constant heading as shown on the right in Figure 18 b).

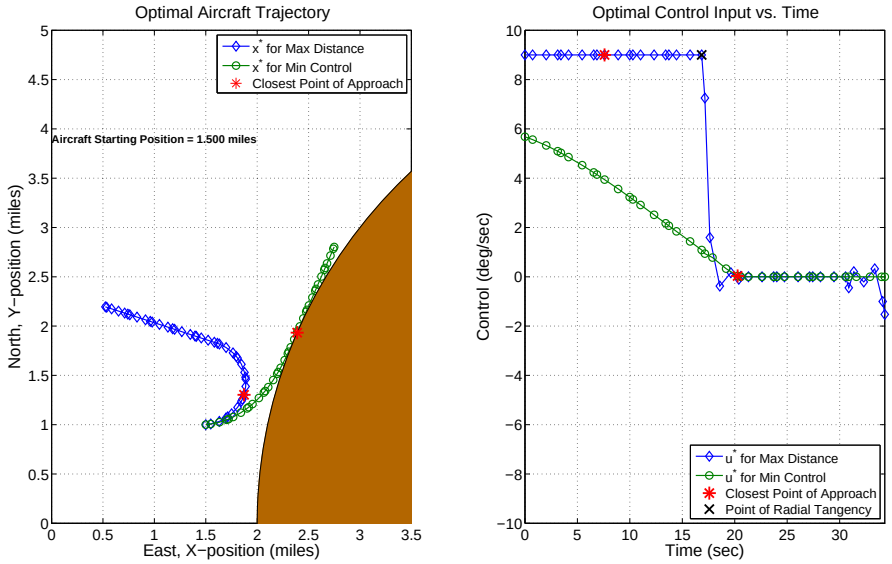


Figure 18. a) Optimal Aircraft Trajectory and b) Optimal Control Results for Max Distance and Min Control Formulations Prior to Triggering the Auto Recovery

Figure 19 shows the moment the automatic recovery was initiated for both formulations. From this plot it is clear that the Max Distance and Min Control formulations

admit the exact same optimal path and optimal control up until the point of closest approach denoted by the red asterisk (just after this point the control would be given back to the pilot). Figure 19 does not show the control being returned to the pilot, but rather shows the optimal control solution admitted by the optimizer for the entire time horizon. However, in a real-time implementation, control would be immediately returned to the pilot just after the point of closest approach once clearance of terrain was assured.

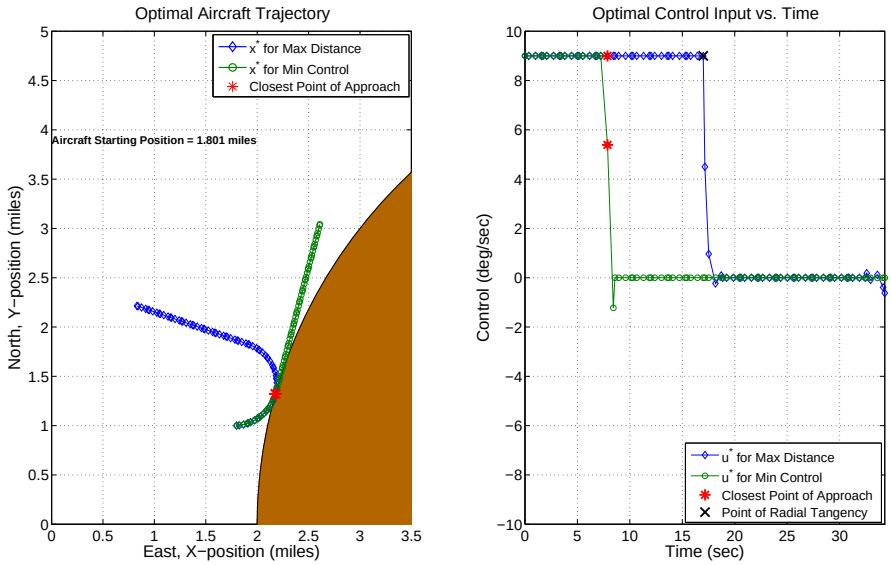


Figure 19. Optimal Aircraft Trajectory and Optimal Control Results for Max Distance and Min Control Formulations at the Trigger Point of the Auto Recovery

Figure 20 shows a zoomed in view of the optimal path and optimal control solution in order to more clearly see that the optimal paths are identical for both formulations up until the CPA. This demonstrates that both formulations admit the same path at the point in time that the automatic recovery is triggered. Note in both Figure 19 and 20 that control would be transitioned from the autopilot back to the pilot after the algorithm ensures clearance of terrain obstacles. In this one obstacle scenario, control would given back to the pilot right after the CPA, however the entire optimal path and optimal control solution are shown for clarity.

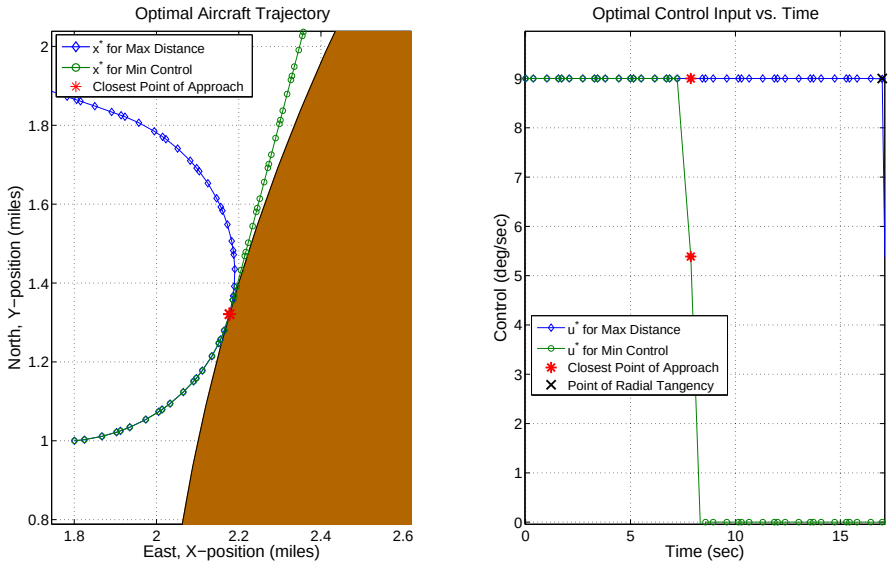


Figure 20. Optimal Aircraft Trajectory and Optimal Control Results for Max Distance and Min Control Formulations at the Trigger Point of the Auto Recovery (Zoomed In View)

While the Max Distance and Min Control optimal solutions are not equivalent prior to trigger, Figure 20 shows that they do admit equivalent optimal control and optimal path solutions at the trigger point for $t \in [t_0, t_{CPA})$.

For clarification, t_{CPA} lies in the interval $[t_0, t_f]$. The time at closest point of approach t_{CPA} is the same for both the Min Control and Max Distance solutions. Control would be given back to the pilot just after t_{CPA}, once the algorithm assures that the aircraft will clear terrain. When to give control back to the pilot is an implementation decision, not an optimal control result.

3.8 Method Comparison

In comparing the Max Distance and Min Control formulations for the specific 2-D example, both the trigger point and the computational speed of the NLP solver IPOPT were considered. Because the automatic recovery maneuvers must be calculated quickly and executed at the last second, computational speed is extremely

important. When the two optimal control formulations are compared, both have the same trigger point of 1.81 miles since they admit equivalent solutions. However, the Min Control formulation has a quicker NLP solver time and therefore Min Control is the recommendation for Auto GCAS implementation.

Table 8 shows the comparison of results for the trigger point and the NLP solver speed for both the Max Distance and Min Control formulations for 80 collocation points.

Table 8. Comparison of Max Distance & Min Control Results for 80 collocation pts

Method	x_0 **at Trigger**	**NLP Solver Time**
Max Distance	1.81 mi	0.386 sec
Min Control	1.81 mi	0.301 sec

It is also important to note that warm starting the algorithm was used in running both the Max Distance and Min Control simulations. Once an optimal solution was found it was fed back as the initial guess for the next iteration as the aircraft moved closer to the obstacle.

3.9 Two-Phase Formulation for Min Control

Up to this point, the problem has been posed as a single-phase optimal control problem. However as a single-phase, it is difficult to seek the exact point in time where triggering the auto recovery needs to take place for the Min Control formulation. Creating a two-phase problem formulation allows the trigger criteria to be imposed on the first phase for the Min Control formulation. This is done by adding two linkage constraints.

Linkage Constraint 1 This constraint enforces continuity of states and time across the two phases. Subscript 1 denotes Phase 1 and subscript 2 denotes Phase 2.

$$x_{0_2} - x_{f_1} = 0 \tag{52a}$$

$$y_{0_2} - y_{f_1} = 0 \tag{52b}$$

$$\theta_{0_2} - \theta_{f_1} = 0 \tag{52c}$$

$$t_{0_2} - t_{f_1} = 0 \tag{52d}$$

Linkage Constraint 2 Phase transition occurs when the distance derivative with respect to the obstacle is zero, i.e. at closest point of approach. Then in Equations (53) and (54), $t = t_{CPA}$.

$$d = \sqrt{(x(t) - x_{DTED})^2 + (y(t) - y_{DTED})^2} \tag{53a}$$

$$d^2 = (x(t) - x_{DTED})^2 + (y(t) - y_{DTED})^2 \tag{53b}$$

Next, take the derivative to get $\dot{d}$, the relative velocity of the aircraft to the obstacle. We know that velocity with respect to the single obstacle equals zero at the CPA, therefore it can be used as the linkage constraint.

$$2d\dot{d} = 2(x(t) - x_{DTED})\dot{x} + 2(y(t) - y_{DTED})\dot{y} \tag{54a}$$

$$\dot{d} = [(x(t) - x_{DTED})\dot{x} + (y(t) - y_{DTED})\dot{y}]/d \tag{54b}$$

$$\dot{d} = [(x(t) - x_{DTED})V\cos\theta + (y(t) - y_{DTED})V\sin\theta]/d \tag{54c}$$

Use the final values of Phase 1 to check $\dot{d}_{t_f}$, which is the distance derivative at the

end of Phase 1. Phase transition occurs when $\dot{d}_{t_f} = 0$.

$$\dot{d}_{t_f} = (x_f - x_{DTED}) \cos\theta_f + (y_f - y_{DTED}) \sin\theta_f \tag{55}$$

Figure 21 shows the two-phase solution for the Min Control formulation. Triggering of the auto recovery is commanded when control $|u(t)| = u_{max}$ for the first phase where $t \in [t_0, t_{CPA})$. Mathematically a discontinuity exists in the optimal control at this jump condition between the two phases, however computationally a finite time transition occurs between the two phases (shown by the three collocation points during the transition at the end of Phase 1).

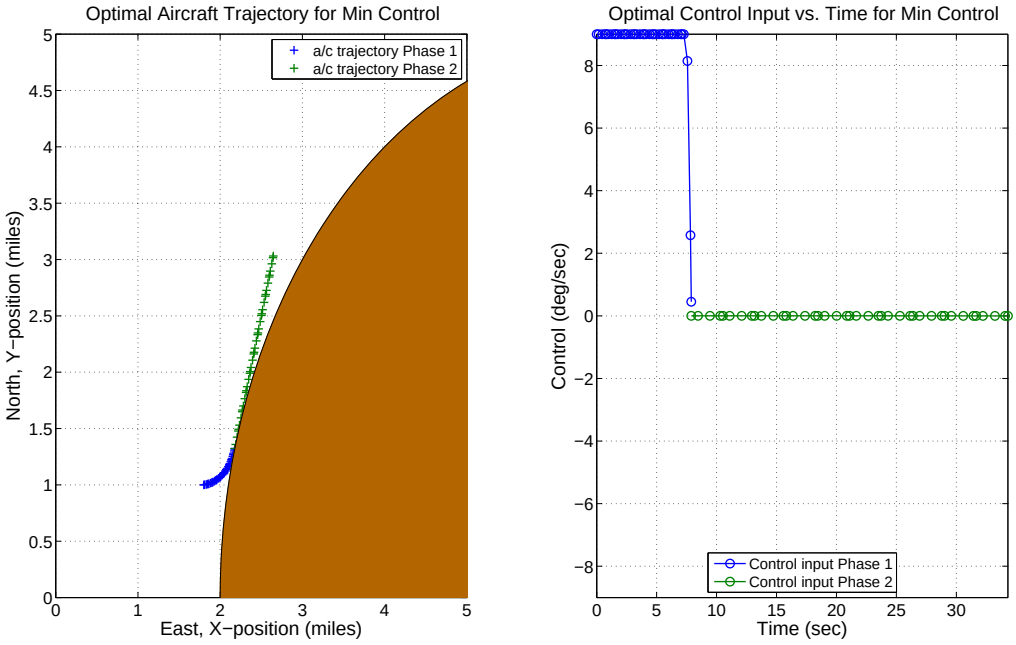


Figure 21. Two-Phase Optimal Aircraft Trajectory and Optimal Control Results for the Min Control Formulation at the Trigger Point of the Auto Recovery

3.10 Cost Functional Selection

As was shown above, the Max Distance and Min Control formulations with the cost functionals given in Equations (30) and (34) respectively are equivalent for the

2-D case with a single obstacle and a single control. For the 3-D case, tractability needs to be considered. In the 3-D case the maximization of Equation (30) requires a search of the entire local terrain map over all possible aircraft trajectories to find the closest point. In contrast, the minimization formulation in Equation (34) only requires searching over a finite number of controls to find the minimum control, which significantly reduces the search space. Based on this and the desire to have a formulation that will be suitable for real-time implementation, the Min Control formulation is chosen for all work in subsequent chapters. In short, the best way to achieve an optimal recovery trajectory for aircraft with lower climb performance is to associate the aggressive requirement with a Min Control cost functional and the timely requirement with the threshold distance constraint.

3.11 Summary

In summary, an optimal control approach was introduced in this chapter to solve the automatic ground collision avoidance problem by mapping the aggressive and timely auto recovery requirements to the optimal control formulation. Two optimal control formulations for optimal Auto GCAS were developed from this mapping. For a 2-D single obstacle example, this research demonstrated the equivalence of the optimal path and optimal control solutions for the Max Distance with a Timely Trigger technique and the Min Control with an Aggressive Trigger technique at the point of trigger of the auto recovery. The fact that the formulations were shown to have the same optimal control solutions when the auto recovery is triggered means that either of them can be chosen with good results for the 2-D case and that the user can decide which implementation makes the most sense for their particular aircraft and mission. Further, an optimal Auto GCAS algorithm must perform in real-time or near real-time to provide constant terrain protection, which means that this au-

tomatic recovery capability would be limited to aircraft with sufficient computing capability. Due to the real-time requirement, the Min Control with an Aggressive Trigger technique with its faster execution time is recommended.

Chapter IV will present the methodology for choosing two of the important pieces of the 3-D optimal control formulation for Auto GCAS. How to model the digital terrain is discussed first, followed by how to implement the terrain path constraint for Auto GCAS.

IV. 3-D Auto GCAS Optimal Control Implementation

The 2-D example problem with a single control in Chapter III demonstrated that the Max Distance and Min Control formulations for Auto GCAS are equivalent at the trigger point for the one obstacle scenario. Before solving for the 3-D model case, the appropriate digital terrain approximation and path constraint for the 3-D optimal control formulation must be chosen. For Auto GCAS, the path constraint is based on digital terrain. Therefore, the focus of this chapter is on two specific pieces of an optimal control formulation for Auto GCAS: 1) how to model the digital terrain, and 2) how to implement the path constraint.

4.1 Path Constraint Implementation/Evaluation

4.1.1 Terrain Approximation.

For the real-world implementation, the aircraft is not simply avoiding a single obstacle or post, but avoiding the ground. Therefore, the best method to approximate the ground for use as a constraint must be determined. In Chapter II, two different ways to approximate the digital terrain were presented: the F-16 Auto GCAS terrain scanning method and the National Aeronautics and Space Administration's (NASA's) compressed digital terrain map (CDTM) method. The F-16 scanning method is computationally fast, which is due to the fact that the 3-D digital terrain is converted into a 2-D terrain profile. This scanning method works very well for one pre-planned trajectory which can be compared to the 2-D profile, however, if multiple trajectories are going to be considered then a fast way to approximate the 3-D terrain is desired (especially for an optimal control method which considers an infinite number of trajectories within the given bounds). Although CDTM is a continuous approximation of 3-D terrain and computationally fast, it is only piece-wise continuous. Hence, it is

not smooth and therefore not differentiable. This research uses an optimal control approach requiring an NLP solver. NLP solvers require continuous gradient information to find local and global minima, therefore continuous and smooth path constraints must be supplied to the NLP. Second-derivative information is required in order to meet second-order optimality criteria. Hence, a continuous and smooth surface approximation of the discrete DTED posts as a path constraint is required. To meet this requirement a third method was developed to create a continuous and differentiable surface approximation. The continuous surface can then be evaluated at any query point using different interpolation schemes. MatLab has a built-in function 'griddedInterpolant' in the Mapping Toolbox that performs interpolation of multi-dimensional gridded data like DTED [59]. The griddedInterpolant function performs "a cubic interpolation of the values at neighboring grid points in each respective dimension" [59]. For three-dimensional data like DTED, the cubic interpolation is performed in two dimensions. The griddedInterpolant function is computationally fast because the interpolating polynomials and coefficients can be determined outside the NLP and then quickly queried in the NLP for path constraint evaluation. Four interpolation schemes were considered for this research: 'nearest', 'linear', 'cubic', and 'spline.'

In order to compare the four interpolation methods, synthetic DTED was created. The red stems in Figure 22 represent a 20x20 grid of equispaced posts created from 0 to 24 units in both the x-axis and y-axis with corresponding terrain heights z created based on the analytical function 'peaks' in MatLab given in Equation (56)

$$z = 3(1-x)^2 \mathrm{e}^{(-(x^2)-(y+1)^2)} - 10\left(\frac{x}{5} - x^3 - y^5\right)\mathrm{e}^{(-x^2-y^2)} - \frac{1}{3}\mathrm{e}^{(-(x+1)^2-y^2)}. \qquad (56)$$

Any negative z values were replaced with a zero height value so that all terrain heights would be greater than or equal to zero. The three x, y, and z matrices are

provided to griddedInterpolant to create the 2-D interpolant F. The interpolant F is queried with an $\mathbf{x}$ and $\mathbf{y}$-vector representing an aircraft path for interpolation in the z-direction, where the vector $\mathbf{z}_{interp} = F(\mathbf{x}, \mathbf{y})$. For this comparison, a finely spaced 0.1 $\mathbf{x}$-vector from 0 to 24 is created along with a $\mathbf{y}$-vector of the same length at a constant value of $y = 23.0$. The interpolant F is then queried with these $\mathbf{x}$ and $\mathbf{y}$-vectors to produce the four different interpolations for $\mathbf{z}_{interp}$ as shown. A comparison of the four interpolation methods is shown in Figure 23, which is zoomed in from Figure 22 to show only the last two rows of DTED in the y-direction. The first interpolation is done at $y = 23.0$ and the second at $y = 23.5$ to illustrate the differences between the interpolation schemes. The red stems represent the digital terrain posts spaced 1.2632 units apart with 'true' terrain height z. The green line shows the nearest neighbor interpolation, the black line shows the linear interpolation, the magenta line shows the cubic interpolation, and the blue line shows the spline interpolation method.

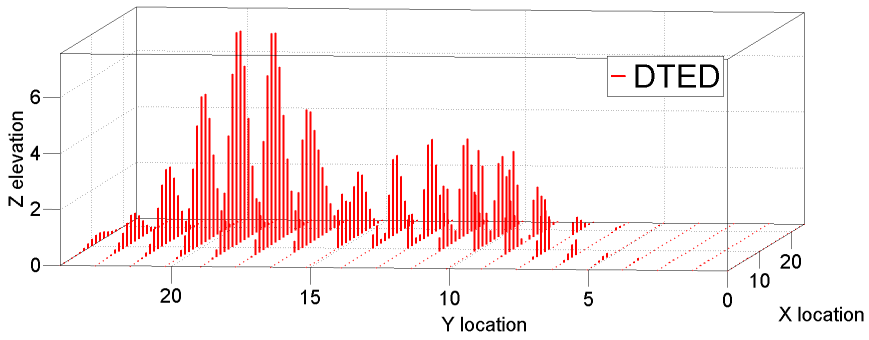


Figure 22. DTED posts for Sample 20x20 Post Grid

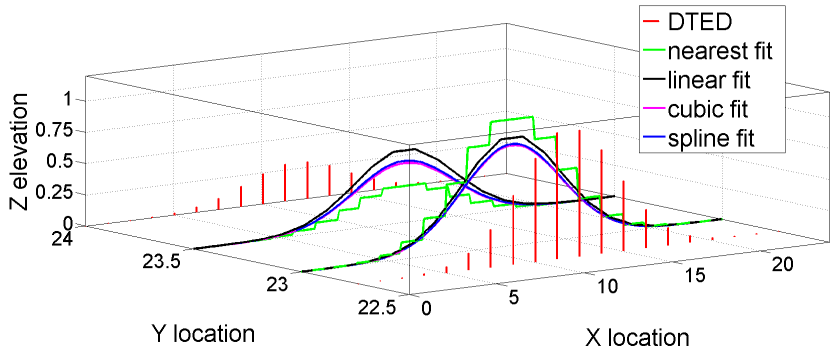


Figure 23. Comparison of Four Interpolation Methods at Two Different Y 'Slices'

4.1.1.1 Advantages and Disadvantages.

The 'nearest' method interpolates by assigning the value of the nearest sample grid point to the query point [59]. As Figure 23 depicts, sometimes the nearest neighbor method overestimates as shown by the interpolation at $y = 23.0$ and sometimes it underestimates as shown by the interpolation at $y = 23.5$. Further, the interpolated data from the 'nearest' method is discontinuous and hence not suitable for use by the NLP. With the 'linear' method, the interpolated value is a linear interpolation of the values of the neighboring grid points in each required dimension. Since the 'linear' method is only C^0 continuous, it is not adequate for use by the NLP solver.

Both the 'cubic' and 'spline' methods use a cubic interpolation of values of neighboring grid points in each dimension. However, 'cubic' uses a cubic convolution scheme and is C^1 continuous, while 'spline' is based on a cubic spline scheme and is C^2 continuous [59]. Since the NLP solver IPOPT (used for this research) requires that both first and second derivatives are continuous to find 2^{nd}-order optimality conditions, the 'spline' method was chosen to interpolate the DTED for this research. If an NLP solver is used, any twice-differentiable interpolating function will do. Aguilar et al. found that both Multiquadric Radial Basis Functions (RBFs) and Multilog RBFs were the best interpolation methods for mountainous terrain since spline methods

smooth the data and may not be representative of the actual terrain [2]. "[A spline] method is very useful for creating elevation models of areas with gently varying terrain and smooth slope transitions. However, it is very sensitive and not suitable for sharp changes in value over short distances, such as steep cliffs or man-made features. In such situations, it has a tendency to over-exaggerate the value of neighboring sample points" [2]. A suggestion for further research is to examine the suitability of other interpolation methods such as Multiquadric and Multilog RBFs for Auto GCAS when operating in mountainous terrain.

Synthetic digital terrain data was created by modifying Equation (56) with an affine transformation on x and y, where $x,\ y$, and z represent the longitude, latitude, and height of the digital terrain. This approach was used because the peaks function has an analytical solution and hence could be used as the truth source to compare against the different interpolation schemes. The new function $z = f(\tilde{x}, \tilde{y})$ is shown in Equation (57)

$$z = 3(1-\tilde{x})^2 \mathrm{e}^{(-(\tilde{x}^2)-(\tilde{y}+1)^2)} - 10\left(\frac{\tilde{x}}{5} - \tilde{x}^3 - \tilde{y}^5\right)\mathrm{e}^{(-\tilde{x}^2-\tilde{y}^2)} - \frac{1}{3}\mathrm{e}^{(-(\tilde{x}+1)^2-\tilde{y}^2)}. \qquad (57)$$

where the x and y values have been scaled and shifted using the affine transformations $\tilde{x} = \alpha(x-\beta)$ and $\tilde{y} = \alpha(y-\beta)$, where $\alpha = 3,600$ m and $\beta = 10,800$ m. The α and β values were chosen to match the same dimensions in meters as the mountainous area of interest in Southern California examined for this research. Negative z values were replaced with a zero height value to resemble the terrain area of interest that is entirely above sea level. The z values were then scaled using $\tilde{z} = z * 100$ to approximately match the height in meters of the candidate terrain. Three grid resolutions of 900 m, 90 m, and 30 m post spacing were created to simulate the resolution of actual DTED and SRTM DTED. The results are shown in Table 9.

Table 9. Error in Meters for Four Interpolation Methods

900 m Post Spacing (similar to DTED-0 spacing)				
	Nearest	**Linear**	**Cubic**	**Spline**
Underestimate (m)	174.82	35.74	19.92	22.98
Overestimate (m)	178.16	64.74	52.68	47.00
Standard Dev of Error (m)	22.08	5.96	2.64	2.33
Mean of Error (m)	-3.85e-02	-4.08e-02	-1.43e-02	-3.10e-03
90 m Post Spacing (similar to DTED-1/SRTM-1 spacing)				
	Nearest	**Linear**	**Cubic**	**Spline**
Underestimate (m)	20.37	0.38	2.75	2.96
Overestimate (m)	20.29	6.96	5.09	4.59
Standard Dev of Error (m)	2.22	0.13	0.07	0.07
Mean of Error (m)	-4.43e-04	-4.43e-04	-7.02e-05	-6.93e-05
30 m Post Spacing (similar to DTED-2/SRTM-2 spacing)				
	Nearest	**Linear**	**Cubic**	**Spline**
Underestimate (m)	5.118	0.038	0.877	0.959
Overestimate (m)	5.119	2.243	1.500	1.375
Standard Dev of Error (m)	0.702	0.022	0.014	0.013
Mean of Error (m)	-4.35e-05	-4.35e-05	-6.17e-06	-6.16e-06

The top two plots in Figure 24 show the comparison of sample digital terrain data to a surface approximation of the digital data using a spline interpolation method. The bottom two plots in Figure 24 show the error between the analytical expression in Equation (57) (the truth source) and the spline interpolation methods for 900 m and 90 m spacing. As shown in Table 9 and Figure 24 the spline interpolation method has a smaller error for 90 m than for 900 m post spacing. 90 m is the same post spacing as SRTM-1 used for this research.

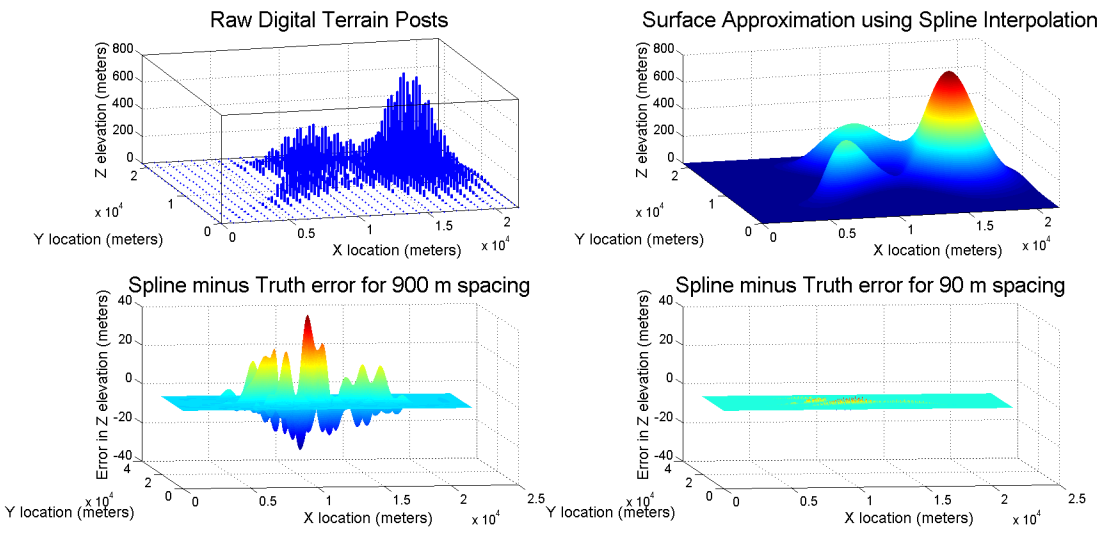


Figure 24. Raw Digital Terrain Posts with a Surface Approximation using Spline Interpolation and Interpolation Error between Truth and Spline for 900 m and 90 m Spacing

Aguilar et al. concluded that "terrain morphology, sampling density of the points observed and the interpolation method have a significant bearing on the accuracy of interpolated heights" for digital gridded terrain models [2]. The factor with the most impact on the quality of the interpolation was terrain morphology, hence an interpolation scheme should be tailored to the type of terrain you are flying over. Further, as expected the higher the sampling density of the raw digital terrain model, the higher accuracy of the interpolated heights. For this research, a compromise was found between an accuracy that is good enough for collision avoidance and computational speed of the on-board algorithm that is fast enough for real-time computations. Therefore, SRTM-1 with ∼90 meter post spacing is used with a spline interpolation method to estimate the height of the terrain under the optimal aircraft path. Based on the results of the limited error analysis given in Table 9, an overestimate of the terrain height of 4.59 m and an underestimate of 2.96 m is possible with 90 m post spacing. The standard deviation of the error is 0.07 m and the mean is -6.93e-05 m. For this research, a terrain buffer of $d_{threshold}$ = 106.68 m (350 ft) is used. Hence,

the overestimate and underestimate error values are small compared to $d_{threshold}$. Up to this point, metric units were used due to the fact that the standard format for DTED is in meters. However, for the remainder of the research herein, English units of nautical miles (NM) for the aircraft (x, y) location and feet (ft) for aircraft altitude and terrain height will be used as that is the standard convention for aviation.

4.2 Path Constraint Implementation

In order to account for DTED and interpolation errors, a path constraint must be implemented that incorporates a buffer. This buffer is necessary to incorporate not only the desired minimum miss distance from terrain but also any uncertainties due to DTED error, terrain approximation uncertainty, and navigation/location uncertainty. Two methods to implement this terrain path constraint were considered for this research, a buffer around the terrain and a buffer around the aircraft.

4.2.1 Terrain Buffer Method.

There are multiple ways to incorporate the terrain buffer. The buffer can be applied to discrete terrain posts, to a 2-D terrain profile (as done in the F-16 Auto GCAS), to a 3-D faceted surface approximation (as done in the NASA Small UAV Auto GCAS), or to a 3-D continuous surface approximation. For this research a 3-D continuous surface approximation is chosen with a terrain buffer. Two methods of implementing a terrain buffer path constraint are considered, a Look-Down approach and a left-center-right (LCR) Look-Down approach. Note that because the constraint will be imposed at each sample point (collocation point) along the aircraft trajectory, this is a 'forward'-looking approach.

4.2.1.1 Look-Down.

The simplest way to implement a path constraint to guarantee the aircraft trajectory stays above the terrain is by using a Look-Down approach. First, the continuous surface approximation of the DTED is created outside the NLP solver using the griddedInterpolant function as outlined in Appendix B. The NLP verifies that the path constraints for staying above the terrain buffer are satisfied for the optimal path by evaluating the interpolating function F. The interpolating function F is evaluated in the z-dimension using the optimal path $\mathbf{x}^*$ and $\mathbf{y}^*$. The clearance above the terrain is simply calculated by subtracting the interpolated terrain height from the actual altitude of the aircraft from the optimal path $\mathbf{z}^*$ as shown in Equations (58) and (59).

$$Clearance = z^* - Zinterp \tag{58}$$

$$Clearance \geq 350 \text{ ft} \tag{59}$$

The path constraint ensures that the clearance above terrain is always greater than or equal to the terrain buffer, $d_{threshold}$, which in this research is 350 ft. The terrain buffer is analogous to a nuisance-free altitude for Auto GCAS. A nuisance-free altitude of 350 ft was determined to be acceptable to pilots for a 500 ft AGL low-level in mountainous terrain. The 350 ft buffer is explained in more detail in Chapter V Section 5.5.

Advantages and Disadvantages

The advantage of the Look-Down method is that it is extremely fast because there is only one path constraint per collocation point. However, its disadvantage is that although the aircraft is guaranteed to be above the terrain, no lateral separation from terrain is guaranteed. It is possible to have a mountain slope right next to the aircraft. This is not desirable, therefore an alternative method is proposed called Left-Center-Right (LCR) Look-Down.

4.2.1.2 LCR Look-Down.

Because the Look-Down constraint only ensures a vertical separation above the terrain, it is possible that the aircraft could be safely above the terrain below it, but dangerously close to a sheer or steep surface laterally. Therefore, additional lateral constraints were considered. The LCR Look-Down method incorporates two additional path constraints compared to Look-Down, adding the left and right look-down constraints. Once the NLP solver creates a candidate optimal path, parallel paths offset to the left and to the right are created within the NLP solver as shown in Equations 60 where x^* and y^* are the optimal aircraft path and χ^* is the optimal aircraft heading. The lateral offset distance 'buffer' is the same distance as the vertical terrain buffer.

$$x_{left} = x^* + \text{buffer} * \cos(\chi^* + \frac{\pi}{2}) \tag{60a}$$

$$x_{right} = x^* - \text{buffer} * \cos(\chi^* + \frac{\pi}{2}) \tag{60b}$$

$$y_{left} = y^* - \text{buffer} * \sin(\chi^* - \frac{\pi}{2}) \tag{60c}$$

$$y_{right} = y^* + \text{buffer} * \sin(\chi^* - \frac{\pi}{2}) \tag{60d}$$

Then the 'griddedInterpolant' function representing the surface approximation is queried for the left and right constraint paths as well as at x^*, y^*, and the clearance for the left and right paths is calculated as detailed in Appendix B. The clearance values for the left and right paths are compared to a terrain buffer of 0 ft as shown in Equations (61) and (62). Hence, the left and right path constraints ensure that the

left and right parallel paths are always greater than or equal to the terrain height.

$$Clearance_{left} \geq 0 \text{ ft} \tag{61}$$

$$Clearance_{right} \geq 0 \text{ ft} \tag{62}$$

Advantages and Disadvantages

The disadvantage of the LCR Look-Down method is that the constraints are in three discrete locations, so it is possible that with higher resolution DTED (where post spacing is smaller than the lateral buffer), there could be a post that falls in between the aircraft optimal path and the left or right constraint check (within the accuracy of the interpolation method). But with the current implementation of SRTM-1 this is not as much of an issue. The main advantage of the LCR Look-Down method is that it is computationally fast while still giving extra collision protection to the left and right of the aircraft. Overall, the LCR Look-Down constraint method is a good combination of speed and accuracy. It provides a greater measure of protection than the Look-Down method since LCR Look-Down incorporates the additional lateral constraints. With LCR Look-Down there are now three constraint checks per collocation point with only a modest decrease in computational speed.

4.2.2 Aircraft Buffer Method.

An alternative method to adding a buffer to the terrain is to add a buffer around the aircraft. The method explored for this research is accomplished by adding a sphere-shaped buffer around the aircraft.

4.2.2.1 Sphere Method.

In this method, a sphere buffer is utilized around the aircraft using the equation $(x-x_{ac})^2+(y-y_{ac})^2+(z-z_{ac})^2=r^2$ where $x_{ac},\ y_{ac}$, and z_{ac} are the aircraft location at the center of the sphere and r is the radius of the sphere, which is the buffer.

The aircraft buffer acts in the same way as the terrain buffer in that both have the purpose of protecting the aircraft from terrain. However, in the terrain buffer case the discrete aircraft location vector is being compared to a continuous and differentiable surface approximation of the terrain plus the buffer. In the aircraft buffer method, the continuous sphere buffer is being compared to the discrete SRTM DTED posts.

First, it is necessary to parse the SRTM DTED database down to just the SRTM DTED posts that lie below (in the shadow) of the aircraft sphere. This will speed up computation time by only considering the posts of interest. A 350 ft sphere radius (buffer) is used for this research, exactly the same as the terrain buffer, $d_{threshold}$. With the 350 ft sphere radius and SRTM DTED Level 1, a minimum of four and maximum of seven posts can be captured at each collocation point as depicted in Figure 25.

"A minimum of four captured posts is ideal because they will essentially cover a quadrant of terrain below the aircraft and will minimize missing large deviations in the height of terrain" [91]. SRTM DTED Level 1 has a nominal post spacing (at the equator) of ∼90 m which is ∼295 ft post spacing. It makes sense that the diameter of the sphere must be greater than this post spacing in order to ensure that a minimum of one SRTM-1 post is captured in the shadow of the sphere. In this specific example, the sphere diameter must be greater than 295 ft, thus the sphere radius must be greater than 147.5 ft. Note that at the latitude and longitude of the Southern California terrain used in this research, the posts in latitude are spaced at 92.45 m (303.3 ft) apart South to North, while the posts in longitude are spaced

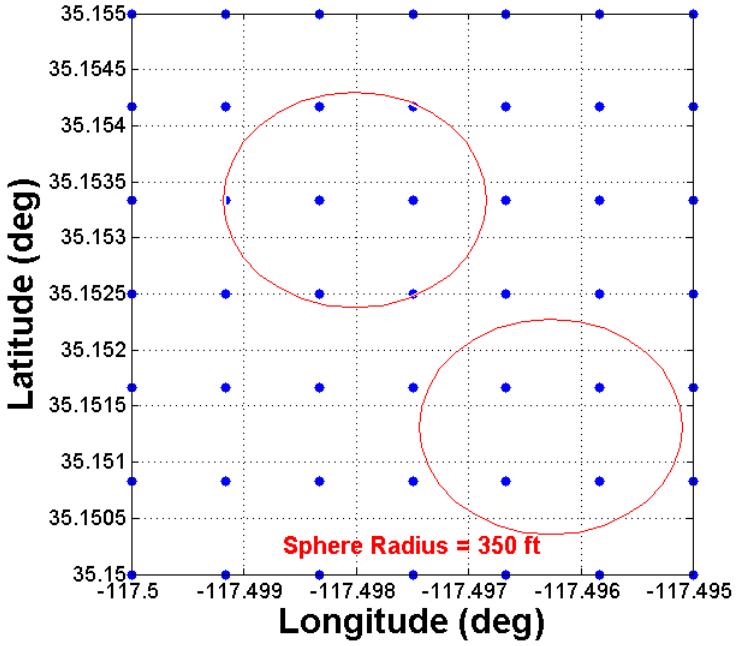


Figure 25. SRTM DTED Level 1 with Four to Seven Captured Posts in Sphere Shadow

75.98 m (249.3 ft) apart West to East due to the 35.1° N location. These latitude and longitude spacings were calculated using the reference vector information provided for this 1° by 1° SRTM DTED Level 1 tile (that is referenced by its Southwest corner of -117.5° W and 35.1° N).

Since the majority of operational low levels are now flown at 500 ft AGL, SRTM DTED level 1 can be used in a large majority of anticipated operational missions. However, SRTM DTED Level 2 should be available for contingency purposes when flight below 500 AGL is desired. "It is recommended that DTED Level 2 only be loaded on memory-limited aircraft if flight below 300 ft AGL is anticipated or desired" [91].

The next step in determining whether a collision would occur is to evaluate the SRTM DTED posts in the shadow on the ground against the sphere itself. To do this, an algorithm was developed to compare the position of the sphere to the discrete SRTM DTED posts using the inside-outside function developed by Barr [8]. The

equation of a sphere with radius r and aircraft position $(x_{ac},\ y_{ac},\ z_{ac})$ is shown in Equation (63).

$$\frac{(x - x_{ac})^2}{r^2} + \frac{(y - y_{ac})^2}{r^2} + \frac{(z - z_{ac})^2}{r^2} = 1 \tag{63}$$

For each propagated aircraft position all points on the sphere will satisfy Equation (63). Next, each candidate SRTM DTED post position $(x_{DTED},\ y_{DTED},\ z_{DTED})$ is checked against the sphere using Equation (64) which incorporates Barr's inside-outside function [8]

$$\frac{(x_{DTED} - x_{ac})^2}{r^2} + \frac{(y_{DTED} - y_{ac})^2}{r^2} + \frac{(z_{DTED} - z_{ac})^2}{r^2} = S \tag{64}$$

where S is the normalized radius of the sphere. Equation (64) allows any point in space to be evaluated relative to the sphere.

$$\text{If } \begin{cases} S = 1, (x_{DTED}, y_{DTED}, y_{DTED}) \text{ is on the surface.} \\ S > 1, (x_{DTED}, y_{DTED}, y_{DTED}) \text{ lies outside the sphere.} \\ S < 1, (x_{DTED}, y_{DTED}, y_{DTED}) \text{ lies inside the sphere.} \end{cases} \tag{65}$$

As shown in Equation (65), values of S greater than or equal to 1 indicate that the top of the DTED post lies outside or on the sphere, thus the constraint is $S \geq 1$ for all DTED posts. Equation (64) was put in matrix form to quickly determine if the top of a DTED post is located inside of the sphere buffer as shown in Equations (66) to (68) [91]. The index i represents each collocation point on the aircraft trajectory, and the index j represents each SRTM DTED post.

$$Q_{ij} = \left[(x_{DTED_j} - x_{ac_i}) \ \ (y_{DTED_j} - y_{ac_i}) \ \ (z_{DTED_j} - z_{ac_i})\right] \tag{66}$$

$$N = \begin{bmatrix} \frac{1}{r^2} & 0 & 0 \\ 0 & \frac{1}{r^2} & 0 \\ 0 & 0 & \frac{1}{r^2} \end{bmatrix} \tag{67}$$

Then a collision occurs IF and only IF

$$Q_{ij} N Q_{ij}^T < 1 \ \forall \ i, j. \tag{68}$$

If the QNQ^T term is less than 1 then a SRTM DTED post is located inside the sphere buffer, indicating a collision. This calculation is performed for each DTED post in the shadow of the sphere at each collocation point.

Advantages and Disadvantages

With the 350 ft buffer as the sphere radius, four to seven SRTM DTED posts are in the sphere shadow at any given aircraft position. Hence, for each collocation point, four to seven constraints exist. Within the NLP solver these constraint evaluations check if the tops of the four to seven SRTM DTED posts are located inside the protective sphere buffer at each collocation point. In the NLP solver, if the tops of any of the posts are located inside the sphere (i.e. path constraint violated) at any collocation point then another iteration must be generated to satisfy the constraints which requires another iteration of gradient checks. This process is repeated until none of the SRTM DTED post tops lie inside the sphere and an optimal control and optimal path are found. Although tractable, this process is time-consuming and impractical in an NLP solver. This is due to the requirement to find which SRTM DTED posts lie in the shadow of the sphere which includes if/then statements which are not suitable for use in an NLP. Therefore, the sphere method is problematic to implement in the NLP solver.

For the Multi-Trajectory Auto GCAS with no NLP solver involved, Trombetta [91]

has implemented the sphere buffer successfully and has demonstrated that the method keeps the aircraft avoidance path out of terrain. The logic works as designed and is accurate. This research demonstrates that the terrain buffer method also successfully successfully keeps the aircraft avoidance path clear of terrain. However, the sphere method is not an option to implement inside the NLP solver, hence, cannot be used for this research. Therefore, due to its ease of implementation and fast computational speed, the terrain buffer with LCR Look-Down method is recommended and is used in the research herein.

4.3 Summary

Chapter IV answered two important questions in order to formulate the 3-D Auto GCAS optimal control solution technique: 1) how to best approximate the digital terrain and 2) how to best implement the path constraint.

After reviewing different alternatives, this chapter provided the rationale for modeling the digital terrain as a continuous and differentiable surface approximation using a spline interpolation of discrete DTED posts and for implementing the path constraint as a terrain buffer using a LCR Look-Down method approach. Chapter V presents the 3-D Auto GCAS methodology and results.

V. 3-D Results with Two Controls

Chapter I presented the motivation and background for the Auto GCAS problem and showed that the requirement to auto maneuver close to the ground at the last second makes this problem challenging. The pertinent literature was reviewed in Chapter II, highlighting that a gap exists for ground collision avoidance in that existing Auto GCAS systems use pre-planned trajectories and optimal control has only been applied as a solution method to a counter-hijack problem but not to a last-second Auto GCAS problem. Chapter III defined the aggressive and timely requirements, presented the mapping of aggressive and timely to the optimal control formulation, and introduced the methodology for the two Auto GCAS optimal control formulations. In Chapter IV, two specific pieces of a 3-D optimal control implementation for Auto GCAS were developed: 1) the terrain surface was modeled as a continuous and differentiable approximation, and 2) the path constraint chosen was an LCR Look-Down constraint with a terrain buffer. This chapter illustrates the method and presents results for the 3-D case for an aircraft with two control inputs in some representative scenarios. To do this, it is necessary to understand the terrain used in this research as well as the flight performance of the aircraft models used for simulation. The terrain classification and aircraft performance are described because they necessary inputs to determine the length of the time horizon necessary for the optimizer. Once the time horizons are presented, the five-state and two-control simplified point-mass model is described along with the Min Control cost functional and aggressive trigger. Next, the 3-D low-level scenarios are introduced and results are presented for representative scenarios followed by a limited study to determine the effect of weighting the bank angle μ and load factor N_z terms in the cost functional. Finally, a comparison of the Optimal Auto GCAS to Multi-Trajectory Auto GCAS is presented.

5.1 Terrain Used in the Research

Freemont Peak (referred to as GCAS Mountain (Mt) by the Auto GCAS test team) has been used repeatedly for F-16 Auto GCAS testing at Edwards AFB for three decades. For commonality, this research also uses the GCAS Mt test area for simulations for the Optimal Auto GCAS method. Figures 26 and 27 depict a surface plot of the SRTM DTED Level 1 for the GCAS Mt test area in the East-North-Up (ENU) reference frame used in this research. The origin (0,0) in the Southwest corner equates to (35.1° N, 117.5° W) in latitude and longitude. Figure 26 more accurately depicts the scaling of the terrain height in relation to the x and y-axes and is a rotated view to highlight the valley leading up to GCAS Mt which is used for the run-ins for the simulations. The blue lines show the aircraft pilot paths for the 'high' and 'low' altitude low-levels at a 286° compass heading. In Figure 27 the z-axis is exaggerated in order to highlight the elevation change of GCAS Mt and to more easily see the plotted optimal trajectories for the scenarios in the remainder of the document.

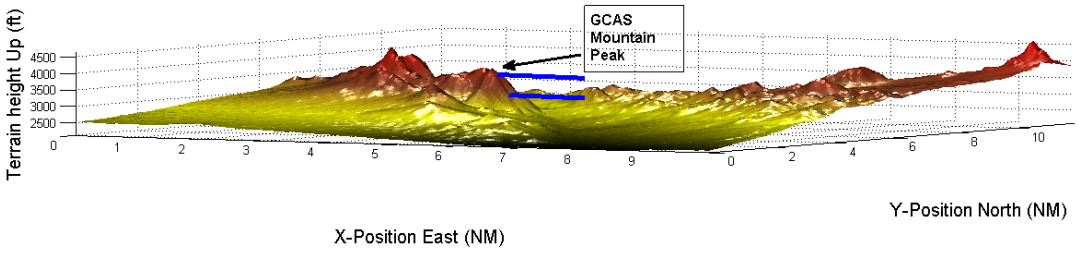


Figure 26. GCAS Mountain Test Area in ENU Rotated to Show Valley

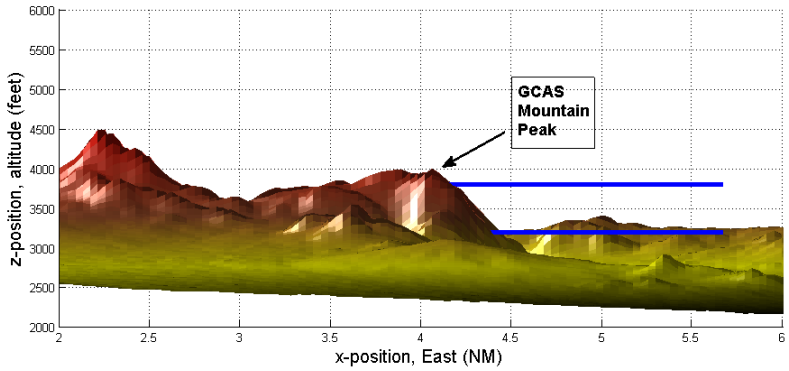


Figure 27. GCAS Mountain Test Area in ENU, Z-axis Zoomed

5.2 Terrain Classification

Before the optimal control formulation can be run, an appropriate time horizon must be chosen. In the case of Auto GCAS, the time horizon depends both on the terrain in which the aircraft is flying and on the aircraft's performance. The terrain type is important because a shorter time horizon could be sufficient for flight over flat terrain, while a longer time horizon would be necessary for flight over mountainous terrain with steep inclines. Therefore, classification of terrain is necessary and is covered in this section. Dragut determined that three classifications for terrain were adequate and named them Upland, Midland, and Lowland. Trombetta combined Dragut's three hierarchical levels of Upland, Midland, and Lowland [26] with the current Air Force classification structure (as defined in Air Force Instruction 11-202 Vol. 3 [92]) to maintain some relation to current Air Force standards [91]. The Air Force categorizes terrain in two levels, as either mountainous or non-mountainous. The Air Force's definition of mountainous terrain is terrain with $\geq$ 500 ft change in altitude in $\frac{1}{2}$ NM [92]. This mountainous definition will remain unchanged, but Trombetta [91] calls it Upland to be consistent with Dragut's three levels [26]. In order to define the remaining two classification levels, Midland and Lowland, Trombetta halved the criteria for mountainous terrain and the resulting Terrain Class Definitions are shown in Table 10 [91].

Table 10. Terrain Classification Based on Terrain Height Data [91]

Terrain Class Definitions
500 ft per $\frac{1}{2}$ NM $\leq$ **Upland**
250 ft per $\frac{1}{2}$ NM $\leq$ **Midland** $<$ 500 ft per $\frac{1}{2}$ NM
Lowland $<$ 250 ft per $\frac{1}{2}$ NM

5.3 Aircraft Performance

Next, the performance of candidate military heavy aircraft is discussed. Correspondence with test pilots in the 418^{th} and 419^{th} Flight Test Squadrons (FLTS) at Edwards AFB provided the operational low-level airspeed and altitudes for the aircraft of interest [21, 60]. Both the 418^{th} and 419^{th} FLTS have personnel who specialize in low-level missions in order to conduct developmental flight test for United States Air Force cargo and bomber aircraft. The C-5 Galaxy and the B-2 Spirit are purposefully excluded from the table since neither currently have a low-level mission [21, 60]. Table 11 shows typical low-level airspeeds (in knots ground speed (KGS)) and altitude ranges (in AGL) flown for each aircraft type. The turn radius and turn rate are based on level turns.

Table 11. Military Aircraft Low Level Maneuvering Flight Performance [21, 60]

Aircraft	Airspeed (KGS)	Altitude AGL (ft)	Nz (g)	Bank Angle (deg)	Turn Radius (ft)	Turn Rate ($\frac{deg}{sec}$)
C-130	210	300-500	2	60	2,254	9.01
C-17	310	300-500	2	60	4,913	6.10
B-52	350	500	2	60	6,262	5.41
B-1	540	500	2	60	14,906	3.50

The turn radius R and turn rate ω are calculated using Equations (69) and (70) [99]

$$\mathrm{R} = \frac{V^2}{g\sqrt{N_z^2 - 1}} \tag{69}$$

$$\omega = \frac{V}{R} \tag{70}$$

where g is defined as the acceleration of gravity, N_z is normal acceleration from Table 11, and V is the low-level airspeed from Table 11. By examining Table 11, three easily separable speed classifications are apparent: low (C-130), medium (C-17/B-52), and high-speed (B-1). For this research, scenarios are run for a representative

low-speed heavy at 210 KGS and a medium-speed heavy at 310 KGS. As an initial proof-of-concept study, the model used for this research is a simplified point-mass model rather than the actual high-fidelity aircraft-specific models.

The data in Table 11 is used as a basis for Trombetta's Path Propagation Criteria to determine appropriate time horizons [91]. Specifically, the low-level airspeed is used to determine how far forward an avoidance path needs to be propagated such that the aircraft will achieve an altitude climbed of 4,000 ft (for terrain clearance) [91]. Next, the turn radius and rate are used to ensure that a minimum 90° of turn is possible in order to clear terrain laterally [91]. The Path Propagation Criteria are discussed in more detail in Section 5.4.

5.4 Time Horizons

Now that terrain classification and aircraft performance have been described, the time horizons can be determined. Using the Path Propagation Criteria developed by Trombetta, appropriate time horizons for the optimizer were determined for each candidate aircraft and terrain type [91]. First the length (ground distance) of the propagated path for the Upland case was determined based on the time to climb 4,000 feet. The rest of the criteria are derived from this first measurement. The time horizon is the maximum of either the time it takes for the aircraft to turn 90° (in a 2-g/60° bank) in order to clear terrain laterally (t_{90°) or the time for the forward path (2-g pull) to out-climb the terrain in front of it ($t_{fwd\ path}$) as shown in Equation (71) [91]

$$\text{Path Propagation Time} = \max(t_{90^\circ}, t_{fwd\ path}). \tag{71}$$

For the lateral paths, it is only necessary for the aircraft to achieve 90° of turn to avoid the terrain forward of the aircraft's 3-9 line [91]. Hence, the lateral paths are relatively insensitive to the slope of the terrain in front of the aircraft [91].

The time horizons recommended by Trombetta to be used for the optimizer for the low, medium, and high-speed heavy for each terrain type are shown in Table 12. For this research, only the Upland scenarios will be analyzed because the digital terrain utilized for this research meets the Upland criteria as outlined in Table 10. Therefore, 45.0, 31.0, and 28.5 sec will be used as the time horizons for the low, medium, and high-speed heavy cases, respectively.

Table 12. Time Horizons Used for Low, Medium, and High-Speed Heavies [91]

Low-Speed Heavy (velocity = 210 KGS)			
	Lowland	**Midland**	**Upland**
Path Propagation Time (sec)	15.5	29.5	45.0
Medium-Speed Heavy (velocity = 310 KGS)			
	Lowland	**Midland**	**Upland**
Path Propagation Time (sec)	17.5[17]	21.5	31.0
High-Speed Heavy (velocity = 540 KGS)			
	Lowland	**Midland**	**Upland**
Path Propagation Time (sec)	28.5[17]	28.5[17]	28.5[17]

[17] Time to turn 90°, t_{90°, was the driving factor for the Path Propagation Time

To design an Auto GCAS for any candidate aircraft, Trombetta's Path Propagation Criteria will need to be applied for the specific mission airspeed, terrain type, load factor N_z and bank angle μ for the Auto GCAS recovery, and max flight path angle γ for that aircraft. The methodology to apply Trombetta's criteria is detailed in Appendix B and more detail on the criteria can be found in Trombetta's research [91].

5.5 Terrain Buffer and Minimum Clearance Altitude

The terrain buffer, $d_{threshold}$, is the inequality path constraint for Auto GCAS as shown in Equation (35). 500 ft AGL was chosen as the low-level scenario for this research because it is uncommon for heavy or fighter type aircraft to fly below 500 ft AGL, although for certain operational missions 300 ft AGL missions are authorized. Discussions with operational pilots regarding a 500 ft AGL low level determined that a terrain buffer of 350 ft AGL in mountainous (Upland) terrain would be acceptable to the pilot community. Therefore, $d_{threshold} = 350$ ft AGL is what is used for this research. However, this threshold should be selected based on mission requirements. The terrain buffer takes into account all of the error plus the desired level of safety (minimum clearance altitude). Hence, the terrain buffer is comprised of the min clearance altitude and the worst-case error.

$$\text{Terrain Buffer} = \text{Min Clearance Altitude} + \text{Worst-Case Error} \tag{72}$$

For this research, notional but representative numbers are used for the errors that make up the worst-case error. The worst-case error is comprised of two types of error, ground error and aircraft error. Worst-case ground error is comprised of DTED error, interpolation error, and a tree buffer. The interpolation error is based on a limited study done in this research in Chapter IV Section 4.1. As explained in Chapter II, the SRTM radars sometimes only captured partial canopy heights, therefore the tree buffer is included to account for trees and power lines not captured in the SRTM DTED and is based on an estimate of actual tree heights versus the SRTM DTED

terrain heights in the same area.

$$\begin{aligned} \text{DTED Error} &= +30 \text{ ft} \\ \text{Interpolation Error} &= +15 \text{ ft} \\ \text{Tree Buffer} &= +70 \text{ ft} \\ \text{Worst-case Ground Error} &= 30 \text{ ft} + 15 \text{ ft} + 70 \text{ ft} = +115 \text{ ft} \end{aligned} \tag{73}$$

Worst-case aircraft error is comprised of GPS vertical error [25] and trajectory error[18].

$$\begin{aligned} \text{GPS Vertical Error} &= \pm 45 \text{ ft} \\ \text{Trajectory Error} &= \pm 50 \text{ ft} \\ \text{Worst-case Aircraft Error} &= \pm 45 \text{ ft} \pm 50 \text{ ft} = \pm 95 \text{ ft} \end{aligned} \tag{74}$$

Summing the worst-case ground and aircraft error in Equations (73) and (74) yields a total worst-case error of 210 ft.

$$\text{Worst-Case Error} = \text{Worst-Case Ground Error} + \text{Worst-Case Aircraft Error} \tag{75}$$

$$= 115 \text{ ft} + 95 \text{ ft} = 210 \text{ ft} \tag{76}$$

Therefore, for a 350 ft terrain buffer with a worst-case error of 210 ft, the guaranteed min clearance altitude will be 140 ft.

$$\text{Min Clearance Altitude} = 350 \text{ ft} - 210 \text{ ft} = 140 \text{ ft} \tag{77}$$

Given the notional error values used for this research, this means that the aircraft is expected never to be closer than 140 ft to the actual terrain during an automatic

[18]The trajectory error is an estimate of the difference in actual aircraft performance versus the model in the vertical.

recovery. An analysis similar to the above with the appropriate error values would have to be completed for other candidate aircraft.

5.6 Simplified 3-DOF Point Mass Model (5 States/2 Controls)

Chapter II presented the 3-DOF point mass model in Equations (17)-(21). National Aeronautics and Space Administration's (NASA's) Small UAV Auto GCAS test team concluded that winds are not necessary to consider for systems where the winds are less than 15% of the slowest airspeed for the given platform, "and the remaining wind effects can be accounted for as part of other uncertainties" [83]. Further, "if the expected winds that may be encountered are greater than 30% of the slowest airspeed, it is likely that some sort of wind estimate will be needed" [83]. These two rules-of-thumb are based on lessons learned from both the Small UAV Auto GCAS and F-16 Auto GCAS flight tests. Note that the F-16 flies fast enough that no wind estimates needed to be incorporated into the F-16 Auto GCAS algorithm. Thus, there exists a gray area in between 15% and 30% of the slowest airspeed. For this research, wind estimates are not included in the model, which is a reasonable assumption for the relatively fast military heavy-type aircraft studied herein. Therefore, assume $v_{w_x} = v_{w_y} = v_{w_z} = 0$ in Equations (9), (10), and (11), and the simplified 3-DOF point mass model equations of motion used for this research reduce to

$$\dot{x} = V \cos\gamma \cos\chi \tag{78}$$

$$\dot{y} = V \cos\gamma \sin\chi \tag{79}$$

$$\dot{z} = V \sin\gamma \tag{80}$$

$$\dot{\gamma} = \frac{N_z \cos\mu - g\cos\gamma}{V} \tag{81}$$

$$\dot{\chi} = \frac{N_z \sin\mu}{V\cos\gamma}. \tag{82}$$

How to best calculate winds aloft and then how to incorporate these wind estimates into an Auto GCAS is an area for further research included in Chapter VI.

5.7 Auto GCAS 3-D Min Control Formulation

The Auto GCAS Min Control with an Aggressive Trigger optimal control formulation is described in Chapter III, Section 3.4.2. As discussed in Chapter III, the Auto GCAS Min Control formulation is a fixed final time, free final state minimization.

5.7.1 Min Control Cost Functional.

For this specific 3-D scenario, the control vector $\mathbf{u}$ is comprised of two controls, bank angle μ and load factor N_z. For the Min Control cost functional, both controls are minimized to their level-flight trim condition. Hence, bank angle is minimized to 0° while the load factor is minimized to 1 g. Next, the controls are normalized such that when either control is at a maximum (or minimum) value its contribution to the cost is equal to one. Thus, the general Min Control cost functional given in Equation (34) becomes

$$\underset{\mathbf{u}\in\mathbf{A}}{\text{Minimize: }} J = \int_{t_0}^{t_f} \begin{bmatrix} \mu(t) & N_z(t)-1 \end{bmatrix} \mathbf{R} \begin{bmatrix} \mu(t) \\ N_z(t)-1 \end{bmatrix} \, \mathrm{d}t \tag{83}$$

where the control weighting matrix $\mathbf{R}$ is defined as

$$\mathbf{R} = \begin{bmatrix} \frac{r_1}{(\mu_{max})^2} & 0 \\ 0 & \frac{r_2}{(N_{z_{max}}-1)^2} \end{bmatrix} . \tag{84}$$

For a majority of the scenarios, an equal weighting of $r_1 = r_2 = 1$ is used. Additionally, results from a limited study of different weighting matrices $\mathbf{R}$ are provided in Section 5.8.2.

The matrix form of the cost functional in Equations (83) and (84) is equivalent to

$$\underset{\mathbf{u}\in\mathbf{A}}{\text{Minimize: }} J = \int_{t_0}^{t_f} r_1 \left(\frac{\mu(t)}{\mu_{max}}\right)^2 + r_2 \left(\frac{N_z(t)-1}{N_{z_{max}}-1}\right)^2 \, \mathrm{d}t \tag{85}$$

where $\mu_{max} = \frac{\pi}{3}$ and $N_{z_{max}} = 2$ g, and the bounds on the controls are

$$-\frac{\pi}{3} \leq \mu(t) \leq \frac{\pi}{3} \tag{86}$$

$$0 \; g \leq N_z(t) \leq 2 \; g. \tag{87}$$

With the squared terms in Equation (85), the NLP solver would minimize $\mu(t)$ to zero and minimize $N_z(t)$ to one. To improve NLP performance, values were scaled to be symmetric about zero and on the order of one. Therefore, $N_z(t)$ is shifted to be from -1 g to 1 g and a variable $N_{z_{Scaled}}(t)$ is defined as $N_{z_{Scaled}}(t) = N_z(t) - 1$. The cost functional in Equation (85) becomes

$$\underset{\mathbf{u}\in\mathbf{A}}{\text{Minimize: }} J = \int_{t_0}^{t_f} r_1 \left(\frac{\mu(t)}{\mu_{max}}\right)^2 + r_2 \left(\frac{N_{z_{Scaled}}(t)}{N_{z_{Scaledmax}}}\right)^2 \, \mathrm{d}t \tag{88}$$

where $N_{z_{Scaledmax}} = 1$ g, and the bounds on $N_{z_{Scaled}}(t)$ are

$$-1 \; g \leq N_{z_{Scaled}}(t) \leq 1 \; g. \tag{89}$$

Notice that $N_{z_{Scaled}}(t)$ is only used in the cost functional J and that $N_z(t)$ is used in Equations (78) to (82) in order to satisfy the equations of motion.

5.7.2 Aggressive Trigger.

Per the definition of aggressive in Chapter III, an aggressive trajectory is one where at least one control is at its min or max limit $\forall \; t \in [t_0, t_{CPA})$. Any combination of controls that satisfy this results in an aggressive maneuver. As explained in Chapter

III, every solution admitted by the Min Control with an Aggressive Trigger solution is timely. When the min control solution admitted by the optimizer meets the aggressive criteria, then the auto recovery will be triggered. For the two controls μ and N_z used in this research, the general aggressive trigger criteria given in Equation (36) become

$$\left.\begin{array}{c} \mu(t) = \mu_{\max} \\ \text{or} \\ \mu(t) = \mu_{\min} \\ \text{and/or} \\ N_z(t) = N_{z_{\max}} \\ \text{or} \\ N_z(t) = N_{z_{\min}} \end{array}\right\} \forall\ t \in [t_0, t_{CPA}). \tag{90}$$

Because the optimal solutions are calculated at a discrete 0.5 sec update rate (2 Hz) for these 3-D scenarios, the aggressive trigger criteria are only approximated. For these simulations, these criteria in Equation (90) are approximately satisfied by triggering the last converged optimal solution. This allows getting arbitrarily close to the theoretical optimal solution, which often lies between runs.

5.8 3-D Results

Several cases were run to illustrate the performance of the Min Control with an Aggressive Trigger methodology. First, the medium-speed heavy results are explained for four representative low-level scenarios, two at ∼1,000 ft AGL and two at ∼500 ft AGL where the symbol ∼ means approximately. Next, one scenario is run for the medium-speed heavy using different weighting matrices $\mathbf{R}$ with the purpose of determining the most appropriate control weightings and to show the flexibility of the cost functional tailored to meet mission needs for climbing or turning performance

constraints. Finally, a comparison is run between the Optimal Auto GCAS solution and the Multi-Trajectory Auto GCAS solution for the low-speed heavy.

5.8.1 Medium-Speed Heavy Results.

5.8.1.1 3-D Test Scenarios.

For illustrative purposes, the medium-speed heavy is used for the majority of the scenarios. Two different altitude cases were run for the medium-speed heavy aircraft. Case 1 is a 'High' altitude low-level at ∼1,000 ft AGL and case 2 is a 'Low' altitude low-level at ∼500 ft AGL (see Table 13). Note that the aircraft initial MSL altitude is kept constant, not the AGL altitude.

Table 13. 3-D Test Scenarios

Medium-Speed Heavy - Case 1 'High'		
Airspeed (KGS)	**Altitude AGL (ft)**	**Init Flight Path Angle γ_0 (deg)**
310	∼1,000	0° (level)
310	∼1,000	-5° (dive)
Medium-Speed Heavy - Case 2 'Low'		
Airspeed (KGS)	**Altitude AGL (ft)**	**Init Flight Path Angle γ_0 (deg)**
310	∼500	0° (level)
310	∼500	5° (nose-high, level)

As shown in Table 13, for Case 1 two scenarios were run: one with an initial flight path angle of $\gamma_0 = 0°$ (level) and the other with $\gamma_0 = -5°$ (dive). This $\gamma_0 = -5°$ (dive) scenario was run for the 'High' scenario to show that the algorithm can also

protect the aircraft in a dive at low altitude. For Case 2, the $\gamma_0 = 0°$ level scenario was run as well as a $\gamma_0 = 5°$ (nose-high, level) scenario. This nose-high level scenario was run in place of a climb scenario because with a constant $\gamma_0 = 5°$ climb in altitude, the aircraft trajectory was steep enough such that the terrain was no longer in the time horizon of the optimizer.

For this section, an equal control weighting of $r_1 = r_2 = 1$ is used. For each simulation case, results are shown at each iteration (0.5 sec update rate, i.e. 2 Hz) in a tabular format as well as in a 'horizontal' and 'bird's-eye' view. The medium-speed heavy has a maximum flight path angle γ of 15°, N_z limits of 0 g/+2 g, and bank angle μ limits of ± 60°. The medium-speed heavy has a time horizon of 31 sec in Upland terrain based on its speed of 310 KGS from Table 12. 91 collocation points are used for all four scenarios. A new optimal solution is calculated every 0.5 sec along a straight-line pilot path from the initial conditions to the peak of GCAS Mountain located at $x = 4.09$ NM East, $y = 5.10$ NM North of the origin. With this 0.5 sec update rate (time step), the distance step between each run of the optimizer is ~ 260 ft for the medium-speed heavy.

Case 1 'High' Level

This scenario was selected to show that when the aircraft is flying just below but near the top of mountainous terrain, that the most likely a climbing maneuver will be the lowest cost. The initial conditions for this Case 1 'High' altitude and $\gamma_0 = 0°$ level

scenario are shown below

$$x_0 = 5.66 \text{ NM East} \tag{91}$$

$$y_0 = 4.72 \text{ NM North} \tag{92}$$

$$z_0 = 3,750 \text{ ft MSL} \tag{93}$$

$$\gamma_0 = 0^\circ \tag{94}$$

$$\chi_0 = 164^\circ \text{ (polar)}/286^\circ \text{ (compass)}. \tag{95}$$

Note that z_0, γ_0, and χ_0 remain constant for each iteration, while x_0 and y_0 change as the aircraft progresses further towards the GCAS Mountain peak. Table 14 shows the data for each run (each time the optimizer was run at a new initial condition) which include: the run number, t_0, t_{CPA}, the aggressiveness ratio, the max and min bank angle μ required, the max and min load factor N_z required, the NLP solver time, and the minimum vertical clearance for the left, center, and right constraint paths. The data for the run where the Auto GCAS recovery was triggered is colored orange.

Table 14. Data Table for Medium-Speed Heavy, Case 1 High, $\gamma_0 = 0°$

Run #	t_0 (sec)	t_{CPA} (sec)	Agg ratio	Bank max (deg)	Bank min (deg)	N_z max (g)	N_z min (g)	NLP time (sec)	Clr Lft (ft)	Clr Ctr (ft)	Clr Rgt (ft)
1	0	21.04	0.00	0.35	-0.00	1.12	1.00	2.145	269	350	383
2	0.5	20.55	0.00	0.21	-0.00	1.12	1.00	0.505	265	350	386
3	1.0	20.01	0.00	0.10	-0.00	1.13	1.00	0.427	266	350	387
4	1.5	19.05	0.00	0.02	-0.06	1.14	1.00	0.444	266	350	391
5	2.0	18.74	0.00	0.00	-0.53	1.14	1.00	0.580	276	350	405
6	2.5	18.32	0.00	0.01	-0.22	1.15	1.00	0.339	260	350	391
7	3.0	17.45	0.00	0.02	-0.56	1.16	1.00	0.351	255	350	393
8	3.5	16.91	0.00	0.02	-0.81	1.17	1.00	0.309	251	350	398
9	4.0	16.39	0.00	0.02	-1.03	1.18	1.00	0.291	252	350	399
10	4.5	15.95	0.00	0.02	-1.29	1.19	1.00	0.345	249	350	403
11	5.0	15.64	0.00	0.00	-1.51	1.21	1.00	0.342	247	350	407
12	5.5	14.84	0.00	0.00	-1.81	1.22	1.00	1.076	243	350	406
13	6.0	14.35	0.00	0.01	-2.36	1.24	1.00	0.348	238	350	407
14	6.5	11.25	0.00	0.01	-2.81	1.27	1.00	0.333	234	350	413
15	7.0	10.71	0.00	0.00	-3.29	1.30	1.00	0.280	238	350	414
16	7.5	10.19	0.00	0.00	-3.48	1.34	1.00	0.774	235	350	418
17	8.0	9.75	0.00	0.00	-3.46	1.38	1.00	0.334	238	350	418
18	8.5	9.44	0.00	0.00	-5.33	1.42	1.00	0.304	253	350	424
19	9.0	9.02	0.00	0.00	-6.56	1.47	1.00	0.871	254	350	428
20	9.5	8.64	0.00	0.00	-5.48	1.54	1.00	0.618	244	350	419
21	10.0	8.15	0.00	0.00	-5.39	1.61	1.00	0.702	250	350	412
22	10.5	7.61	0.00	0.01	-6.23	1.70	1.00	0.924	255	350	410
23	11.0	6.65	0.00	0.00	-4.66	1.83	1.00	0.685	264	350	399
24	11.5	6.34	0.00	0.01	-4.82	1.97	1.00	0.927	278	350	397
25	12.0	5.92	0.15	0.01	-5.23	2.00	1.00	0.751	283	350	397
26	12.5	5.05	0.56	0.14	0.00	2.00	0.96	0.725	322	350	383

Trigger criteria were approximately satisfied on Run 26 with t_{CPA} =5.05 sec, hence the auto recovery lasts 5.05 sec before control would be returned to the pilot. The green dots in Figure 28 show the collocation points on the triggered path while the green line in Figure 29 shows a top view of the triggered path. In Figures 28 and 29 only the optimal paths from the even iterations are plotted in order to de-clutter the figures. The aggressiveness ratio is a measure of how long the control was at a max

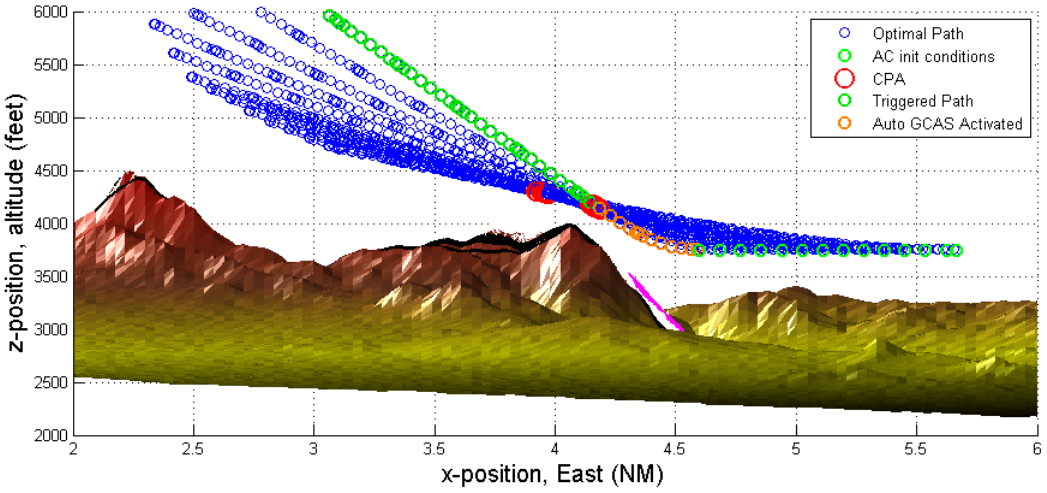


Figure 28. Optimal Path (26th Run), Med-Speed Heavy, Case 1 High, $\gamma_0 = 0°$

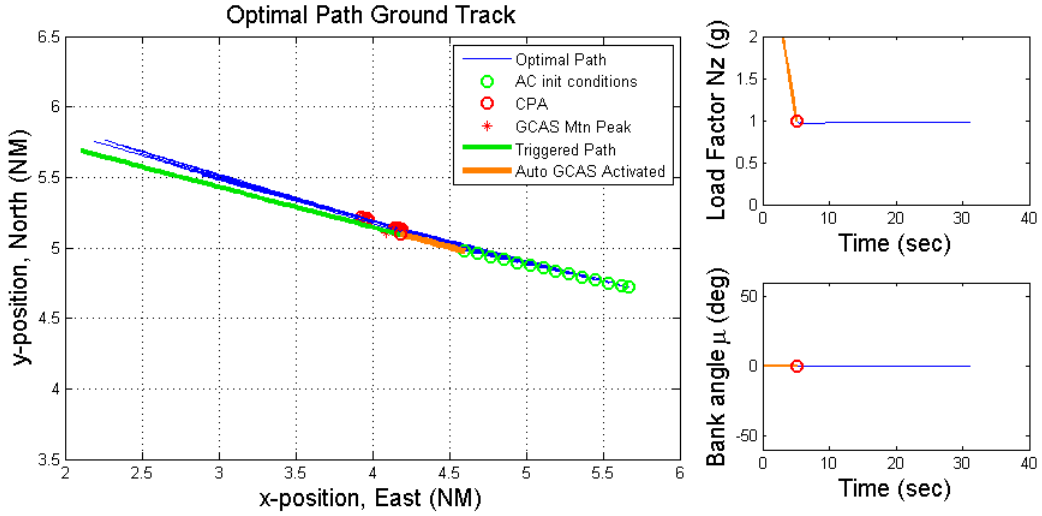


Figure 29. Optimal Path and Control (26th Run), Med-Speed Heavy, Case 1 High, $\gamma_0 = 0°$

or min limit compared to the t_{CPA} and is defined as

$$\text{Agg ratio } = \frac{\text{time at max/min}}{t_{CPA}}. \tag{96}$$

In the 2-D example with a single control the Agg ratio was 1 at trigger. However, in this 3-D example with two controls, the Agg ratio only approaches 1 at trigger. This is due to using a discrete number of 91 collocation points and marching forward in time at a discrete step size (update rate of 0.5 sec). As the step size is decreased and the number of collocation points is increased the closer the Agg ratio is to 1 at trigger. This is explored for a specific case and is presented at the end of this Section in Section 5.8.1.2 (refer to Figure 36). For this scenario the time at the max limit of 2 g was 2.83 sec, hence the Agg ratio = 0.56. The two control plots for N_z and μ for the triggered path show that the max N_z of 2 g was required for 2.83 sec but almost no bank was required. The trigger criteria of 2 g was approximately satisfied on Run 26 with the Agg ratio = 0.56 as shown in Table 14 with minimum vertical clearance for the left, center, and right paths are 322 ft, 350 ft, and 383 ft AGL. The NLP solver time for Run 26 was 0.725 sec, while the average NLP solver time was 0.547 sec (largest and smallest time thrown out and the rest averaged).

Case 1 'High' Dive

The Case 1 'High' low-level dive scenario is covered next. The initial conditions for this medium-speed heavy scenario with an initial $\gamma_0 = -5^\circ$ are listed below

$$x_0 = 5.66 \text{ NM East} \tag{97}$$

$$y_0 = 4.72 \text{ NM North} \tag{98}$$

$$z_0 = 3,800 \text{ ft MSL} \tag{99}$$

$$\gamma_0 = -5^\circ \tag{100}$$

$$\chi_0 = 164^\circ \text{ (polar)}/286^\circ \text{ (compass)} \tag{101}$$

Figures 30 and 31 show the side and top view for this dive scenario. Table 15 lists the data for each run until trigger. The aggressive trigger criteria of 2 g is approximately satisfied on Run 18 with a time at the max limit of 2 g of 3.25 sec and $t_{CPA} = 9.02$ sec, resulting in an Agg ratio = 0.36. Load factor $N_z = 2$ g was only required from 0 to 3.25 sec (vice until 9.02 sec), indicating that the theoretical optimal lies after Run 18 and before Run 19. At Run 19, the NLP solver reported that the max number of iterations (internal to the NLP) was achieved and that an optimal solution was not found. The max bank angle μ required was -12.92° (slight right bank). At trigger, z_0 = 3,410.8 ft MSL. The NLP solver time at Run 18 was 0.987 sec, while the average NLP solver time was 0.581 sec. The minimum vertical clearance values for the left, center, and right constraints were 282, 350, and 455 ft AGL. The 350 ft clearance for the center path shows that the auto recovery was timely, while the Agg ratio of 0.36 shows that the maneuver approximated the aggressive criteria.

Table 15. Data Table for Medium-Speed Heavy, Case 1 High, $\gamma_0 = -5°$

Run #	t_0 (sec)	t_{CPA} (sec)	Agg ratio	Bank max (deg)	Bank min (deg)	N_z max (g)	N_z min (g)	NLP time (sec)	Clr Lft (ft)	Clr Ctr (ft)	Clr Rgt (ft)
1	0	21.64	0.00	0.00	-5.91	1.27	1.00	1.412	211	350	507
2	0.5	16.91	0.00	0.00	-6.13	1.29	1.00	0.762	213	350	507
3	1.0	16.39	0.00	0.00	-7.29	1.31	1.00	0.588	213	350	515
4	1.5	15.95	0.00	0.00	-8.75	1.33	1.00	0.725	204	350	517
5	2.0	15.64	0.00	0.00	-9.86	1.36	1.00	0.623	198	350	503
6	2.5	15.22	0.00	0.00	-10.51	1.40	1.00	0.715	200	350	501
7	3.0	14.35	0.00	0.01	-12.31	1.42	1.00	0.687	186	350	499
8	3.5	13.81	0.00	0.01	-12.89	1.46	1.00	0.307	192	350	505
9	4.0	13.29	0.00	0.01	-13.19	1.51	1.00	0.306	202	350	509
10	4.5	12.54	0.00	0.00	-13.21	1.56	1.00	0.314	211	350	518
11	5.0	12.34	0.00	0.01	-13.54	1.62	1.00	0.338	215	350	523
12	5.5	11.74	0.00	0.01	-15.57	1.67	1.00	0.307	207	350	516
13	6.0	11.25	0.00	0.02	-16.39	1.73	1.00	0.276	212	350	522
14	6.5	10.71	0.00	0.03	-17.30	1.81	1.00	0.554	218	350	529
15	7.0	10.19	0.00	0.03	-17.77	1.90	1.00	0.366	229	350	522
16	7.5	9.75	0.01	0.02	-17.09	2.00	1.00	0.368	242	350	508
17	8.0	9.44	0.15	0.00	-2.42	2.00	0.96	1.353	317	350	396
18	8.5	9.02	0.36	0.01	-12.92	2.00	0.96	0.987	282	350	455

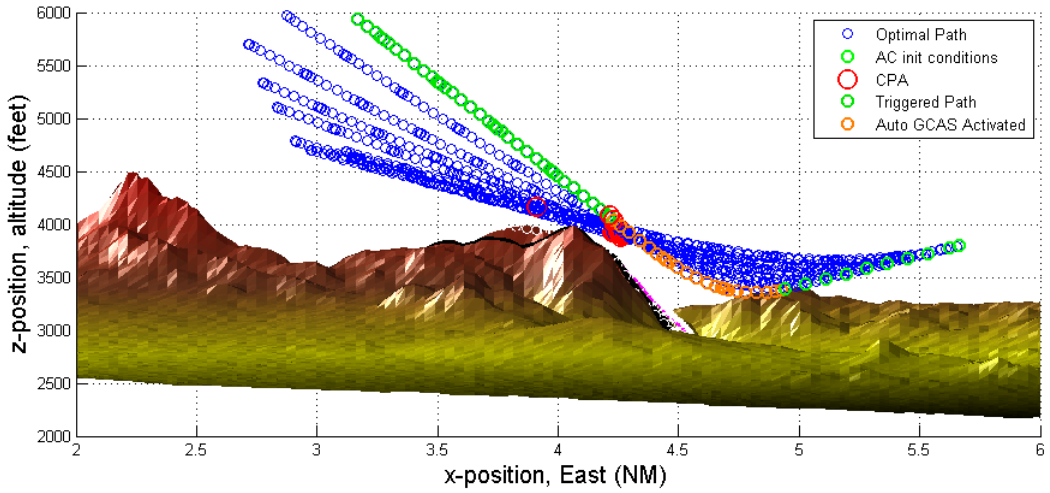


Figure 30. Optimal Path (18th Run), Med-Speed Heavy, Case 1 High, $\gamma_0 = -5°$ Dive

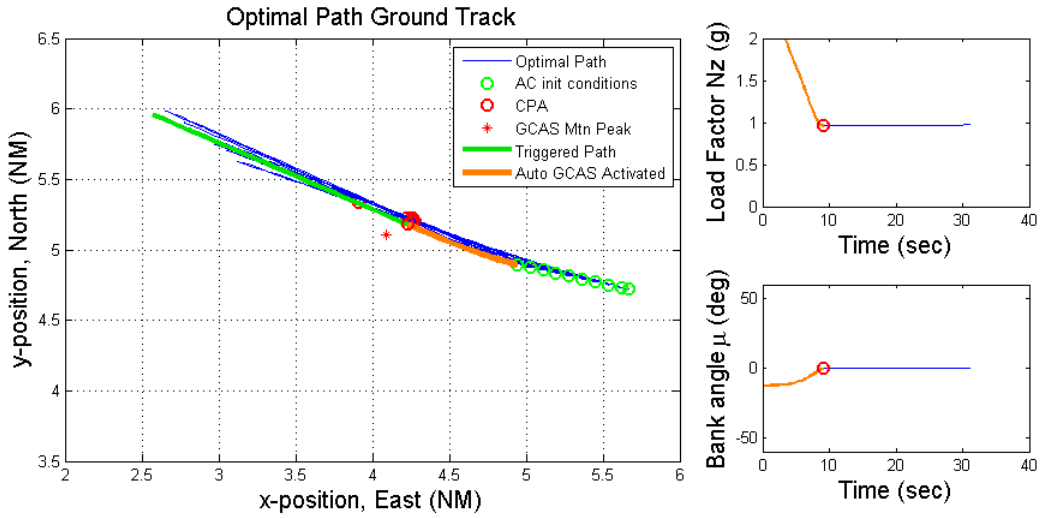


Figure 31. Optimal Path and Control (18th Run), Med-Speed Heavy, Case 1 High, $\gamma_0 = -5°$ Dive

Table 16 shows a comparison of the data for the Case 1 High altitude level and dive scenarios at trigger. The aircraft (x, y) initial conditions are identical for each of the runs in both the level and dive scenarios for comparison. The auto recovery is triggered four seconds earlier in the dive scenario (8.5 vs. 12.5 sec), while the t_{CPA} is four seconds longer (9.02 vs. 5.05 sec) indicating a longer Auto GCAS activation time, which makes sense due to the more demanding nature of the dive. The aggressiveness ratio for the dive scenario is lower with a value of 0.36 and the average NLP time for the dive is higher at 0.581 sec.

Table 16. Data Comparison of Case 1 High Alt Level and Dive Scenarios at Trigger

Scenario	**Trigger Point**	**Auto Activation Duration (t_{CPA})**	**Aggressiveness Ratio**	**NLP time average**
High Alt Level	Run 26 (12.5 sec)	5.05 sec	0.56	0.547 sec
High Alt Dive	Run 18 (8.5 sec)	9.02 sec	0.36	0.581 sec

Case 2 'Low' Level

This Case 2 'Low' altitude and $\gamma_0 = 0°$ level scenario has initial conditions identical to those of the medium-speed heavy Case 1 'High' level scenario given in Equations (91) to (95), except that $z_0 = 3,200$ ft MSL. The data for each run are given in Table 17. The green circles and green line represent the triggered auto recovery as shown in Figures 32 and 33. Aggressive criteria for trigger were approximately satisfied at Run 20, and $t_{CPA} = 7.09$ sec where control would be returned to the pilot. Load factor $N_z = 2$ g from 0 sec to 5.03 sec (vice until 7.09 sec) with an Agg ratio = 0.71 due to the 0.5 sec update rate. In other words, the theoretical optimal solution is somewhere after Run 20 and before Run 21 0.5 sec later. Note that at Run 21 (not shown) the NLP solver reported that the max number of iterations was achieved (4,000 iterations) without finding an optimal solution. The NLP time at trigger was 1.056 sec, while the average NLP time was higher at 1.137 sec. The left, center, and right minimum clearance values were 240 ft, 350 ft, and 448 ft AGL, respectively.

Table 17. Data Table for Medium-Speed Heavy, Case 2 Low, $\gamma_0 = 0°$

Run #	t_0 (sec)	t_{CPA} (sec)	Agg ratio	Bank max (deg)	Bank min (deg)	N_z max (g)	N_z min (g)	NLP time (sec)	Clr Lft (ft)	Clr Ctr (ft)	Clr Rgt (ft)
1	0	17.45	0.00	0.00	-6.67	1.24	1.00	1.991	236	350	532
2	0.5	16.91	0.00	0.00	-2.43	1.27	1.00	0.567	284	350	473
3	1.0	16.91	0.00	0.00	-3.13	1.29	1.00	0.661	283	350	482
4	1.5	16.39	0.00	0.00	-2.24	1.31	1.00	1.157	293	350	458
5	2.0	15.64	0.00	0.00	-5.41	1.33	1.00	1.038	285	350	506
6	2.5	15.22	0.00	0.00	-6.65	1.34	1.00	0.866	272	350	503
7	3.0	14.84	0.00	0.00	-4.71	1.37	1.00	0.907	287	350	480
8	3.5	13.81	0.00	0.00	-5.19	1.40	1.00	1.360	285	350	475
9	4.0	13.29	0.00	0.01	-6.24	1.43	1.00	0.904	284	350	475
10	4.5	13.29	0.00	0.00	-4.38	1.47	1.00	0.946	304	350	453
11	5.0	12.54	0.00	0.00	-1.35	1.52	1.00	1.195	298	350	409
12	5.5	12.12	0.00	0.00	-1.40	1.57	1.00	1.118	302	350	411
13	6.0	11.74	0.00	0.00	-3.38	1.61	1.00	1.378	318	350	423
14	6.5	11.25	0.00	0.01	-3.79	1.67	0.99	1.472	320	350	421
15	7.0	10.71	0.00	0.01	-3.40	1.75	0.98	1.124	326	350	416
16	7.5	10.19	0.00	0.01	-3.14	1.83	0.96	1.077	329	350	407
17	8.0	9.75	0.00	0.00	-3.32	1.96	0.96	1.339	326	350	400
18	8.5	9.02	0.16	0.00	-2.77	2.00	0.96	1.608	320	350	397
19	9.0	8.15	0.35	0.00	-13.95	2.00	0.96	1.260	285	350	445
20	9.5	7.09	0.71	0.25	-60.00	2.00	0.95	1.056	240	350	448

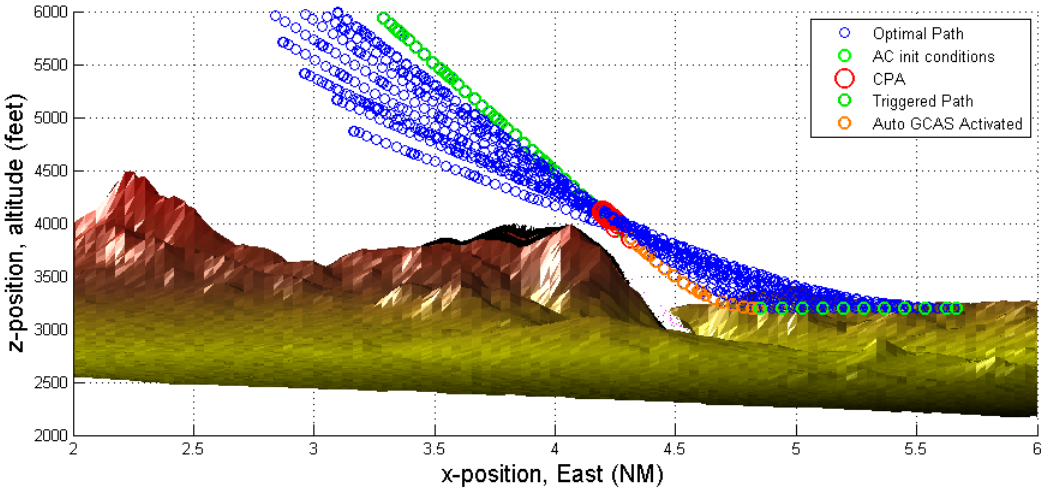


Figure 32. Optimal Path (20th Run), Med-Speed Heavy, Case 1 Low, $\gamma_0 = 0°$ Level

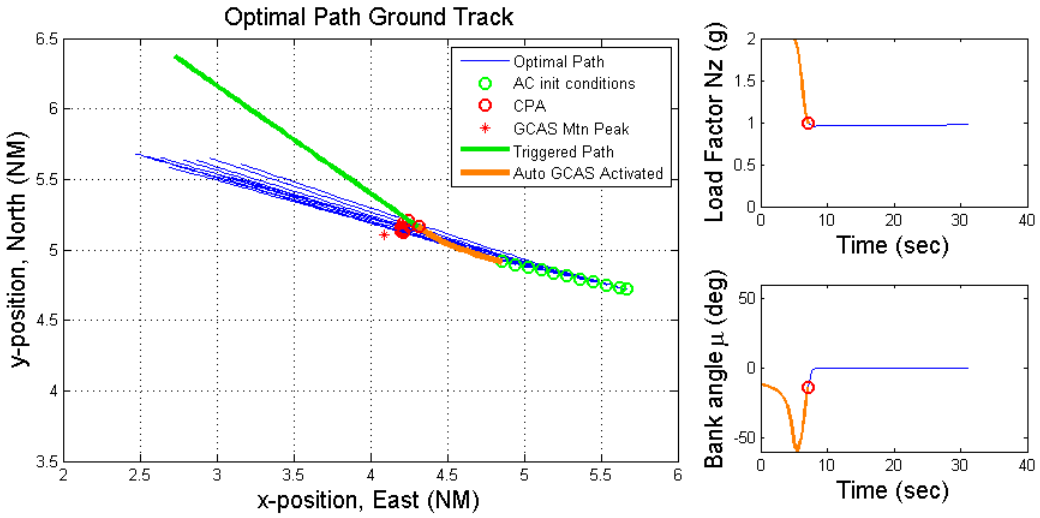


Figure 33. Optimal Path and Control (20th Run), Med-Speed Heavy, Case 1 Low, $\gamma_0 = 0°$ Level

Case 2 'Low' Nose High, Level

The fourth scenario was designed to show for the low altitude scenario that a slight nose up flight path angle would be beneficial in that it would result in a later trigger time compared to the Case 2 'Low' Level scenario. The data for each run is listed in Table 18, while the top and side view of the optimal path are shown in Figures 34 and 35. The aggressive trigger criteria was approximately satisfied at Run 22 with 2 g and a max bank angle of -36.19°. This equates to a trigger time of 1 second later than the level case. The time at the max limit of 2 g was 3.26 sec, the activation duration t_{CPA} was 6.65 sec, which yields an Agg ratio = 0.49. The NLP solver time at trigger was 0.838 sec, while the average NLP solver time was 0.618. The minimum vertical clearance for the left, center (optimal), and right paths was 246 ft, 350 ft, and 456 ft AGL.

Table 18. Data Table for Medium-Speed Heavy, Case 2 Low, $\gamma_0 = 5°$

Run #	t_0 (sec)	t_{CPA} (sec)	Agg ratio	Bank max (deg)	Bank min (deg)	N_z max (g)	N_z min (g)	NLP time (sec)	Clr Lft (ft)	Clr Ctr (ft)	Clr Rgt (ft)
1	0	21.04	0.00	0.20	-0.00	1.02	1.00	2.148	249	350	407
2	0.5	17.45	0.00	0.02	-0.04	1.03	1.00	0.608	246	350	414
3	1.0	16.91	0.00	0.00	-0.44	1.04	1.00	0.538	254	350	427
4	1.5	16.39	0.00	0.00	-0.54	1.05	1.00	0.413	259	350	436
5	2.0	15.95	0.00	0.00	-0.76	1.07	1.00	0.413	263	350	432
6	2.5	15.44	0.00	0.00	-0.93	1.08	1.00	0.411	265	350	433
7	3.0	14.84	0.00	0.00	-0.85	1.10	1.00	0.411	261	350	433
8	3.5	14.35	0.00	0.00	-1.05	1.12	1.00	0.416	262	350	433
9	4.0	13.81	0.00	0.00	-1.11	1.14	1.00	0.411	266	350	431
10	4.5	13.29	0.00	0.00	-1.22	1.16	1.00	0.415	267	350	423
11	5.0	12.85	0.00	0.00	-1.68	1.19	1.00	0.412	274	350	416
12	5.5	12.34	0.00	0.00	-1.93	1.22	1.00	0.569	278	350	415
13	6.0	11.74	0.00	0.00	-1.52	1.25	1.00	0.363	275	350	416
14	6.5	11.25	0.00	0.00	-1.81	1.30	1.00	0.362	280	350	416
15	7.0	10.71	0.00	0.00	-1.70	1.34	1.00	0.491	287	350	409
16	7.5	10.19	0.00	0.00	-1.67	1.40	1.00	0.710	291	350	404
17	8.0	9.75	0.00	0.00	-2.60	1.47	0.99	0.454	301	350	396
18	8.5	9.24	0.00	0.00	-2.92	1.56	0.98	1.296	308	350	394
19	9.0	8.64	0.00	0.00	-1.38	1.67	0.96	1.071	311	350	389
20	9.5	8.15	0.00	0.00	-2.73	1.91	0.95	1.049	310	350	395
21	10.0	7.61	0.19	0.05	-15.03	2.00	0.96	1.077	310	350	448
22	10.5	6.65	0.49	0.14	-42.33	2.00	0.95	0.838	246	350	456

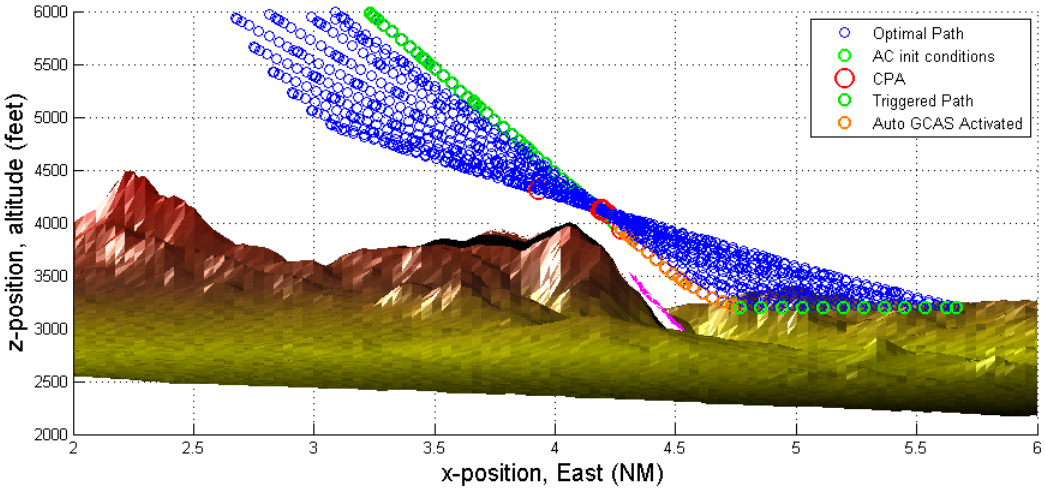


Figure 34. Optimal Path (22nd Run), Med-Speed Heavy, Case 1 Low, $\gamma_0 = 5°$ Level

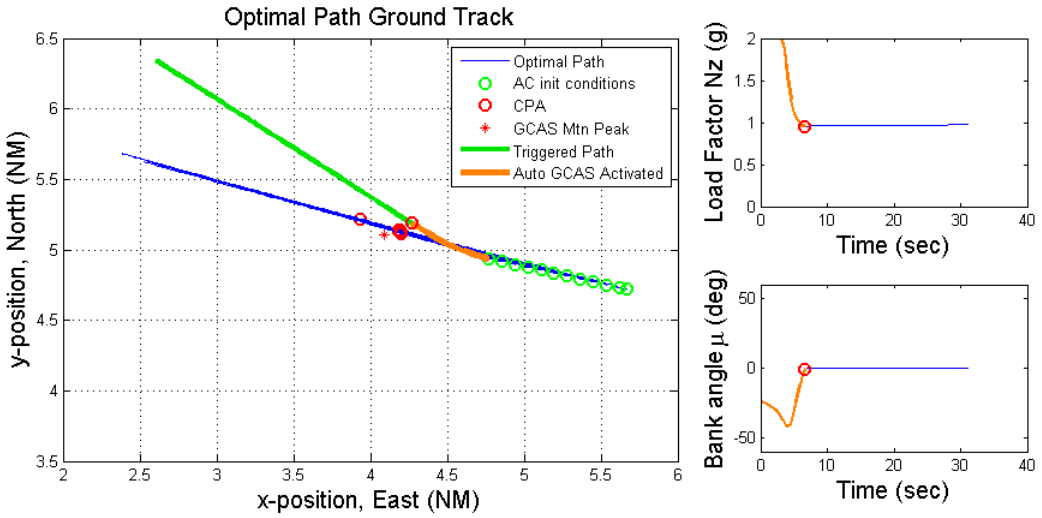


Figure 35. Optimal Path and Control (22nd Run), Med-Speed Heavy, Case 1 Low, $\gamma_0 = 5°$ Level

Table 19 is a data comparison for the Case 2 Low altitude level and nose-high scenarios at the trigger point. Here, the aircraft position (x, y, z) and heading χ initial conditions are identical for each of the runs in both the level and nose-high scenarios, however, for the level scenario the initial flight path angle $\gamma_0 = 0°$ and for the nose-high scenario $\gamma_0 = 5°$. Although it would be difficult for an aircraft to fly with a 5° and level altitude, the point here is to simply show that if at any point the pilot input a slight nose-up attitude that the trigger of the Auto GCAS recovery will be later than if the aircraft remained at $\gamma_0 = 0°$. Table 19 shows that the auto recovery is triggered one second later for the nose-high scenario (10.5 vs. 9.5 sec), while the t_{CPA} is shorter (6.65 vs. 7.09 sec) indicating a shorter Auto GCAS activation time. This is due to the advantage of the nose-high initial flight path angle of $\gamma_0 = 5°$ which makes the nose-high scenario less demanding for the optimizer. The aggressiveness ratio for the nose-high scenario is lower with a value of 0.49 vice 0.71 and the average NLP time is significantly lower at 0.618 sec vice the 1.137 sec average NLP time for the more challenging low altitude level scenario.

Table 19. Data Comparison of Case 2 Low Alt Level and Nose-High Scenarios at Trigger

Scenario	Trigger Point	Auto Activation Duration (t_{CPA})	Aggressiveness Ratio	NLP time average
Low Alt Level	Run 20 (9.5 sec)	7.09 sec	0.71	1.137 sec
Low Alt Nose-High	Run 22 (10.5 sec)	6.65 sec	0.49	0.618 sec

5.8.1.2 Satisfaction of the Aggressive Criteria.

As seen in the four previous scenarios, the aggressiveness ratios approached but did not equal 1, hence the aggressive trigger criteria are only approximated. This is due both to the discrete 0.5 sec update rate (2Hz) and the number of collocation points used (91 points). Therefore, the update rate is changed to 0.01 sec (100 Hz) and the number of collocation points is increased to 181 in order to demonstrate that the aggressiveness ratio approaches 1 in the limit. Figure 36 shows t_0 vs. the aggressiveness ratio for the Case 2 'Low' Level scenario with the faster update rate and increased number of collocation points.

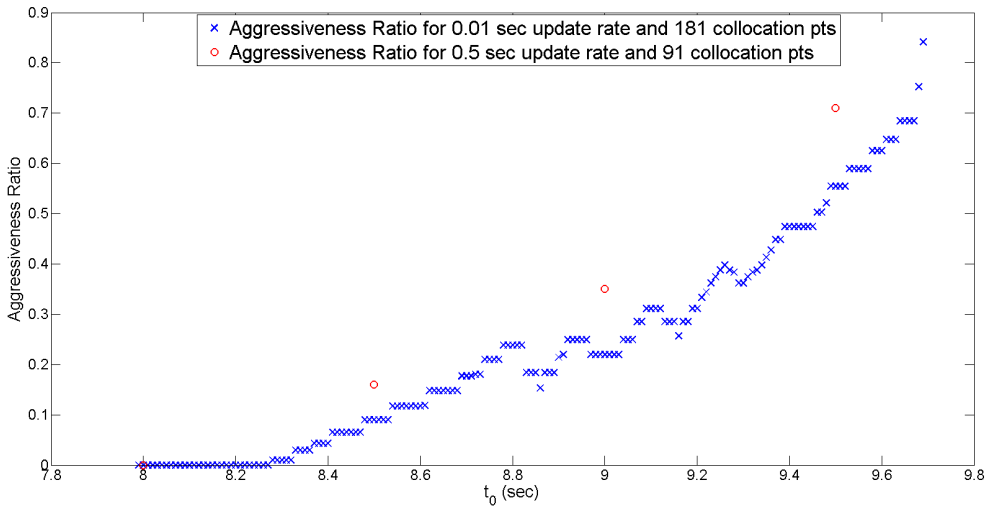


Figure 36. t_0 vs. Aggressiveness Ratio, Med-Speed Heavy, Case 2 'Low' Alt Level

The blue Xs are the aggressiveness ratio for the 0.01 sec update rate and 181 collocation points (max Agg ratio is 0.84) and the red circles are the aggressiveness ratio for the 0.5 sec update rate and 91 collocation points (max Agg ratio is 0.71). As demonstrated in Figure 36, the aggressiveness ratio does approach 1 in the limit as the update rate becomes faster and the number of collocation points increases. Figure 36 also shows that the results presented previously with 91 collocation points and the 0.5 update rate provided conservative estimates of aggressiveness criteria, meaning it would trigger earlier. This is evidenced by $t_{trigger} = 9.5$ sec and Agg ratio = 0.71 for the 91 collocation points and 0.5 sec update rate runs and $t_{trigger} = 9.69$ sec and Agg ratio = 0.84 for the 181 collocation points and 0.01 sec update rate runs.

5.8.2 Effect of Weighting Terms in the Cost Functional.

A limited evaluation of changing the control weighting matrix $\mathbf{R}$ in the cost functional given in Equation (83) is presented here, where $\mathbf{R}$ is defined in Equation (84). The purpose is to determine the most appropriate weightings for Auto GCAS as well as investigate how responsive the cost functional is to changing the control weightings in order to tailor the optimal solution to climbing or turning. The weighting on the bank angle μ term in the cost functional, r_1, is set to 1 for all of the scenarios in this research. A limited comparison is done here for the weighting on the load factor N_z term of $r_2 = 0.1$, $r_2 = 1$, $r_2 = 5$, $r_2 = 10$, and $r_2 = 50$.

The scenario run for this comparison is the Case 2 'Low' Level for the medium-speed heavy. An update rate of 0.5 sec is used, however 41 collocation points are used in the optimizer for this evaluation to demonstrate the difference in NLP solver time with less points. All of the data for each of the five control weightings are shown first in Tables 20 to 24 and Figures 37 to 46. Then, the results for all of the control weightings on N_Z are summarized in Table 25 followed by a discussion of the results.

5.8.2.1 Penalty on Load Factor, $r_2 = 0.1$.

Table 20 shows the data for each time the optimizer was run at a new initial condition which include: the run number, t_0, t_{CPA}, the aggressiveness ratio, the max and min bank angle μ required by the optimizer (recall that positive bank is a left turn and negative bank is a right turn), the max and min load factor N_z required by the optimizer, the NLP solver time, and the minimum vertical clearance for the left, center, and right constraint paths. Recall that $d_{threshold} = 0$ ft for the left and right parallel paths, while $d_{threshold} = 350$ ft for the center (optimal) path.

Table 20. Data Table for Med-Speed Heavy, Case 2 Low, $\gamma_0 = 0°$ Level, $r_2 = 0.1$

Run #	t_0 (sec)	t_{CPA} (sec)	Agg ratio	Bank max (deg)	Bank min (deg)	N_z max (g)	N_z min (g)	NLP time (sec)	Clr Lft (ft)	Clr Ctr (ft)	Clr Rgt (ft)
1	0	18.33	0.00	0.00	-0.72	1.25	1.00	1.357	266	350	425
2	0.5	17.33	0.00	0.00	-0.28	1.28	1.00	0.312	283	350	443
3	1.0	17.33	0.00	0.00	-0.83	1.28	1.00	0.198	270	350	428
4	1.5	16.16	0.00	0.00	-0.16	1.32	1.00	0.227	279	350	422
5	2.0	16.16	0.00	0.00	-0.49	1.33	1.00	0.249	285	350	416
6	2.5	15.23	0.00	0.00	-0.26	1.36	1.00	0.227	287	350	422
7	3.0	15.23	0.00	0.00	-0.91	1.37	1.00	0.225	280	350	418
8	3.5	14.23	0.00	0.00	-0.31	1.41	1.00	0.204	280	350	418
9	4.0	14.23	0.00	0.00	-1.03	1.43	1.00	0.212	287	350	426
10	4.5	13.06	0.00	0.00	-0.06	1.48	1.00	0.199	280	350	396
11	5.0	13.06	0.00	0.00	-0.92	1.51	1.00	0.565	302	350	400
12	5.5	12.40	0.00	0.00	-0.66	1.56	1.00	0.302	304	350	401
13	6.0	12.13	0.00	0.00	-1.17	1.60	0.99	0.276	295	350	407
14	6.5	11.13	0.00	0.00	-0.25	1.68	0.98	0.222	295	350	393
15	7.0	11.13	0.00	0.00	-1.43	1.73	0.97	0.228	307	350	416
16	7.5	9.96	0.00	0.24	-0.01	1.83	0.96	0.229	304	350	370
17	8.0	9.96	0.00	0.00	-1.75	1.96	0.96	0.413	320	350	387
18	8.5	9.30	0.07	0.00	-1.90	2.00	0.96	0.369	325	350	388
19	9.0	9.03	0.31	0.00	-7.55	2.00	0.95	0.439	294	350	398

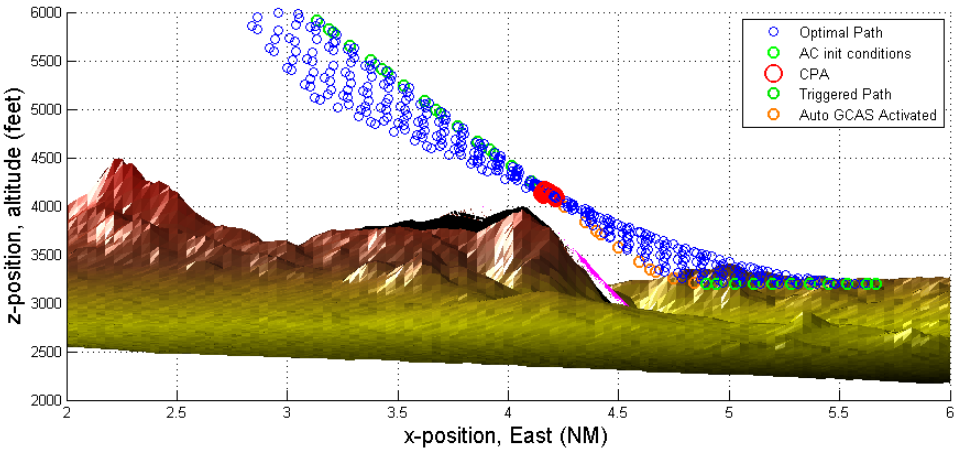


Figure 37. Optimal Path (19th Run), Med-Speed, Case 2 Low, $\gamma_0 = 0°$ Level, $r_2 = 0.1$

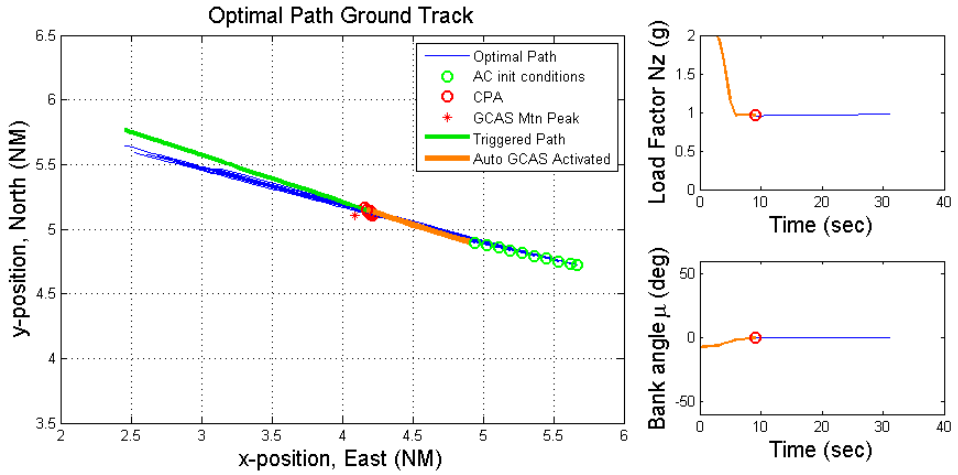


Figure 38. Optimal Path and Control (19th Run), Med-Speed, Case 2 Low, $\gamma_0 = 0°$ Level, $r_2 = 0.1$

5.8.2.2 Penalty on Load Factor, $r_2 = 1$.

Table 21. Data Table for Med-Speed Heavy, Case 2 Low, $\gamma_0 = 0°$ Level, $r_2 = 1$

Run #	t_0 (sec)	t_{CPA} (sec)	Agg ratio	Bank max (deg)	Bank min (deg)	N_z max (g)	N_z min (g)	NLP time (sec)	Clr Lft (ft)	Clr Ctr (ft)	Clr Rgt (ft)
1	0	17.33	0.00	0.00	-1.67	1.25	1.00	0.950	265	350	450
2	0.5	17.33	0.00	0.01	-6.62	1.25	1.00	0.904	229	350	514
3	1.0	16.16	0.00	0.00	-2.10	1.28	1.00	0.292	241	350	426
4	1.5	16.16	0.00	0.02	-7.15	1.29	1.00	0.319	248	350	523
5	2.0	15.50	0.00	0.00	-0.50	1.34	1.00	0.616	284	350	417
6	2.5	15.23	0.00	-0.00	-7.73	1.34	1.00	0.241	254	350	514
7	3.0	14.23	0.00	-0.00	-2.30	1.37	1.00	0.523	270	350	420
8	3.5	14.23	0.00	0.02	-7.99	1.39	1.00	0.490	267	350	508
9	4.0	14.23	0.00	0.01	-4.23	1.42	1.00	0.494	248	350	416
10	4.5	13.06	0.00	0.00	-0.52	1.48	1.00	0.598	286	350	402
11	5.0	12.40	0.00	0.00	-1.60	1.52	1.00	0.620	306	350	404
12	5.5	12.13	0.00	-0.00	-1.54	1.57	1.00	0.615	308	350	413
13	6.0	11.13	0.00	-0.00	-4.24	1.59	1.00	0.570	282	350	406
14	6.5	11.13	0.00	0.01	-1.75	1.68	0.98	0.909	310	350	408
15	7.0	11.13	0.00	0.02	-6.95	1.70	0.99	0.572	263	350	398
16	7.5	9.96	0.00	1.45	-0.01	1.83	0.96	0.667	308	350	359
17	8.0	9.30	0.00	0.02	-3.05	1.94	0.96	0.715	325	350	390
18	8.5	9.03	0.07	-0.00	-2.27	2.00	0.96	0.750	323	350	393
19	9.0	8.03	0.23	-0.00	-7.24	2.00	0.96	0.651	289	350	397
20	9.5	6.86	0.55	0.62	-39.08	2.00	0.95	0.492	230	350	434

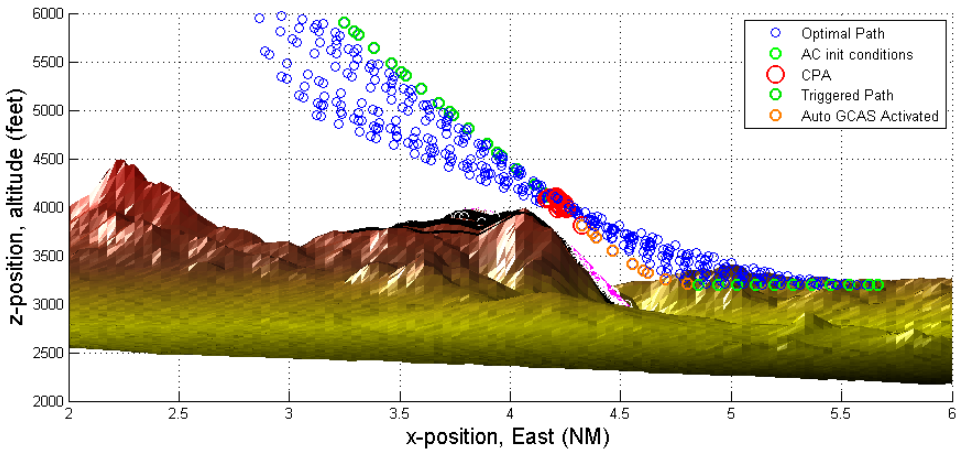


Figure 39. Optimal Path (20th Run), Med-Speed, Case 2 Low, $\gamma_0 = 0°$ Level, $r_2 = 1$

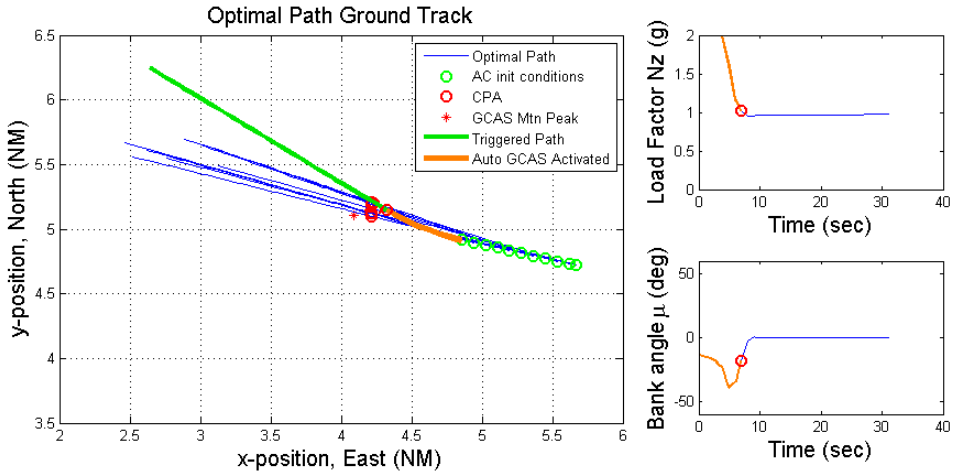


Figure 40. Optimal Path and Control (20th Run), Med-Speed, Case 2 Low, $\gamma_0 = 0^\circ$ Level, $r_2 = 1$

5.8.2.3 Penalty on Load Factor, $r_2 = 5$.

Table 22. Data Table for Med-Speed Heavy, Case 2 Low, $\gamma_0 = 0°$ Level, $r_2 = 5$

Run #	t_0 (sec)	t_{CPA} (sec)	Agg ratio	Bank max (deg)	Bank min (deg)	N_z max (g)	N_z min (g)	NLP time (sec)	Clr Lft (ft)	Clr Ctr (ft)	Clr Rgt (ft)
1	0	17.33	0.00	0.05	-12.85	1.18	1.00	0.862	181	350	455
2	0.5	17.33	0.00	0.05	-11.52	1.21	1.00	0.343	198	350	483
3	1.0	16.16	0.00	0.06	-15.38	1.20	1.00	0.249	165	350	467
4	1.5	16.16	0.00	0.05	-12.37	1.24	1.00	0.268	179	350	484
5	2.0	15.50	0.00	0.00	-13.12	1.26	1.00	0.246	166	350	489
6	2.5	15.23	0.00	0.00	-14.25	1.28	1.00	0.290	188	350	484
7	3.0	14.23	0.00	0.11	-17.64	1.29	1.00	0.250	168	350	475
8	3.5	14.23	0.00	0.04	-16.23	1.33	1.00	0.248	190	350	485
9	4.0	13.06	0.00	0.14	-21.33	1.35	1.00	0.246	155	350	443
10	4.5	12.40	0.00	0.05	-20.00	1.40	1.00	0.246	168	350	488
11	5.0	12.13	0.00	0.00	-21.41	1.43	1.00	0.271	162	350	468
12	5.5	11.13	0.00	0.13	-23.24	1.49	1.00	0.308	171	350	467
13	6.0	11.13	0.00	0.26	-22.21	1.55	1.00	0.256	165	350	484
14	6.5	9.96	0.00	0.36	-26.26	1.63	1.00	0.286	183	350	483
15	7.0	9.96	0.00	0.28	-19.47	1.73	1.00	0.307	241	350	523
16	7.5	9.30	0.00	0.21	-15.53	1.80	1.00	0.339	235	350	489
17	8.0	9.03	0.00	0.00	-15.90	1.92	0.98	0.365	254	350	491
18	8.5	9.03	0.00	0.00	-17.55	2.00	0.98	0.365	232	350	465
19	9.0	8.03	0.23	0.51	-20.60	2.00	0.95	0.684	286	350	415
20	9.5	8.03	0.47	1.95	-41.25	2.00	0.95	0.418	229	350	430

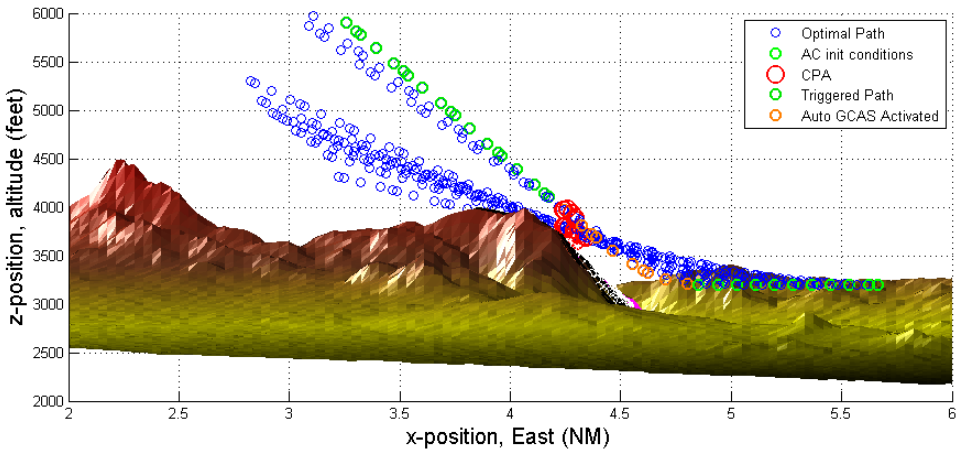


Figure 41. Optimal Path (20th Run), Med-Speed, Case 2 Low, $\gamma_0 = 0°$ Level, $r_2 = 5$

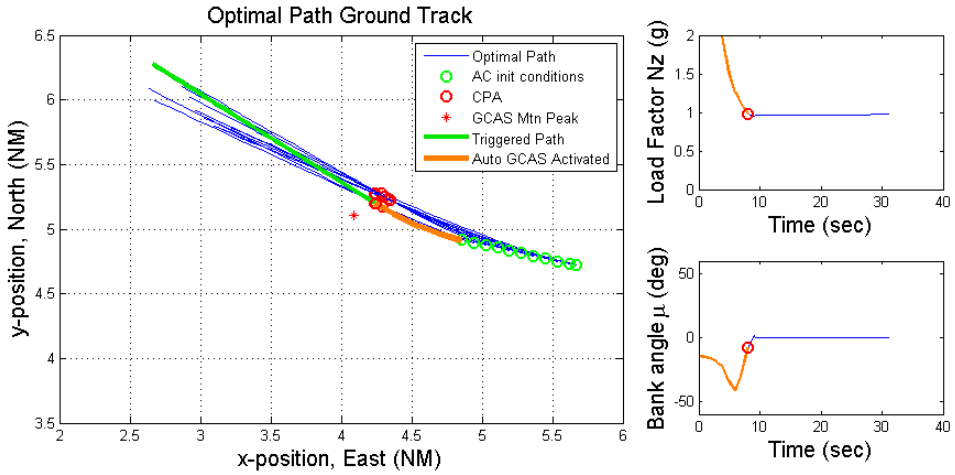


Figure 42. Optimal Path and Control (20th Run), Med-Speed, Case 2 Low, $\gamma_0 = 0^\circ$ Level, $r_2 = 5$

5.8.2.4 Penalty on Load Factor, $r_2 = 10$.

Table 23. Data Table for Med-Speed Heavy, Case 2 Low, $\gamma_0 = 0°$ Level, $r_2 = 10$

Run #	t_0 (sec)	t_{CPA} (sec)	Agg ratio	Bank max (deg)	Bank min (deg)	N_z max (g)	N_z min (g)	NLP time (sec)	Clr Lft (ft)	Clr Ctr (ft)	Clr Rgt (ft)
1	0	17.33	0.00	0.00	-13.32	1.17	1.00	1.276	191	350	454
2	0.5	17.33	0.00	0.09	-14.74	1.19	1.00	0.570	206	350	454
3	1.0	16.16	0.00	0.10	-15.55	1.19	1.00	0.296	169	350	441
4	1.5	16.16	0.00	0.11	-15.96	1.23	1.00	0.963	199	350	459
5	2.0	15.50	0.00	0.00	-16.34	1.24	1.00	0.337	205	350	463
6	2.5	15.23	0.00	0.00	-16.96	1.27	1.00	0.677	186	350	459
7	3.0	14.23	0.00	0.19	-18.81	1.28	1.00	0.608	176	350	434
8	3.5	14.23	0.00	0.17	-15.71	1.32	1.00	0.958	196	350	485
9	4.0	13.06	0.00	0.12	-21.37	1.34	1.00	0.334	149	350	424
10	4.5	12.40	0.00	0.17	-18.77	1.39	1.00	0.993	166	350	488
11	5.0	12.13	0.00	0.00	-20.09	1.43	1.00	0.366	162	350	467
12	5.5	12.13	0.00	0.00	-22.13	1.48	1.00	0.576	166	350	464
13	6.0	9.96	0.00	0.12	-22.58	1.55	1.00	0.300	162	350	468
14	6.5	9.96	0.00	0.61	-26.13	1.63	1.00	0.366	165	350	469
15	7.0	9.30	0.00	0.00	-26.78	1.73	1.00	0.333	174	350	462
16	7.5	9.03	0.00	0.00	-28.37	1.82	1.00	0.299	170	350	484
17	8.0	9.03	0.00	0.00	-21.95	1.92	1.00	0.509	208	350	503
18	8.5	9.03	0.00	0.00	-17.34	2.00	0.98	0.441	231	350	465
19	9.0	8.03	0.23	1.50	-26.06	2.00	0.95	0.479	287	350	420
20	9.5	8.03	0.47	4.92	-41.71	2.00	0.95	0.653	229	350	430

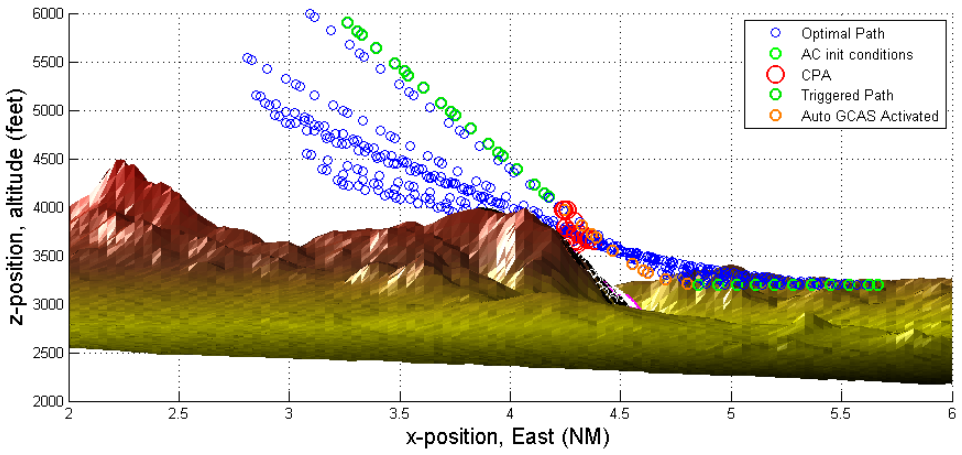


Figure 43. Optimal Path (20th Run), Med-Speed, Case 2 Low, $\gamma_0 = 0°$ Level, $r_2 = 10$

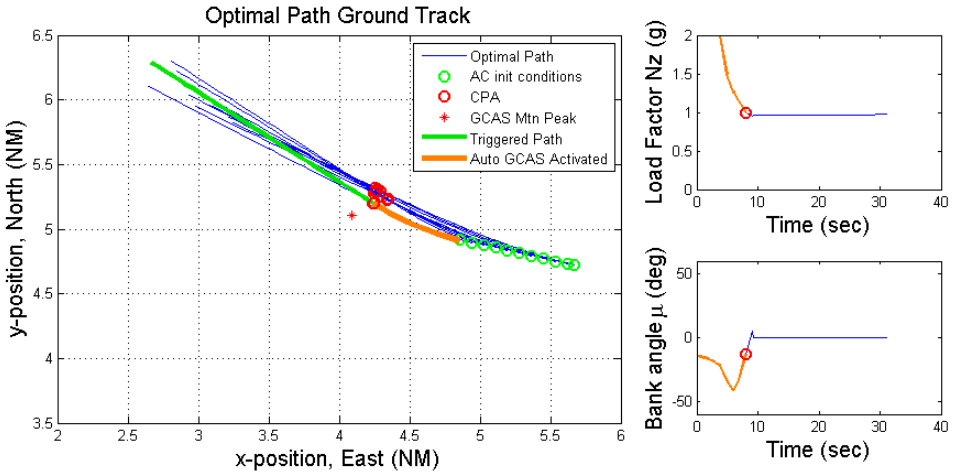


Figure 44. Optimal Path and Control (20th Run), Med-Speed, Case 2 Low, $\gamma_0 = 0^\circ$ Level, $r_2 = 10$

5.8.2.5 Penalty on Load Factor, $r_2 = 50$.

Table 24. Data Table for Med-Speed Heavy, Case 2 Low, $\gamma_0 = 0°$ Level, $r_2 = 50$

Run #	t_0 (sec)	t_{CPA} (sec)	Agg ratio	Bank max (deg)	Bank min (deg)	N_z max (g)	N_z min (g)	NLP time (sec)	Clr Lft (ft)	Clr Ctr (ft)	Clr Rgt (ft)
1	0	17.33	0.00	0.00	-13.46	1.16	1.00	0.840	205	350	461
2	0.5	17.33	0.00	0.34	-14.68	1.18	1.00	0.848	226	350	437
3	1.0	16.16	0.00	0.37	-14.63	1.19	1.00	0.274	178	350	436
4	1.5	16.16	0.00	0.44	-16.07	1.22	1.00	0.330	215	350	436
5	2.0	15.50	0.00	0.00	-16.24	1.24	1.00	0.198	206	350	443
6	2.5	15.23	0.00	0.00	-17.56	1.26	1.00	0.225	201	350	428
7	3.0	14.23	0.00	0.82	-18.04	1.27	1.00	0.252	179	350	420
8	3.5	14.23	0.00	0.91	-19.12	1.31	1.00	0.200	184	350	429
9	4.0	12.13	0.00	0.38	-21.68	1.34	1.00	0.355	154	350	396
10	4.5	13.06	0.00	1.24	-20.66	1.38	1.00	0.225	164	350	449
11	5.0	12.40	0.00	0.00	-22.06	1.41	1.00	0.328	166	350	419
12	5.5	12.13	0.00	0.00	-20.89	1.47	1.00	0.223	160	350	463
13	6.0	9.96	0.00	2.63	-22.13	1.56	1.00	0.760	163	350	465
14	6.5	9.96	0.00	2.40	-25.10	1.63	1.00	0.405	162	350	476
15	7.0	9.03	0.00	0.00	-25.99	1.73	1.00	0.378	173	350	463
16	7.5	9.03	0.00	0.00	-28.28	1.82	1.00	0.276	167	350	483
17	8.0	9.03	0.00	0.00	-22.21	1.92	1.00	0.279	205	350	504
18	8.5	9.03	0.00	0.00	-23.32	2.00	0.99	0.388	228	350	464
19	9.0	8.03	0.23	0.00	-59.73	2.00	0.98	0.388	285	350	423

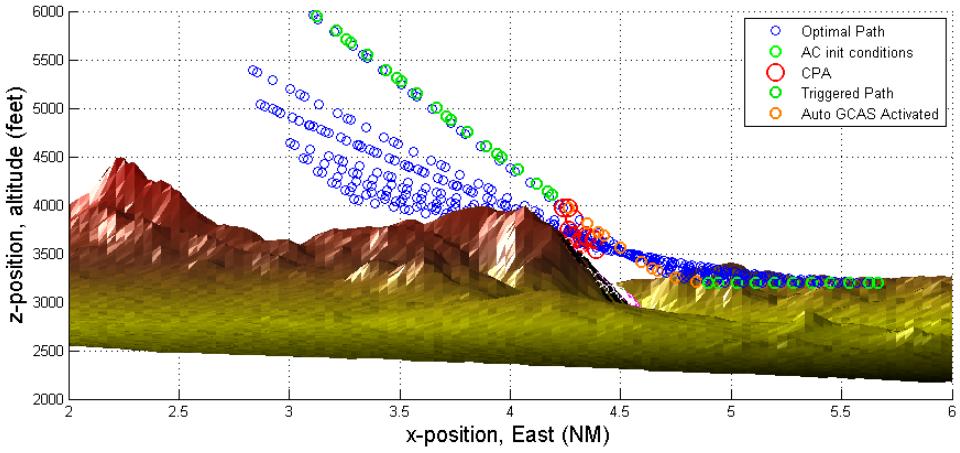


Figure 45. Optimal Path (19th Run), Med-Speed, Case 2 Low, $\gamma_0 = 0°$ Level, $r_2 = 50$

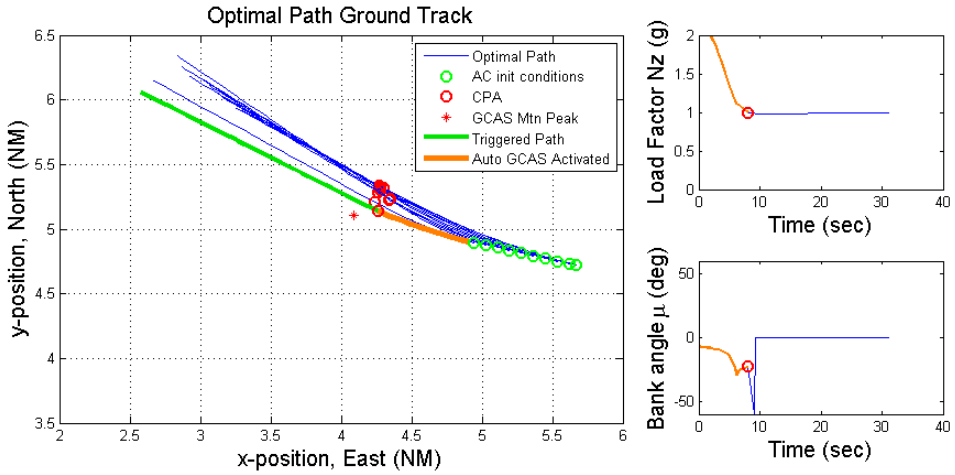


Figure 46. Optimal Path and Control (19th Run), Med-Speed, Case 2 Low, $\gamma_0 = 0°$ Level, $r_2 = 50$

Table 25 lists the data at trigger for each of the control weighting scenarios. The column headings are r_2, run number at trigger, $t_{trigger}$, t_{CPA}, the aggressiveness ratio, the max positive bank angle required (left bank), the max negative bank angle required (right bank), the maximum load factor required, the minimum load factor required, and the average NLP solver time for IPOPT. Average NLP solver time was calculated by throwing out the longest time and the shortest time for a scenario and averaging the remaining NLP solver times.

Table 25. Data Summary of Changing Penalty on the N_z Term in the Cost Functional

$\mathbf{r_2}$ ($\mathbf{N_z}$ Penalty)	Trigger (Run #)	$\mathbf{t_{trigger}}$ (sec)	$\mathbf{t_{CPA}}$ (sec)	Agg ratio	Bank max (deg)	Bank min (deg)	$\mathbf{N_z}$ max (g)	$\mathbf{N_z}$ min (g)	NLP avg time (sec)
0.1	19	9.0	9.03	0.31	0	-7.55	2.0	0.95	0.288
1	20	9.5	6.86	0.55	0.62	-39.08	2.0	0.95	0.576
5	20	9.5	8.03	0.47	1.95	-41.25	2.0	0.95	0.318
10	20	9.5	8.03	0.47	4.92	-41.71	2.0	0.95	0.390
50	19	9.0	8.03	0.23	0	-59.73	2.0	0.98	0.318

For each of the above simulations presented in Table 25 at the run past the trigger point, the NLP Solver IPOPT reported either 'Max Number of Iterations Reached' (max iterations is set to 4,000) or 'Infeasible'. Hence, the optimal control solution

triggered was the closest to the theoretical optimal solution given the update rate of 2 Hz and the number of collocation points used.

For $r_2 = 0.1$ where bank angle μ is penalized more than N_z, the optimal auto recovery is almost a pure pull maneuver (see Figures 37 and 38). At trigger at Run 19 (see Figure 20), the auto recovery is at 2 g and requires only -7.55° (slight right bank). From the data in Table 25, we see that as the value of r_2 increases from 0.1 to 50 that the maximum left bank required increases from 0° to 4.92° while the maximum right bank required increases from -7.55° to -59.73°. (However, -59.73° was required after t_{CPA}; a max right bank value of -29.16° was required prior to t_{CPA}. This shows that the more N_z is penalized, the more bank is required for the optimal control solution. Hence, the cost functional is able to be easily tailored to climbing or turning based on mission needs.

F-16 flight tests have demonstrated that the latest trigger time (represented here by Trigger Run # and $t_{trigger}$) and the shortest activation time (represented here by t_{CPA}) are the most desirable for the pilot to not consider the activation a nuisance [7]. Thus, latest trigger time and shortest auto activation time are used as measures of performance for this weighting comparison. From the results in Table 14 it can be seen that the aggressive trigger criteria are approximately satisfied for $r_2 = 0.1$ and $r_2 = 50$ at Run 19, while the aggressive trigger criteria are approximately satisfied for $r_2 = 1$, $r_2 = 5$, and $r_2 = 10$ at Run 20. Since triggering later is desirable in an Auto GCAS to reduce nuisances to the pilot, $r_2 = 1$, $r_2 = 5$, and $r_2 = 10$ show the best results for the latest trigger time at Run 20. Next, recall that t_{CPA} represents how many seconds that the Auto GCAS recovery will be activated (commanded by the autopilot) and that control is returned back to the pilot immediately after this point, once clearance of terrain is assured. In comparing t_{CPA}, note that $r_2 = 1$ has the smallest total activation time of 6.86 sec. The only disadvantage of the $r_2 = 1$

weighting is that even though t_{CPA} is the smallest, the average NLP time is the greatest at 0.576 sec. Finally, in comparing the average NLP solver time in the last column, we see that $r_2 = 5$ has the fastest time of 0.318 sec. The average

The conclusion from this limited study is that $r_2 = 1$ is the most appropriate weighting due to both its later trigger time (Run 20) and shortest automatic recovery activation time ($t_{CPA} = 6.86$ sec). Although the larger average NLP is not desirable, it was not the primary purpose of this research to pick the fastest computational time for real-time implementation, but rather the best method that meets the aggressive and timely operational requirements of Auto GCAS.

In summary, it was demonstrated that changing the weighting of the control terms in the cost functional does influence the optimal solution to be more of a vertical or more of a lateral maneuver. Hence, the cost functional can be tailored to what is desired based on the mission of the aircraft. It is recommended that simulation with a higher-fidelity model be done in order to choose the weighting of the control terms that is most appropriate for the operational mission of the candidate aircraft.

5.8.3 Comparison of Optimal Control to Multi-Trajectory Auto GCAS.

As stated in Chapter I, the Multi-Trajectory Auto GCAS is follow-on research that is currently being conducted by Trombetta at AFIT [91]. The Multi-Trajectory Auto GCAS is designed for military heavy type aircraft and chooses the best of five pre-planned automatic recovery maneuvers. A limited flight test of the Multi-Trajectory Auto GCAS will be conducted in September 2015 at the US Air Force Test Pilot School (TPS) on the Calspan Learjet with an onboard programmable Variable Stability System (VSS) [96]. Use of the Calspan Learjet's VSS will enable in-flight simulation of a military heavy aircraft and will allow comparison between the Multi-Trajectory Auto GCAS and the Optimal Auto GCAS in this research.

Before the comparison is made between the two algorithms, a description of the Multi-Trajectory Auto GCAS is necessary. Trombetta's Multi-Trajectory Auto GCAS includes five pre-planned avoidance paths: a Forward path (wings-level 2-g pull until γ_{max} is reached), Left-Up and Right-Up paths (first a 30° bank is commanded and once established then a 2-g pull is commanded until γ_{max} is reached), and Left and Right paths (a 60° bank and 2-g level turn is commanded) [91]. For this comparison, $\gamma_{max} = 15.8°$ and $\gamma_{min} = -15.8°$ for both the Multi-Trajectory and the Optimal Auto GCAS. Figure 47 illustrates the five pre-planned avoidance paths used in the Multi-Trajectory Auto GCAS.

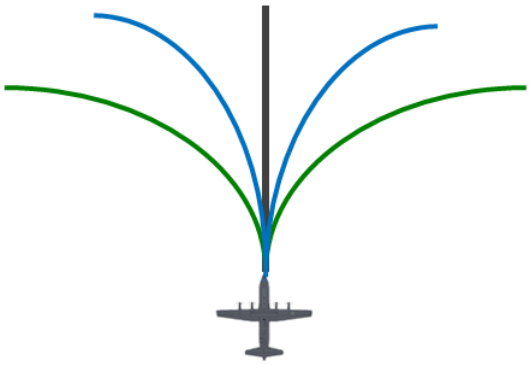


Figure 47. Illustration of the Five Pre-Planned Paths in Multi-Trajectory Auto GCAS

The Multi-Trajectory Auto GCAS implements a 'last man standing' approach similar to Small UAV Auto GCAS, where any predicted avoidance paths that intersect digital terrain are eliminated but no auto recovery is commanded as long as there is at least one viable avoidance path [83, 91]. Once the last of the five avoidance paths intersects with the digital terrain, then the algorithm commands that path as the auto recovery.

Metrics that allow comparison between optimal recovery trajectories and the Multi-Trajectory Auto GCAS recovery trajectories were developed for Auto GCAS design and analysis purposes and are presented in the next section.

5.8.4 Metrics.

Two metrics were developed to compare each method, an Aggressiveness Metric and a Timeliness Metric. The Aggressiveness Metric compares the control of the Multi-Trajectory Auto GCAS solution to the Optimal Auto GCAS solution at the same trigger time and initial conditions. The comparison is made at the trigger time for the Multi-Trajectory Auto GCAS. The Aggressiveness Metric is a measure of how much less control would be required had a Min Control optimal solution been found at the same trigger point and initial conditions. The goal is to show how well (or not so well) the Multi-Trajectory solutions span the space of all available aggressive and timely trajectories. Because control is the measure of aggressiveness, the Min Control cost functional is used as a basis of comparison between the two algorithms. Applying the Min Control cost functional to the triggered path of the Multi-Trajectory Auto GCAS yields

$$J_{Multi} = \int_{t_0}^{t_{CPA_{Multi}}} r_1 \left(\frac{\mu_{Multi}(t)}{\mu_{max}} \right)^2 + r_2 \left(\frac{N_{z_{Multi}}(t) - 1}{N_{z_{max}} - 1} \right)^2 \, \mathrm{d}t \tag{102}$$

where $\mu_{max} = 60^\circ$ and $N_{z_{max}} = 2$ g. Recall that the Min Control cost functional, $J_{Optimal}$, was defined as shown in Equation (85). An equal weighting of the two control terms is used for this comparison, hence, $r_1 = r_2 = 1$ for both Equation (85) and (102). The Aggressiveness Metric is defined as the difference between the two performance measures divided by the performance measure of the triggered path of the Multi-Trajectory Auto GCAS as shown below in Equations (103) and (104)

$$\text{Aggressiveness Metric} \equiv \frac{J_{Multi} - J_{Optimal}}{J_{Multi}} \tag{103}$$

$$= 1 - \frac{J_{Optimal}}{J_{Multi}}. \tag{104}$$

The Aggressiveness Metric approaches zero if the Multi-Trajectory solution approaches the optimal solution. A high value (closer to one) of the Aggressiveness Metric indicates that there still exists a Min Control path that an aware pilot could have executed at the time of trigger for the pre-planned auto recovery that were not included as one of the Multi-Trajectory path options.

The Timeliness Metric measures how much later the Optimal Auto GCAS auto recovery is triggered than the triggered Multi-Trajectory auto recovery as shown in Equation (105).

$$\text{Timeliness Metric} \equiv \Delta t_{trigger} = t_{trigger_{Optimal}} - t_{trigger_{Multi}} \tag{105}$$

A $\Delta t_{trigger} = 0$ represents that the triggered Multi-Trajectory avoidance path triggered at the same time as the Min Control optimal solution, and a non-zero number represents how much additional time is left before the initiation of an Optimal Auto GCAS avoidance path is required. The larger the $\Delta t_{trigger}$ the higher the likelihood that the pilot will consider the triggered Multi-Trajectory avoidance path a nuisance.

5.8.5 Comparison Scenario - Case 1 'Low' Level.

The initial conditions for this scenario are identical to those of the Case 1 'Low' Level and shown below

$$x_0 = 5.66 \text{ NM East} \tag{106}$$

$$y_0 = 4.72 \text{ NM North} \tag{107}$$

$$z_0 = 3,200 \text{ ft MSL} \tag{108}$$

$$\gamma_0 = 0^\circ \tag{109}$$

$$\chi_0 = 164^\circ \text{ (polar)}/286^\circ \text{ (compass)}. \tag{110}$$

However, for this comparison scenario γ_{max} for the Optimal Auto GCAS was set to 15.8° (vice 15.0°) in order to match the Multi-Trajectory Auto GCAS algorithm. This larger γ_{max} allows steeper trajectories which results in later trigger times than were seen in the Case 1 'Low' Level scenario. The update rate is 0.5 sec (2 Hz) as before.

At Run 31 ($t_{trigger}$ = 15.0 sec), the Multi-Trajectory Auto GCAS triggered the 'Right-Up' path which is a -30° bank (right) climbing turn. The Right-Up avoidance path is designed as follows: once a bank of $\mu = -30°$ is attained, then a 2-g pull is commanded until flight path angle $\gamma_{max} = 15.8°$ is reached. At Run 34 ($t_{trigger}$ = 16.5 sec), the Optimal Auto GCAS triggered. The the optimal path is shown together with the five pre-planned paths in Figures 48 and 49. The green path is the optimal path that was triggered, the blue path is the Multi-Trajectory path that was triggered, the red paths are the Multi-Trajectory paths that collided with terrain, while the yellow Xs indicate where along the Multi-Trajectory paths the first collision with terrain occurred. Recall that the Multi-Trajectory path uses a sphere buffer (with a 350 ft radius) around each time point in the propagated path. Hence, the yellow Xs indicate the first time point in the propagated path where the sphere buffer intersected with the terrain.

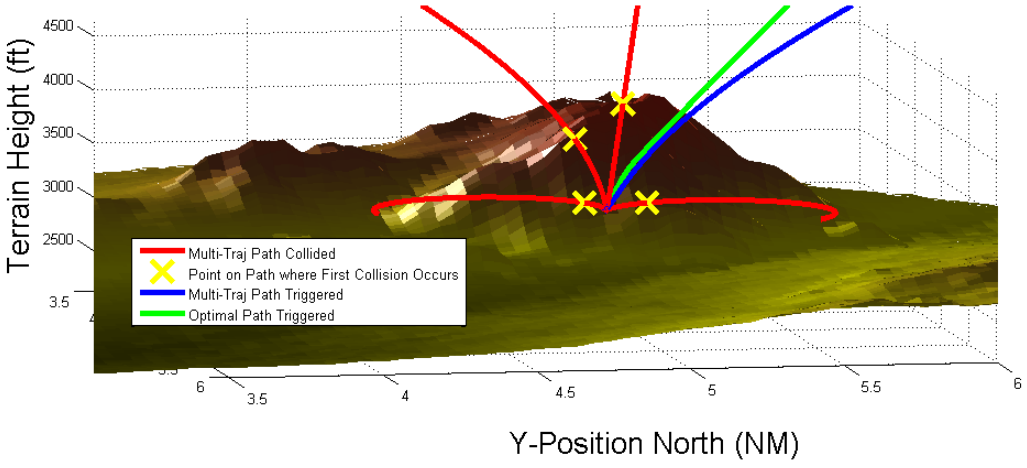


Figure 48. Side View of Multi-Trajectory Auto GCAS Paths (Run 31) vs. Optimal Auto GCAS Path (Run 34), Low-Speed Heavy, Case 1 Low, $\gamma_0 = 0°$

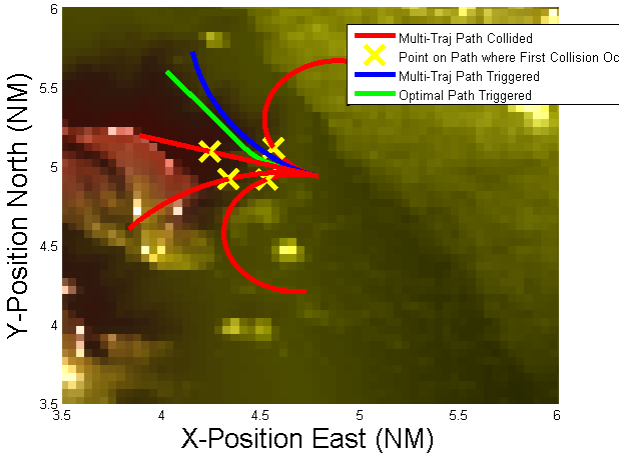


Figure 49. Top View of Multi-Trajectory Auto GCAS Paths (Run 31) vs. Optimal Auto GCAS Path (Run 34), Low-Speed Heavy, Case 1 Low, $\gamma_0 = 0°$

5.8.6 Comparison Results.

The Aggressiveness Metric calculation is shown in Equations (111) to (113). The Aggressiveness Metric of 0.72 means that the effective control budget of the Optimal Auto GCAS maneuver is only 28% of the effective control budget of the triggered Multi-Trajectory Auto GCAS maneuver. Hence, the Aggressiveness Metric indicates that there do exist other paths that an aware pilot could have executed that were not included as one of the five Multi-Trajectory pre-planned paths.

$$\text{Aggressiveness Metric} = 1 - \frac{J_{Optimal}}{J_{Multi}} \tag{111}$$

$$= 1 - \frac{1.8296}{6.5414} \tag{112}$$

$$= 0.7203 \tag{113}$$

The Multi-Trajectory Auto GCAS auto recovery was triggered at $t_{trigger} = 15.0$ sec (Run 31), while the Optimal Auto GCAS auto recovery was triggered at $t_{trigger} = 16.5$

sec (Run 34), representing a $\Delta t_{trigger} = 1.5$ sec as shown in Equation (116).

$$\Delta t_{trigger} = t_{trigger_{Optimal}} - t_{trigger_{Multi}} \tag{114}$$

$$= 16.5 \text{ sec} - 15.0 \text{ sec} \tag{115}$$

$$= 1.5 \text{ sec} \tag{116}$$

These metrics indicate that if the optimal path was included as one of the five pre-planned paths, that the Multi-Trajectory Auto GCAS should choose the optimal path and therefore trigger at a later time, reducing nuisances.

To investigate if including the optimal path as one of the options would in fact result in the optimal path being triggered, the Multi-Trajectory Auto GCAS algorithm was modified so that the Optimal Auto GCAS path replaced the Left path. The same Case 1 'Low' Level scenario was run at 210 kts for the low-speed heavy and the results were that at $t_{trigger} = 15.5$ sec (Run 32) the modified Multi-Trajectory Auto GCAS triggered the Optimal Auto GCAS path. Therefore, the Timeliness Metric in this case is

$$\Delta t_{trigger} = 16.5 \text{ sec} - 15.5 \text{ sec} \tag{117}$$

$$= 1.0 \text{ sec}, \tag{118}$$

demonstrating that the Multi-Trajectory Auto GCAS algorithm did in fact choose the Optimal Auto GCAS path as its 'Best of 5 paths' which resulted in a later trigger time which is more desirable. Figure 50 depicts the results of the modified Multi-Trajectory Auto GCAS run for the same scenario as previous. As before, the red paths represent the Multi-Trajectory paths that collided with terrain, while the green path now represents the Optimal Auto GCAS path that was included as one of the five pre-planned paths and was triggered by the Multi-Trajectory Auto GCAS.

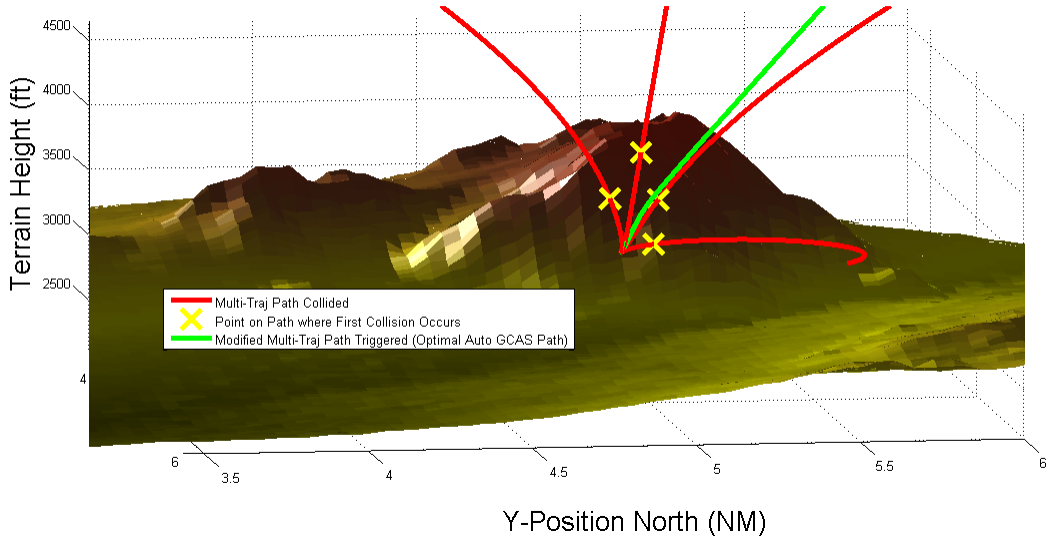


Figure 50. Multi-Trajectory Auto GCAS with Optimal Path (Run 32), Low-Speed Heavy, Case 1 Low, $\gamma_0 = 0°$

It is interesting to note that the modified Multi-Trajectory Auto GCAS did not trigger the optimal path at Run 34 ($t_{trigger_{Optimal}} = 16.5$ sec) like the Optimal Auto GCAS algorithm did. This can be explained by two reasons. First, in the Optimal Auto GCAS algorithm, there is a unique optimal path calculated for each run. Hence, the optimal path at Run 34 is not the same as the unique optimal path that was calculated for any other run. The unique combination of bank and pull for Run 34 was optimized for the specific initial conditions, dynamic and path constraints, and the aircraft's relationship to the changing terrain, whereas on a pervious run this unique bank and pull combination may hit terrain. Second, although each algorithm uses 91 time points in the propagated optimal trajectory, these time points are spaced differently. This means that the collision logic for the Multi-Trajectory Auto GCAS is evaluated at different time points than where the path constraint evaluation (terrain buffer) is performed for the Optimal Auto GCAS. The 91 time points are equispaced for the Multi-Trajectory algorithm (for use in a fixed-step ODE solver) while they are non-equispaced for the Optimal Auto GCAS algorithm. Note that 91 time points (the

initial conditions plus 90 collocation points) are purposefully non-equispaced in order to reduce error that can be seen in polynomial interpolation. This node placement is fundamental to direct orthogonal collocation (pseudospectral methods) [52]. Had the 91 time points been identical between both the modified Multi-Trajectory and the Optimal Auto GCAS algorithms, it would increase the probability of the two algorithms triggering the optimal path at the same time. Nonetheless, this research demonstrated that the inclusion of the Optimal Auto GCAS path in the modified Multi-Trajectory Auto GCAS algorithm did result in triggering the Optimal Auto GCAS path which produced a later trigger time.

5.9 Summary

This chapter further defined the optimal control method for Auto GCAS by explaining the terrain classification, aircraft performance, time horizons, the five-state and two-control aircraft model, as well as the specific Min Control cost functional and aggressive trigger criteria used for this 3-D case. Finally, this chapter presented the 3-D results which included four scenarios and a limited sensitivity analysis of control weighting for a medium-speed heavy aircraft as well as a comparison between the Optimal Auto GCAS and the Multi-Trajectory Auto GCAS for a low-speed heavy aircraft. Next, Chapter VI explores the importance of the results by presenting the conclusions and contributions of this research followed by the recommendations for future work.

VI. Conclusions

As was initially discussed, there is a desire to extend the benefits of Auto GCAS beyond fighters to military heavy aircraft. A thorough review of the literature was conducted showing that there are currently no automated ground collision avoidance systems that fulfill this need. Therefore, the objectives of this research were to:

1. Optimal Recovery Trajectory for Aircraft with Lower Climb Performance

   Develop and evaluate an optimal control technique to calculate an Auto GCAS recovery trajectory (to include lateral avoidance maneuvers) for aircraft with lower climb performance. Candidate aircraft considered are low, medium, and high-speed military heavy aircraft.

2. Comparison of Optimal Auto GCAS to a Pre-Planned Trajectory Algorithm

   Conduct a limited comparison of the Optimal Auto GCAS algorithm and an algorithm with pre-planned trajectories. For this objective, the Optimal Auto GCAS will be compared to the Multi-Trajectory Auto GCAS algorithm with five pre-planned automatic recovery maneuvers [91].

The first objective was answered by developing appropriate cost functionals and constraints, and comparing different methods to determine which is most suitable for an Auto GCAS for a heavy aircraft. As part of this, two optimal control methods were formulated and applied first to a 2-D single control and single obstacle scenario. The Min Control with an Aggressive Trigger method and the Max Distance with a Timely Trigger method were demonstrated to be equivalent at the trigger point of the automatic recovery. For the 3-D case, a method for approximating the discrete SRTM DTED posts by using a continuous and differentiable surface approximation was presented and implemented to model the terrain surface as a constraint in the optimal control formulation. Using that surface approximation method, a terrain buffer with a Left-Center-Right Look-Down approach was successfully implemented

as the path constraint to always keep the aircraft trajectory above the terrain. In short, it was shown that the best way to achieve an optimal recovery trajectory for aircraft with lower climb performance is to associate the aggressive requirement with a Min Control cost functional and the timely requirement with the threshold distance constraint.

Using the Min Control with an Aggressive Trigger method, different simulations for heavy aircraft were successfully demonstrated in simulation to show the flexibility of the Optimal Auto GCAS algorithm in different scenarios. A limited comparison of the effects of weighting the control terms in the cost functional showed that an equal weighting produced a later trigger and the shortest activation time (time the auto recovery is commanded) which are desirable characteristics of an Auto GCAS recovery (although weightings will be mission and scenario dependent). To answer the second research objective, the Optimal Auto GCAS algorithm was compared to the Multi-Trajectory Auto GCAS algorithm in a representative low-level scenario, demonstrating that the Optimal Auto GCAS algorithm can be a useful comparative tool for evaluating other Auto GCAS algorithms. An Aggressiveness Metric and a Timeliness Metric were developed to quantify the improvement of using the Optimal Auto GCAS compared to a the Multi-Trajectory algorithm. The results showed that the proposed Optimal Auto GCAS required less control and triggered later than the Multi-Trajectory Auto GCAS, which leads to a system that satisfies the aggressive and timely operational requirements. These metrics can be used to evaluate the performance and to influence the design of any Auto GCAS system for any type of aircraft.

6.1 Contributions

This research effort provided three specific contributions to the area of applying optimal control to Auto GCAS:

1. Developed a mapping of the aggressive and timely requirements into two different optimal control formulations for Auto GCAS. For the 2-D case, demonstrated that Max Distance with a Timely Trigger formulation is equivalent to Min Control with an Aggressive Trigger at the trigger point.

2. For the 3-D case, created a viable optimal control formulation (Min Control with an Aggressive Trigger) for Auto GCAS that is adaptable to multiple platforms. Although not currently ready for real-time implementation, this research demonstrated run times (NLP solver times) ranging from 0.15 to 1.1 sec which show the feasibility of real-time optimal control for Auto GCAS in the near future.

3. Developed a method of comparison (or comparative tool) between the Optimal Auto GCAS solution and any other Auto GCAS algorithm. Demonstrated the method against Trombetta's Multi-Trajectory Auto GCAS solution for a low-speed heavy [91].

6.2 Recommendations for Future Work

Although the Min Control with an Aggressive Trigger optimal control formulation was shown to be a viable method for Auto GCAS, there are several areas of research that could improve this method. Each of these areas are detailed in the paragraphs below.

In order to improve the evaluation of the terrain constraint:

R1. Examine the suitability of other interpolation methods such as Multiquadric and Multilog Radial Basis Functions (RBFs) when operating in mountainous terrain. A starting point would be Aguilar et al.'s research who showed that spline methods tend to overestimate in mountainous terrain [2].

R2. Develop a faster terrain scanning algorithm to parse the entire DTED down to a small local terrain set that is of significance. As was shown throughout the research, time was spent in the evaluation of the constraint function. The method chosen in this research was gridded data interpolation via the 'griddedInterpolant' MatLab function which performed well, but this was not based on an exhaustive search [59]. Therefore, there may be other methods that would outperform.

R3. Further development of the path constraints by improving the Left-Center-Right Look-Down constraints.

R4. Research other types of digital elevation models (to include SRTM DTED Level 2) in order to improve both accuracy and computational speed. Future DEMs may include LIDAR data. While this will improve accuracy, this will challenge the computational efficiency. The impacts of using these more accurate DEMs need to be explored for their efficacy.

R5. To ensure that this algorithm will operate as desired in a variety of terrain, we should examine running the Min Control with an Aggressive Trigger algorithm against more varied terrain (Lowland, Midland, and Upland) using varied approach angles to the terrain.

In order to further develop the method of triggering the auto recovery:

R6. Develop a two-phase formulation for the 3-D formulation for Min Control with an Aggressive Trigger. In a single phase formulation there is an abrupt transition from max to min control at the CPA. If the discretization is fine (high number of collocation points) and the optimal trajectory update rate is sufficiently fast, the transition is a jump condition at the CPA when the aggressive trigger criteria is met. If the discretization is coarse (low number of collocation points), it is possible to step over the exact jump condition and have an optimal solution that transitions to min control prior to the CPA. Also if the update rate is too slow, it is possible to step

over the exact jump condition and have an optimal solution that doesn't meet the aggressive trigger criteria on one time step and then an infeasible solution on the next time step. As shown in Chapter III for the 2-D case, a two-phase formulation is useful to determine the exact trigger point where the aggressive criteria is met for at least one control to be at a max $\forall\ t \in [t_0, t_{CPA})$. In this way, a two-phase problem formulation for the 3-D and two control model would enable determination of the theoretical optimal trajectory based on the aggressive trigger criteria. The two-phase approach would be an appropriate method for determining the theoretical optimal trajectory in simulation or post-facto but may not be appropriate for real-time implementation.

R7. Investigate the proper optimal trajectory update rate (time step between each optimal path calculation) which will depend on the speed of the aircraft, aircraft performance, the time horizon, NLP solver time, etc. The time step must be greater than the NLP solver time, however, too large a time step will degrade the safety performance of the Auto GCAS algorithm.

Regarding the aircraft model, further research can be done to:

R8. Determine if the 3-DOF model is sufficient to model the candidate aircraft or if a higher-order model is necessary, keeping in mind the trade-off between accuracy and computational speed. Based on the literature, a 3-DOF model was considered adequate due to the short time horizons of the predicted trajectories. Recommendation is to implement this algorithm in a six-degrees of freedom (6-DOF) aircraft simulator and compare performance between the two models.

R9. Investigate using roll rate as a control (with appropriate bank angle and roll rate limits).

R10. Determine the effect of wind on algorithm performance. It was assumed for this research that due to the military aircraft weight and speed that wind did not

need to be incorporated. This may not be true for other applications such as general aviation aircraft and Remotely Piloted Aircraft (RPA).

R11. Evaluate the benefit of including auto-throttle as a control. Military aircraft like the B-1, C-17, and F-35 do have auto-throttle, and therefore using auto-throttle as a control input may be appropriate for future implementations of Auto GCAS.

6.3 Future of Automatic Collision Avoidance

In recent years, much progress has been made in the arena of automatic collision avoidance technology. AFRL and Lockheed Martin have fielded the F-16 Auto GCAS, developed an Analog F-16 Auto GCAS, developed an F-16 Automatic Air Collision Avoidance System (Auto ACAS), and are currently integrating Auto GCAS and Auto ACAS together as a system called Automatic Integrated Collision Avoidance System (Auto ICAS). AFRL's Auto ACAS flight testing was successfully completed in September 2014 [18]. The areas of improvement identified during flight test will be incorporated and integrated with Auto GCAS into Auto ICAS [18, 66]. The USAF TPS is scheduled to perform risk reduction flight testing for Auto ICAS in the fall of 2015 which will be followed by a larger scale Auto ICAS flight test program in 2016 and 2017 [18, 66]. The Swedish Air Force plans to field Auto GCAS on its Gripen fighters in 2015. The Line-in-the-Sky variant of Auto GCAS has been flight tested on the F-22 and is planned to be fielded in 2015, and Auto GCAS is planned to be fielded on the F-35 no later than 2024. NASA developed and flight tested a Small UAV Auto GCAS, proved the DTED map and the Auto GCAS algorithm could be stored on an on-board smart phone, and is currently developing a manual GCAS for general aviation aircraft as an app on a smart phone or tablet [83, 84]. The technology now being developed for Auto ACAS in fighters could be extended to heavy aircraft and UAVs/RPAs in the future when needed [18]. This research has shown that the

technology exists to integrate Auto GCAS on heavy aircraft. However, a vetted requirement and funding for research and flight test to fully develop the system on candidate aircraft are required.

This Optimal Auto GCAS research is a step forward in the DoD's quest for continual improvements in flight safety. This research has demonstrated that an optimal control approach can be applied to Auto GCAS and has applications to military heavy aircraft. The Min Control with an Aggressive Trigger technique developed in this research will be used as an evaluation tool for a limited Multi-Trajectory Auto GCAS flight test on the variable stability Learjet at USAF TPS in the fall of 2015 [91]. Results from the TPS flight tests will provide more insight into the efficacy of an Auto GCAS for heavy aircraft and propel the DoD one step further on the path towards reducing CFIT and other Auto GCAS-preventable mishaps in heavy aircraft.

Appendix A. Building the DTED Matrices in MatLab

A.1 Building the DTED map in Lat, Lon, Altitude (LLA) Format

The DTED matrices are constructed in latitude, longitude, and altitude (LLA) format. The Z matrix terrain heights from the DTED source data are provided in meters, and the $XLonMatrix$ and $YLatMatrix$ built in this section are in arclengths (degrees).

1. Inputs to 'dted' function

- South/North Latitude Limits

- West/East Longitude Limits

- Specify type of DTED file - Level 0, 1, or 2

2. Outputs from 'dted' function

- Z matrix of terrain heights

- Reference vector

- Latitude and Longitude to define Southwest corner of DTED grid

3. Build X Matrix (Longitude)

- extract cells/degree from reference vector

- degrees/cell = 1/(cells/degree)

- create row vector of longitude posts from West to East

– $Xvector = lonlimit_{West} : degrees/cell : lonlimit_{East}$

- create XLonMatrix by replicating rows to form a square matrix

4. Build Y Matrix (Latitude)

- create row vector of latitude posts from South to North

– $Yvector = latlimit_{South} : degrees/cell : latlimit_{North}$

- create YLatMatrix by replicating rows to form a square matrix

– transpose YLatMatrix so that it is now a series of columns

5. Plot 3D DTED in LLA Format (surface or mesh plot for example)

- $surf(XLonMatrix, YLatMatrix, Z)$

A.2 Reference Frames

A.2.1 Mapping of DTED from LLA to gridded ENU Format.

The function 'lla2griddedenu' was created to perform the calculations for the mapping from LLA to gridded ENU, using the following affine transformation.

$$\mathbf{X}_{griddedENU} = (\mathbf{X}_{Lon} - Lon_0) * \left(\frac{meters}{deg\ of\ longitude} \right) \tag{119}$$

$$\mathbf{Y}_{griddedENU} = (\mathbf{Y}_{Lat} - Lat_0) * \left(\frac{meters}{deg\ of\ latitude} \right) \tag{120}$$

where $Lat_0 =$ and $Lon_0 =$ are defined as the South and West corner reference points of the DTED grid.

A.2.2 Reverse Mapping from gridded ENU to LLA Format.

The optimal path calculated by the NLP solver is in the gridded ENU format. In order to have the optimal path data in LLA format (especially useful for flight test), a reverse mapping is performed from gridded ENU to LLA. The function 'griddedenu2lla' was created to perform the calculations for the reverse mapping back to LLA format, using the following affine transformation:

$$\mathbf{X}_{Lon} = \mathbf{X}_{griddedENU} * \left(\frac{deg\ of\ longitude}{meter} \right) + Lon_0 \tag{121}$$

$$\mathbf{Y}_{Lat} = \mathbf{Y}_{griddedENU} * \left(\frac{deg\ of\ latitude}{meter} \right) + Lat_0 \tag{122}$$

Appendix B. Interpolation Methodology

B.1 Interpolation Methodology

The continuous surface approximation of the DTED is created outside the NLP solver using the MatLab 'griddedInterpolant' function [59] as shown in Equation (123).

$$F = griddedInterpolant(X_{DTED}, Y_{DTED}, Z_{DTED}, \text{'spline'}) \tag{123}$$

Then inside the NLP solver, the 'griddedInterpolant' function F is simply queried in the z dimension using the optimal path x^* and y^*. The clearance above the terrain is then calculated by subtracting the interpolated height of the terrain from the altitude of the aircraft from the optimal path z^* as shown in Equation (124).

$$Zinterp_{center} = F(x^*, y^*) \tag{124a}$$

$$Clearance_{center} = z^* - Zinterp_{center} \tag{124b}$$

$$350 \text{ ft} \leq Clearance_{center} \leq \text{Service Ceiling} \tag{124c}$$

Then the 'griddedInterpolant' function F representing the surface approximation is queried for the left and right constraint paths as well as at x^*, y^*, and and then the

clearance for the left and right paths are calculated as shown in Equation (125).

$$Zinterp_{left} = F(x_{left}, y_{left}) \tag{125a}$$

$$Clearance_{left} = z^* - Zinterp_{left} \tag{125b}$$

$$0 \text{ ft} \leq Clearance_{left} \leq \text{Service Ceiling} \tag{125c}$$

$$Zinterp_{right} = F(x_{right}, y_{right}) \tag{125d}$$

$$Clearance_{right} = z^* - Zinterp_{right} \tag{125e}$$

$$0 \text{ ft} \leq Clearance_{right} \leq \text{Service Ceiling} \tag{125f}$$

Appendix C. Path Propagation Criteria Methodology

C.1 Methodology to use the Path Propagation Criteria

When an Auto GCAS is to be integrated into a candidate aircraft, then Trombetta's path propagation criteria (PPC) will need to be applied for the specific mission airspeed, specific terrain type, g for the Auto GCAS recovery, bank angle for the Auto GCAS recovery, and max flight path angle for that aircraft. The methodology to apply Trombetta's criteria is

1. **Set mission airspeed and max flight path angle in simulation**

2. **Forward Avoidance Path**:
   a. Setup simulation for wings-level pull (at g for the Auto GCAS pull-up recovery)
   b. Upland: Run simulation until 4,000 ft of climb is achieved. Record Distance Covered d_{climb} (for 4,000 ft climb). Record $t_{fwd\ path}$.
   c. Midland: Using d_{climb}, apply 500 ft per $\frac{1}{2}$ NM to calculate the required Minimum Altitude Climbed. Run simulation until Min Altitude Climbed is achieved. Record $t_{fwd\ path}$.
   d. Lowland: Using d_{climb}, apply 250 ft per $\frac{1}{2}$ NM to calculate the required Minimum Altitude Climbed. Run Simulation until Min Altitude Climbed is achieved. Record $t_{fwd\ path}$.

3. **Lateral Avoidance Paths**:
   a. Setup simulation for level turn (at g and bank angle for Auto GCAS lateral recovery)
   b. Upland/Midland/Lowland: Run simulation until 90° of turn is achieved. Record $t_{90°}$.

4. **Compare $t_{90°}$ & $t_{fwd\ path}$ for the Lowland, Midland, and Upland cases**. Path Propagation Time = $\max(t_{90°}, t_{fwd\ path})$

Appendix D. Settings for the GPOPS-II Software

D.1 GPOPS-II Settings

The GPOPS-II settings used for both the Max Distance with a Timely Trigger and Min Control with an Aggressive Trigger algorithms are listed in Table 26. GPOPS-II version 2.0 (May 2014) is used in this research. IPOPT was used as the nonlinear programming (NLP) solver for all of the scenarios in this research.

Table 26. GPOPS-II settings for Max Distance and Min Control Algorithms

GPOPS-II option	Setting
setup.nlp.solver	IPOPT (v3.11.0)
setup.nlp.options.ipopt.linear_solver	ma57
setup.nlp.options.tolerance	1e-5
setup.nlp.options.maxiterations	4000
setup.derivatives.supplier	sparseCD (sparse Center Difference)
setup.derivatives.derivativelevel	second
setup.derivatives.dependencies	sparseNaN (sparse Not a Number)
setup.method	RPM-Integration
setup.mesh.method	hp-LiuRao
setup.mesh.tolerance	1e-1
setup.mesh.maxiterations	10
min num of collocation points per interval	3
max num of collocation points per interval	10
setup.scales.method (scaling)	none (done manually)

D.2 State and Control Limits

Table 27 displays the state and control limits used in both the Max Distance and Min Control algorithms. These are the standard state and control limit settings used for the majority of the scenarios analyzed in this research. The limits for the state γ and the control N_z were changed when necessary due to varying performance characteristics of the low, med, and high-speed heavy aircraft. These exceptions are listed in the applicable sections below.

Table 27. State and Control Limits

		Minimum	Maximum	Units
States	x-location	DTED	Limits	NM
	y-location	DTED	Limits	NM
	z-location	0	Service Ceiling	feet
	flight path angle γ	-15	15	degrees
	heading angle χ	-180	180	degrees
Controls	load factor N_z	0	2	g
	bank angle μ	-60	60	degrees

D.3 Low and Medium-Speed Heavy Specific Settings

Table 28. Settings Unique to the Low and Medium-Speed Heavy Aircraft

Variable	Setting	Units
Velocity (low-speed)	210	NM/hour
Time horizon (low-speed)	45	seconds
Velocity (medium-speed)	310	NM/hour
Time horizon (medium-speed)	31	seconds

D.4 Low-Speed Heavy Settings for Comparison with Multi-Trajectory Auto GCAS

Table 29. Settings for Low-Speed Heavy Comparison with Multi-Trajectory Auto GCAS

Variable	Setting	Units
Velocity	210	NM/hour
Time horizon	45	seconds
Min flight path angle γ	-15.8	degrees
Max flight path angle γ	15.8	degrees

x

Bibliography

1. Agte, Jeremy. AFIT Assistant Professor of Aeronautics, Personal Communication, 2013-2014.

2. Aguilar, Fernando J, Francisco Agüera, Manuel A Aguilar, and Fernando Carvajal. "Effects of Terrain Morphology, Sampling Density, and Interpolation Methods on Grid DEM Accuracy". *Photogrammetric Engineering & Remote Sensing*, 71(7):805–816, 2005.

3. Air Force Flight Test Center (AFFTC). *Automated Recovery System flight testing Advanced Fighter Technology Integration/F-16 Close Air Support (AFTI/F-16 CAS) Program, Volume 1 - Program Summary*. Technical report, General Dynamics, 1993.

4. Anderson, Tom. Navy TAWS Algorithm Designer, Personal Communication, 2012-2014.

5. Anderson, Tom, Warren Jones, and Kathleen Beamon. "Design and Implementation of TAWS for Rotary Wing Aircraft". *IEEE Aerospace Conference*, 1–7. IEEE, Big Sky, MT, Mar 2011.

6. Aslinger, Jerry. OSD P&R Personnel Risk Reduction Program Manager, Personal Communication, 2013-2014.

7. Barfield, Arthur F. AFRL Auto GCAS Technical Advisor, Personal Communication, 2011-2014.

8. Barr, Alan H. "Superquadrics and Angle-Preserving Transformations". *IEEE Computer Graphics and Applications*, 1(1):11–23, 1981.

9. Benson, David. *A Gauss pseudospectral transcription for optimal control*. Ph.D. thesis, Massachusetts Institute of Technology, 2005.

10. Betts, JT and WP Huffman. *Sparse Optimal Control Software SOCS, Mathematics and Engineering Analysis Tech*. Technical report, Document MEA-LR-085, Boeing Information and Support Services, The Boeing Company, PO Box 3707, Seattle, WA 98124-2207, 1997.

11. Bicchi, Antonio and Lucia Pallottino. "On Optimal Cooperative Conflict Resolution for Air Traffic Management Systems". *IEEE Transactions on Intelligent Transportation Systems*, 1(4):221–231, 2000.

12. Biegler, L.T. and V.M. Zavala. "Large-scale nonlinear programming using IPOPT: An integrating framework for enterprise-wide dynamic optimization". *Computers & Chemical Engineering*, 33(3):575–582, 2009.

13. Bolkcom, Christopher. "Military Aviation Safety". *Library of Congress Congressional Research Service (CRS)*, 2004.

14. Bollino, Kevin P and L Ryan Lewis. "Collision-free multi-UAV optimal path planning and cooperative control for tactical applications". *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit, Honolulu, Hawaii*. 2008.

15. Bollino, Kevin P, L Ryan Lewis, Pooya Sekhavat, and I Michael Ross. "Pseudospectral Optimal Control: A Clear Road for Autonomous Intelligent Path Planning". *Proceedings of the AIAA InfoTech at Aerospace Conference and Exhibit*, 1228–1241. 2007.

16. Bresnik, Randy and Tom Anderson. "Developmental Flight Testing of the Terrain Awareness Warning System (TAWS) in the F/A-18 Hornet". *Society of Experimental Test Pilots Symposium (SETP)*, Sep 2001.

17. Burns, Amy, Daniel Harper, Arthur F Barfield, Shawn Whitcomb, and Brian Jurusik. "Auto GCAS for analog flight control system". *30th Digital Avionics Systems Conference (DASC)*, 8C5–1. IEEE/AIAA, Seattle, WA, Oct 2011.

18. Burns, Amy C. AFRL Auto GCAS Deputy Program Manager, Personal Communication, 2011-2014.

19. Burrough, P.A. *Principles of Geographical Information Systems for Land Resources Assessment*. Clarendon Press, Oxford, 1986.

20. Byrd, Richard H, Jorge Nocedal, and Richard A Waltz. "KNITRO: An integrated package for nonlinear optimization". *Large-scale nonlinear optimization*, 35–59. Springer, 2006.

21. Calhoun, Paul. 418 Flight Test Squadron Experimental Test Pilot, Personal Communication, Sep 2014.

22. Coenen, F.P., G.P. Smeaton, and A.G. Bole. "Knowledge-Based Collision Avoidance". *Journal of Navigation*, 42(1), 1989.

23. Darland, S.E. and K.L. Armbruster. *F-16 Digital Terrain System Concept of Operations*. Technical report, Lockheed, Fort Worth Division, 1995.

24. Defense Safety Oversight Council Aviation Safety Improvements Task Force. *Fighter/Attack Automatic Collision Avoidance Systems Business Case*. Technical Report AFRL Public Affairs, Control Number AFRL-WS 06-0093, Safety Technology Working Group (STWG) Defense Safety Oversight Council (DSOC), Feb 2006.

25. Department of Defense (DoD). "GPS Precise Postioning Service Performance Specification". Feb 2007.

26. Dragut, L. and T. Blaschke. "Automated Classification of Landform Elements using Object-Based Image Analysis". *Geomorphology*, 81(3):330–344, 2006.

27. Drake, Samuel Picton. *Converting GPS Coordinates [phi,lambda,h] to Navigation Coordinates (ENU)*. Technical Report DSTO-TN-0432, Defence Science and Technology Organisation Salisbury (Australia) Electronics and Surveiilance Research, 2002.

28. Egged, Doug. F-16 SPO Flight Test, Personal Communication, 2013-2014.

29. Eller, Bob G, Peter O Stanfill, Russell M Turner, Shawn C Whitcomb, Amy C Burns, Kerianne L Hobbs, and Donald E Swihart. "Test and Evaluation of a Modified F-16 Analog Flight Control Computer". *Infotech Guidance, Navigation, and Control and Co-located Conferences*. AIAA, Boston, MA, Aug 2013.

30. F-16.net. "AFTI/F-16 History website". http://www.f-16.net/news_article778.html, Accessed: Jan 2013.

31. Farr, Tom G, Paul A Rosen, Edward Caro, Robert Crippen, Riley Duren, Scott Hensley, Michael Kobrick, Mimi Paller, Ernesto Rodriguez, Ladislav Roth, et al. "The Shuttle Radar Topography Mission". *Reviews of Geophysics*, 45(2), 2007.

32. Federal Aviation Administration (FAA). "Installation of Terrain Awareness and Warning Systems (TAWS) Approved for Part 23 Airplanes". (AC No: 23-18), June 2000.

33. Federal Aviation Administration (FAA). "Safer Skies General Aviation (GA) Controlled Flight into Terrain (CFIT) Joint Safety Implementation Team (JSIT) Final Report". Feb 2000.

34. Fields, R. Andrew. *Continuous Control Artificial Potential Function Methods and Optimal Control*. Master's thesis, Air Force Institute of Technology, Mar 2014.

35. Frommelt, Paul. "NGA releases high-resolution elevation data to public". http://www1.nga.mil/MediaRoom/LeadingStories/Pages/NGAreleaseshigh-resolutionelevationdatatopublic.aspx, Accessed:10 Oct 2014.

36. Gates, Robert. "Zero Preventable Accidents". Secretary of Defense Memorandum, May 2007.

37. Geiger, Brian R, Joseph F Horn, Gregory L Sinsley, James A Ross, Lyle N Long, and Albert F Niessner. "Flight Testing a Real Time Implementation of a UAV Path Planner Using Direct Collocation". *Proceedings of the AIAA Guidance, Navigation, and Control Conference, Hilton Head, SC*. 2007.

38. Geiger, Brian R, Joseph F Horn, Gregory L Sinsley, James A Ross, Lyle N Long, and Albert F Niessner. "Flight testing a real-time direct collocation path planner". *AIAA Journal of Guidance, Control, and Dynamics*, 31(6):1575–1586, 2008.

39. Gill, Philip E, Walter Murray, and Michael A Saunders. "SNOPT: An SQP algorithm for large-scale constrained optimization". *SIAM Journal on Optimization*, 12(4):979–1006, 2002.

40. Griffin, Edward M, Russell M Turner, Shawn C Whitcomb, Donald E Swihart, James M Bier, Kerianne L Hobbs, and Amy C Burns. "Automatic Ground Collision Avoidance System Design for Pre-Block 40 F-16 Configurations". *Asia-Pacific International Symposium on Aerospace Technology*. APISAT, Jeju, Korea, Nov 2012.

41. Griswold, M.R. *Advanced Fighter Technology Integration/F-16 Close Air Support (AFTI/F-16 CAS) Program, Volume 1 - Program Summary*. Technical report, General Dynamics, 1992.

42. Honeywell. *2004 MK VI and MK VIII Enhanced Ground Proximity Warning System (EGPWS) Pilot's Guide*.

43. Honeywell. *2011 MK V and MK VII Enhanced Ground Proximity Warning System (EGPWS) Pilot's Guide*.

44. Honeywell. "Honeywell EGPWS website". http://www51.honeywell.com/aero/Products-Services/Avionics-Electronics/EGPWS-Home.html?c=21, Accessed: Jul 2013.

45. Hook, Loyd and Mark Skoog. "Auto GCAS Flight Test Briefing to the NTSB", 2012.

46. Hook, Loyd and Mark Skoog. "Small UAV GCAS CDTM Description", 2013.

47. Huntington, Geoffrey T. *Advancement and analysis of a Gauss pseudospectral transcription for optimal control problems*. Ph.D. thesis, Massachusetts Institute of Technology, 2007.

48. Huntington, Geoffrey T, David Benson, and Anil V Rao. "A Comparison of Accuracy and Computational Efficiency of Three Pseudospectral Methods". *AIAA Guidance, Navigation, and Control Conference*, 1–43. AIAA, Hilton Head, SC, Aug 2007.

49. ICAO Industry Controlled Flight Into Terrain (CFIT) Task Force. *Controlled Flight Into Terrain Education and Training Aid*. Technical report, International Civil Aviation Organization (ICAO), Nov 1995.

50. Iijima, Yukito, Hideki Hagiwara, and Hironao Kasai. "Results of Collision Avoidance Maneuver Experiments Using a Knowledge-Based Autonomous Piloting System". *Journal of Navigation*, 44(02):194–204, 1991.

51. Jennings, Gareth. "Saab to offer Gripen C/D upgrades, pushes exports". *IHS Jane's Defence Weekly (online)*, 9 Mar 2014. http://www.janes.com/article/35119/saab-to-offer-gripen-c-d-upgrades-pushes-exports#.VC2N55y0qQ0.email.

52. Jorris, Timothy R. *Common Aero Vehicle Autonomous Reentry Trajectory Optimization Satisfying Waypoint and No-Fly Zone Constraints*. Ph.D. thesis, Air Force Institute of Technology, 2007.

53. Kirk, Donald E. *Optimal control theory: an introduction*. Courier Dover Publications, 2012.

54. Kuchar, J. K. and L. C. Yang. "Survey of Conflict Detection and Resolution Modeling Methods". *in Proceedings AIAA Guidance Navigation Control Conference*, 1997.

55. Kuchar, J. K. and L. C. Yang. "A Review of Conflict Detection and Resolution Modeling Methods". *IEEE Transactions on Intelligent Transportation Systems*, 1(4):179–189, 2000.

56. Lehmann, R.C. and K.W. Trent. *F-16 Ground Proximity Warning System Study*. Technical report, General Dynamics, Fort Worth Division, 1987.

57. Lockheed Martin. *F-16 M2 Modular Mission Computer Avionics Systems Manual*. Technical report, Lockheed, Fort Worth Division, 1997.

58. Lockheed Martin. "Auto GCAS Workshop: Digital Terrain Topics", Jan 2011.

59. Mathworks. *Interpolating Gridded Data Documentation*, matlab 2014a edition, 2014.

60. Middleton, Miles. 419 Flight Test Squadron Experimental Test Pilot, Personal Communication, Sep 2014.

61. Moroze, M. L. and M. P Snow. *Causes and Remedies of Controlled Flight Into Terrain in Military and Civil Aviation*. Technical report, Air Force Research Laboratory (AFRL), 1999.

62. National Aeronautics and Space Administration (NASA). "Shuttle Radar Topography Mission website". http://www2.jpl.nasa.gov/srtm/, Accessed: Jul 2013.

63. National Geospatial-Intelligence Agency (NGA). "NGA website". http://www1.nga.mil, Accessed: Oct 2014.

64. National Imagery and Mapping Agency (NIMA). "Performance Specification DTED, MIL-PRF-89020B". 23 May 2000.

65. Naval Air Systems Command (NAVAIR). "PMA 209 GPWS/TAWS website". http://www.navair.navy.mil/PMA209/Teams/GPWS.aspx, Accessed: 15 Dec 2012.

66. Norris, Guy. "Collision Avoidance Tests Pave the Way for Combined Air-Ground Fighter Protection System". *Aviation Week & Space Technology (online)*, 1 Dec 2014. http://aviationweek.com/node/1226121.

67. Patel, Rushen B and Paul J Goulart. "The Design of Trigger Mechanisms for Aircraft Collision Avoidance Maneuvers". *Proceedings of the AIAA Guidance, Navigation and Control Conference, Toronto, Ontario Canada*. 2010.

68. Patel, Rushen B and Paul J Goulart. "Trajectory generation for aircraft avoidance maneuvers using online optimization". *AIAA Journal of Guidance, Control, and Dynamics*, 34(1):218–230, 2011.

69. Patterson, Michael A and Anil V Rao. "GPOPS-II: A MATLAB Software for Solving Multiple-Phase Optimal Control Problems Using hp–Adaptive Gaussian Quadrature Collocation Methods and Sparse Nonlinear Programming". 2013.

70. Pellebergs, Johan. Saab Engineer, Personal Communication, Oct 2014.

71. Prosser, Kevin. "Next-Next Gen Auto-GCAS". Presentation at Society of Experimental Test Pilots San Diego Symposium, March 2013.

72. Raghunathan, Arvind U, Vipin Gopal, Dharmashankar Subramanian, Lorenz T Biegler, and Tariq Samad. "Dynamic Optimization Strategies for Three-Dimensional Conflict Resolution of Multiple Aircraft". *AIAA Journal of Guidance, Control, and Dynamics*, 27(4):586–594, 2004.

73. Rao, Anil V. "A Primer on Pseudospectral Methods for Solving Optimal Conrol Problems, AFIT lecture", Mar 2012.

74. Rao, Anil V, David A Benson, Christopher Darby, Michael A Patterson, Camila Francolin, Ilyssa Sanders, and Geoffrey T Huntington. "Algorithm 902: GPOPS, A Matlab Software for Solving Multiple-Phase Optimal Control Problems Using the Gauss Pseudospectral Method". *ACM Transactions on Mathematical Software (TOMS)*, 37(2):1–39, 2010.

75. Rao, Anil V, Camila Francolin, Christopher L. Darby, and Michael Patterson. "User's Manual for PSCOL Version 1.0: A MatLab Package for Dynamic Optimization".

76. Raquet, John. "EENG 533: Navigation Using the GPS Course Notes". Air Force Institute of Technology, 2014.

77. Ross, I Michael. "A Beginner's Guide to DIDO (Ver. 7.3)". *Document# TR-710, Elissar, LLC, Monterey, CA*, 2007.

78. Ross, S.M., R.G. Cobb, and W.P. Baker. "Stochastic Real-Time Optimal Control for Bearing-Only Trajectory Planning". *International Journal of Micro Air Vehicles*, 6(1):1–28, March 2014.

79. Rumsfeld, Donald. "Reducing Preventable Accidents". Secretary of Defense Memorandum, May 2003.

80. Skoog, Mark. "Automatic Collision Avoidance Technologies: Auto-GCAS Flight Test Development & Evaluation". Presentation at Unmanned Vehicle Systems International Conference, Paris, June. 2007.

81. Smith, Nathan E. *Optimal Collision Avoidance Trajectories via Direct Orthogonal Collocation for Unmanned/Remotely Piloted Aircraft Sense and Avoid Operations*. Ph.D. thesis, Air Force Institute of Technology, 2014.

82. Snyder, Gregory I. *The 3D Elevation Program–Summary of Program Direction: U.S. Geological Survey Fact Sheet 2012-3089*. Technical report, USGS Document http://nationalmap.gov/3DEP, 2012.

83. Sorokowski, P., M. Skoog, S. Burrows, and S. Thomas. *Small UAV Automatic Ground Collision Avoidance System Design Considerations and Flight Test Results*. Technical Report NASA-TM-2015-XXXXXX unpublished, NASA Dryden Flight Research Center, Edwards AFB, CA, 2015.

84. Sorokowski, Paul. 773 Test Squadron Collision Avoidance Technical Expert, Personal Communication, 2012-2014.

85. Stickney, Heather M. *Performance Characterization, Development, and Application of Artificial Potential Function Guidance Methods*. Master's thesis, Air Force Institute of Technology, Mar 2014.

86. Sweetman, Bill. "New Gripen Aims For Low Cost, High Capability". *Aviation Week & Space Technology (online)*, 17 Mar 2014. http://aviationweek.com/awin/new-gripen-aims-low-cost-high-capability?eid=forward.

87. Swihart, D.E., C. Wiedemann, D.B. Homan, E.M. Griffin, M.A. Skoog, A.F. Barfield, R.M. Turner, E. Smith, and W. Black. "Automatic Collision Avoidance Technology". *Unmanned Vehicle Systems International Conference*. Unmanned Vehicle Systems International Conference, Paris, France, Jun 2007.

88. Swihart, Donald, Arthur Barfield, Edward Griffin, Richard Lehmann, Shawn Whitcomb, Billie Flynn, Mark Skoog, and Kevin Prosser. "Automatic Ground Collision Avoidance System Design, Integration, and Flight Test". *IEEE Aerospace and Electronic Systems Magazine*, 4–15, May 2011.

89. Swihart, Donald E. AFRL Auto GCAS Program Manager, Personal Communication, 2011-2014.

90. Taylor, D. H. "Uncertainty in Collision Avoidance Maneuvering". *Journal of Navigation*, 43(02):238–245, 1990.

91. Trombetta, John V. "Multi-Trajectory Automatic Ground Collision Avoidance System with Flight Tests", Mar 2016. Unpublished.

92. United States Air Force (USAF). "Air Force Instruction 11-202, Vol. 3". *Flying Operations: General Flight Rules*, 2010.

93. United States Geological Survey (USGS). "Land Processes Distributed Active Archive Center website". http://lpdaac.usgs.gov/products/measures_products_table/srtmus1n, Accessed: Oct 2014.

94. United States Geological Survey (USGS). "National Elevation Dataset website". http://ned.usgs.gov/, Accessed: Oct 2014.

95. United States Naval Academy (USNA). "United States Naval Academy, Oceanography Department, SO432, Geographical Information Systems, Chapter 5". www.usna.edu/Users/oceano/pguth/website/so432web/e-text/GEODUC_book/Matrix%20Products_Ch_5.doc, Accessed: Oct 2014.

96. Weingarten, Norman C. "History of In-Flight Simulation & Flying Qualities Research at Calspan". *AIAA Journal of Aircraft*, 42(2):290–298, Mar/Apr 2005.

97. Wilkins, Mark. "Automatic Collision Avoidance Technologies (ACAT) Update: Auto-GCAS and the Fighter Risk Reduction Program (FRRP) Briefing", Jan 2007.

98. Wilkins, Mark. OSD P&R Senior Aviation Safety Analyst, Personal Communication, 2012-2014.

99. Yechout, Thomas R, S Morris, D Bossert, and W Hallgren. *Introduction to Aircraft Flight Mechanics: Performance, Static Stability, Dynamic Stability, and Classical Feedback Control*. AIAA, 2003.

Vita

Col Angela W. Suplisson grew up in Las Vegas, NV, where she graduated from Ed W. Clark High School in 1987 before joining the US Air Force. She holds a B.S. in Aeronautical Engineering from the US Air Force Academy (USAFA), an M.E. in Aerospace Engineering from the University of Florida, an M.A. in Education (Curriculum and Instruction) from Chapman University, and a Master's in International Strategy and Policy from the University of Lyon III, France. Upon commissioning as a Second Lieutenant, she was an Electronic Warfare and Weapons Test Engineer at Eglin AFB, FL, testing F-15s, F-16s, and F-5s. She is a Flight Test Engineer graduate of the USAF Test Pilot School (TPS), Class 95A. Following TPS, she tested F-16s at Edwards AFB, CA, and F-117s at Plant 42, Palmdale, CA. As a Captain, she was selected for the Olmsted Scholarship Program through which she studied French at the Defense Language Institute and political science at l'Institut d'Etudes Politiques (Sciences-Po) and University of Lyon III in France. As a Major, she was the Executive Officer to the Program Executive Officer of the Joint Strike Fighter Program as well as an International Program Manager. Following this assignment, she was the US F-16 and F-16 Multi-National Fighter Program (MNFP) Program Element Monitor in SAF/AQP in the Pentagon where she ran the F-16 budget. She then commanded the Holloman High Speed Test Track/846th Test Squadron (Go Mach 10!) at Holloman AFB, NM, as a Lieutenant Colonel. Next, she went to USAFA where she had the privilege of teaching cadets aeronautics and was the Deputy Head of the Department of Aeronautics. After completing her PhD at AFIT, she returned to USAFA as an Assistant Professor of Aeronautics.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
26-03-2015	PhD Dissertation	Sept 2011 — Jan 2015

4. TITLE AND SUBTITLE

Optimal Recovery Trajectories for
Automatic Ground Collision Avoidance Systems (Auto GCAS)

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Suplisson, Angela W., Col, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENY-DS-15-M-002

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory, Aerospace Systems Directorate (AFRL/RQQC)
Attn: Amy Burns
2210 8th Street
WPAFB OH 45433-7542
DSN 674-6542, COMM 937-904-6542, amy.burns.3@us.af.mil

10. SPONSOR/MONITOR'S ACRONYM(S)

AFRL/RQQC

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Distribution Statement A. Approved for Public Release; Distribution Unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The USAF's F-16 Automatic Ground Collision Avoidance System (Auto GCAS) uses a single pre-planned roll to wings-level and 5-g pull-up to meet the operational requirements of being both aggressive and timely, meaning that extremely agile avoidance maneuvers will be executed at the last second to avoid the ground. There currently exists no similar Auto GCAS for manned military 'heavy' aircraft with lower climb performance such as transport, tanker, or bomber aircraft. This research proposes a new optimal control approach to the ground collision avoidance problem for heavy aircraft by mapping the aggressive and timely requirements of the automatic recovery to an optimal control formulation which includes lateral maneuvers around terrain. Results are presented for representative heavy aircraft scenarios against 3-D digital terrain, which are then compared to a Multi-Trajectory Auto GCAS with five pre-planned maneuvers. Metrics were developed to quantify the improvement from using an optimal approach versus the pre-planned maneuvers. The research results provide a basis to evaluate the expected performance of any future Auto GCAS for all aircraft.

15. SUBJECT TERMS

Auto GCAS, Automatic Ground Collision Avoidance, Optimal Control, Optimization, Direct Orthogonal Collocation

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. Richard G. Cobb, AFIT/ENY
U	U	U	UU	226	19b. TELEPHONE NUMBER *(include area code)* (937) 255-3636, x4559; richard.cobb@afit.edu