

ARTIFICIAL POTENTIAL FUNCTION GUIDANCE FOR AUTONOMOUS IN-
SPACE OPERATIONS

By

ANDREW R TATSCH

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2006

Copyright 2006

by

Andrew R Tatsch

Dedicated to

My parents who brought me into this world and
My friends, family, and colleagues who opened my eyes to it.

ACKNOWLEDGMENTS

To my advisor, Dr. Norman Fitz-Coy, I would like to convey my gratitude. Few graduate students have the opportunity to work with, not for, an advisor that encourages freedom and creativity in the search for a pragmatic solution and uniquely challenges them beyond what they think is possible. He also taught me many lessons beyond the curriculum and for those I am grateful. Without his guidance this dissertation would never have come to fruition.

Many thanks also go out to my supervisory committee, Drs. Rick Lind, Andrew Kurdila, Gloria Wiens, BJ Fregly, and Stanley Dermott, for their invaluable assistance and patience.

Lastly, I would like to acknowledge my brother J Bradley Tatsch for everything that he has done to make this dream a reality. Without his assistance my undergraduate and graduate experiences would have never gotten off the ground. I am forever grateful for all that he has done for me.

TABLE OF CONTENTS

	<u>page</u>
ACKNOWLEDGMENTS	iv
LIST OF TABLES	vii
LIST OF FIGURES	viii
ABSTRACT	x
CHAPTER	
1 INTRODUCTION	1
State of the Art for In-Space Operations (ISO) Functionalities	4
Research Scope	11
Dissertation Outline	12
2 TRAJECTORY DETERMINATION: ORBIT MECHANICS APPROACH.....	14
Midcourse Trajectory Determination	14
Lambert’s Problem	16
Maneuver Command Determination	19
Numerical Example: Midcourse	21
Endgame Trajectory Determination	22
CW Maneuver Command Determination.....	24
Numerical Example: Endgame	25
Alternative Orbit Mechanics Based Methods.....	26
Conclusions.....	28
3 ARTIFICIAL POTENTIAL FUNCTION POSITION GUIDANCE.....	29
Robot Motion Planning.....	29
Artificial Potential Function Position Guidance.....	32
Example: Spacecraft Unconstrained Rendezvous	35
Obstructions.....	37
Numerical Example: Path-Constrained Rendezvous.....	42
Conclusions.....	44
4 ARTIFICIAL POTENTIAL FUNCTION GUIDANCE: A NEW APPROACH.....	45

Dynamic Artificial Potential Function Obstacle Avoidance Algorithm.....	45
Attractive Potential.....	46
Repulsive Potential.....	47
Total Force Interactions.....	49
Simulation Results.....	50
Constant velocity scenario	50
On-orbit rendezvous scenario.....	52
Optimized Harmonic Attractive Potential	54
Conclusions.....	57
5 ARTIFICIAL POTENTIAL FUNCTION ATTITUDE GUIDANCE	60
Attitude Equations of Motion	63
Continuous APF Attitude Guidance	64
Stability Analysis.....	67
Stability of Dynamical System.....	72
Optimization.....	74
Numerical Example	75
Impulsive APF Attitude Guidance.....	79
Numerical Example: Impulse-based APF Attitude Guidance	82
Conclusions.....	85
6 CONCLUSION AND FUTURE WORK	87
Conclusion.....	87
Future Work.....	88
LIST OF REFERENCES	90
BIOGRAPHICAL SKETCH	96

LIST OF TABLES

<u>Table</u>		<u>page</u>
1	Orbit parameters for rendezvous simulation.	52
2	Genetic algorithm statistics	57
3	Mean and standard deviations for the two parameters in the GA optimization	57
4	Genetic algorithm statistics	74
5	Mean and standard deviation for the six parameters in the GA optimization.....	75

LIST OF FIGURES

<u>Figure</u>	<u>page</u>
1 Model of the ROTEX manipulator.....	5
2 Artist's rendition of ETS-VII components	5
3 CMU's Skywalker prototype.....	6
4 NASA's Robonaut	6
5 University of Maryland's Ranger	6
6 AERCam Sprint.....	9
7 Mini AERCam.....	9
8 Artist's depiction of the Orbiter Boom Sensor System	9
9 The SCAMP in the neutral buoyancy tank.....	9
10 Schematic of Lambert's Problem.....	15
11 Transfer trajectory from midcourse trajectory determination algorithm.....	23
12 Depiction of orbital rendezvous	23
13 Endgame maneuver trajectory in the xy-plane (R-bar V-bar plane).....	26
14 Glideslope solution for 5 impulse rendezvous.	27
15 Planar rendezvous in the CW frame under APF guidance.....	36
16 Control efforts in x and y directions for the rendezvous.....	36
17 Difference between IC potential and potential on surface of obstructions	40
18 Contour plot of the potential function.....	40
19 Zoomed view of contour plot.....	41
20 Stem plot of obstruction surface potential error.....	41

21	Trajectory traversed by robot for the path-constrained rendezvous example	42
22	Control action in the x-direction for the constrained rendezvous	43
23	Control action in the y-direction for the constrained rendezvous	43
24	2D Constant velocity simulation results.....	51
25	Planar on-orbit rendezvous results.	53
26	Norm of applied chaser accelerations.	54
27	Optimized sink-vortex pair vs. CW based vector field	58
28	Inequality of Eq. (64)	72
29	Behavior of q_4	73
30	Time response of vector part of quaternion	76
31	Time response of scalar part of quaternion	77
32	Time response of angular velocity	78
33	Time response of control effort.....	79
34	Components of the vector part of the quaternion vs. time	82
35	Components of the angular velocity vs. time	83
36	Scalar part of the quaternion vs. time.....	83
37	Angular velocity impulses vs. time for the numerical example	84
38	Scalar part of the quaternion vs. time for minus one convergence case	85

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

ARTIFICIAL POTENTIAL FUNCTION GUIDANCE FOR AUTONOMOUS IN-
SPACE OPERATIONS

By

Andrew R Tatsch

May 2006

Chair: Norman Fitz-Coy

Major Department: Mechanical and Aerospace Engineering

For most in-space operations (ISO), e.g., assembly, repair, refueling, the target location and its orbit parameters are known a priori. For instance in the case of an assembly mission, the general location of the construction site and the parts depot are known parameters in mission planning. Utilizing this known information, we propose a novel approach to collision-free operations for a heterogeneous team of spacecraft robots performing the ISO mission based on artificial potential function (APF) guidance.

In order to utilize APF guidance two prominent deficiencies of APF methodologies need to be circumvented: premature termination due to local minima in the operational space and suboptimal performance. It is widely known that utilization of harmonic potential functions eliminates the introduction of local minima in the operational space; thus we employ harmonic functions to conquer the first of the aforementioned deficiencies. In order to surmount the suboptimal performance, we propose the use of a sum of harmonic function primitives (simple functions such as sinks and vortices in the

hydrodynamic analog) to shape the velocity field of the operational space to harness the underlying dynamics of the system (i.e., orbit mechanics). Furthermore, given the location and orbit parameters of the target site, a random heuristic search algorithm can be employed to optimize the velocity field shaping for maximizing the performance gains achieved from harnessing these underlying dynamics.

CHAPTER 1 INTRODUCTION

I believe that this nation should commit itself to achieving the goal, before this decade is out, of landing a man on the Moon and returning him safely to the Earth.¹

President John F. Kennedy, May 25, 1961

Just as the speech given forty-three years ago on a Thursday in late May by President John F. Kennedy presented the National Aeronautics and Space Administration (NASA) a new directive, on January 14, 2004, President George W. Bush delivered a speech titled “Vision for Space Exploration,”² whose intent was to again unify NASA around a set of common goals. These goals are centered around sending man and/or robotic missions to the Moon, Mars, and deeper into our solar system. Implicit in the robotic missions proposed by this new space exploration mandate are the utilization and development of autonomous space robotic systems that “will serve as trailblazers, advance guard to the unknown.”² Based on these new directives, autonomous space robotic systems will be imperative for the success of NASA’s new vision.

Space robotics, in the context of this work, are robotic systems whose applications can be roughly dichotomized into two broad classes of missions: in-space operations (ISO), and planetary surface exploration (PSE). Both of these mission profiles share common functional requirements and possess functional requirements unique to each mission. The functionalities typically associated with in-space operations are assembly (including maintenance of pre-existing structures) and inspection. It is important to note that although treated separately, the assembly and inspection functionalities share

common subtasks, such as navigation and maneuvering. Planetary surface exploration (PSE) functionalities include surface mobility (which includes navigation and collision avoidance), instrument deployment and sample manipulation, science planning and execution, and human exploration assistance. Even though space robotics applications are categorized into these two mission classes, generalization of the functionalities listed above allows for other mission profiles such as terrestrial assembly or in-space science. The research presented in the subsequent chapters is primarily applicable for in-space operations; therefore those will be the primary focus here.

The area of in-space robotics operation is far less mature than robotic planetary surface exploration systems, both in terms of real world implementations and theoretical research. Real world implementations of planetary surface exploration have been significantly more numerous, and more successful, than in-space robotics implementations with examples of teleoperated surface exploration robots dating back to the 1970's with the Soviet Lunakhod Rover.³ The sophistication of more recent PSE system successes is evidenced by the Mars Exploration Rovers⁴ developed at NASA's Jet Propulsion Laboratory (JPL). In 1997 the Sojourner, although not designed for advanced capabilities, was the first Mars rover to successfully demonstrate the ability to teleoperate a vehicle on Mars. Most recently the two more sophisticated Mars rovers, Spirit and Opportunity, have demonstrated effective teleoperational capabilities in surface mobility and terrestrial science.

The *raison d'être* that led to this disparity between PSE and ISO implementations is twofold. First, the theory and experience of terrestrial robotics are directly applicable to PSE scenarios. Volumes of research since the 1950's have been devoted to the

kinematics and control of terrestrial robot manipulators with achievements in autonomous navigation, perception, and mobility. Therefore, implementation of PSE robots spring-boarded off the terrestrial robotics knowledge base and needed only to address the technical issues associated with high degrees of mobility, science planning, perception, and exploration capabilities, the harshness of space and space transport, and the high latency communication links implicit in space teleoperation. Conversely, with the exception of autonomous rendezvous and docking (ARD) investigations⁵ that date back to October 1967 in Russia⁶ and preliminary investigation of in-space proximity operations,^{7,8} it has only been in the last decade that autonomous, free-flying robotics applications have appeared in the literature.^{9,10} Therefore, both the underlying theory and its implementation need much further investigation in order to achieve the maturity of PSE robotics systems to date.

Second, most terrestrial and PSE agents are developed without consideration of the system dynamics: they are typically kinematically driven, and the cases that do consider some dynamics do so only in the sense of constraints on the motion, e.g., non-holonomic constraints. For ISO applications, consideration of the system dynamics is essential. This fundamental difference is illustrated by the local operator in each case. In terrestrial or PSE applications, the local operator may be zero. This means that under zero command, the robot remains stationary until it is commanded to move. The local operator in ISO systems, however, is in general not zero. Instead, the local operator is the system dynamics, meaning under zero command, an ISO system will not remain stationary but move according to the topology shaped by the system dynamics. Therefore, consideration of the dynamics of ISO systems is imperative.

The sections that follow outline the state of the art for each of the functionalities of ISO systems for both current implementations and experimental test beds. The expectations for the sophistication of these functionalities over the next decade are also discussed.

State of the Art for In-Space Operations (ISO) Functionalities

In this section both current state of the art and experimental test bed sophistication are assessed for assembly and inspection functionalities of ISO systems. The relatively few implementations of ISO systems to date are emphasized while noting the progress possible from the experimental investigations.

The current state of assembly functionality, with respect to real world implementations, is restricted to the Shuttle Remote Manipulator System (SRMS) and the Space Station Remote Manipulator System (SSRMS) aboard the International Space Station (ISS). Neither of the current systems possesses any autonomy as they are solely teleoperated on-board or from the ground. However, to date there have been on-orbit demonstrations of assembly subsystems such as the Robot Technology Experiment¹¹ (ROTEX), a German experiment flown by NASA, and Engineering Test Satellite⁹ (ETS-VII), flown by the Japanese Aerospace Exploration Agency (JAXA) (formerly National Space Development Agency of Japan (NASDA)). The ROTEX was a robotic arm (see Figure 1) that flew in 1993 on Columbia as part of STS-55, and successfully completed multiple tasks that include replacement of a simulated Orbital Replacement Unit (ORU) and capture of a free-flying object via on-board and ground teleoperation and autonomous scripts. By accomplishing tasks from autonomous scripts during the experiment, ROTEX became the first autonomous space robotic system. The ETS-VII mission was composed of a passive target satellite, Orihime, and a chaser satellite,

Hikoboshi, with a robotic manipulator arm (see Figure 2). ETS-VII successfully demonstrated cooperative control of a robotic arm and satellite attitude, and simple examples of visual inspection, equipment exchange, refueling, and handling of a satellite.

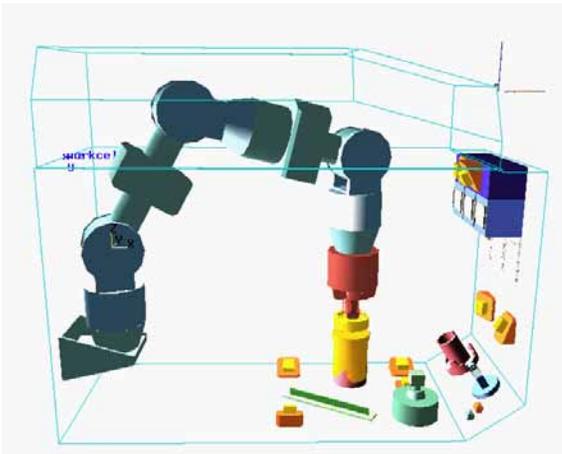


Figure 1. Model of the ROTEX manipulator

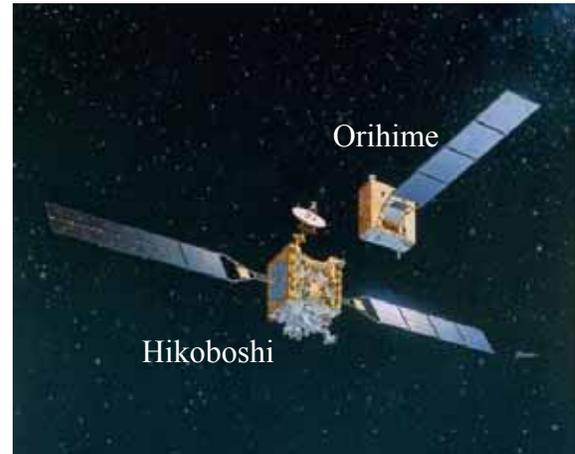


Figure 2. Artist's rendition of ETS-VII components

The state of the art in assembly robotics, implemented via experimental test beds only, is the Skyworker¹² robot at Carnegie Mellon University (see Figure 3), NASA's Robonaut¹³ (see Figure 4), and Ranger¹⁴ at the University of Maryland (see Figure 5). Skyworker is an 11 DOF robot that walks across the structure it is assembling to mate new components to the existing structure. The current prototype allows for high-level command inputs that are then parsed and implemented on-board as motion control commands. Robonaut is a collaborative effort between DARPA and NASA aimed at developing a humanoid robot capable of meeting the increasing requirements for extravehicular activity (EVA). Robonaut, composed of two dexterous arms and two five-fingered hands with teleoperational and autonomous capabilities, has already

demonstrated assembly of complicated EVA electrical connectors and delicate capabilities such as soldering. Ranger is a teleoperated robot at the University of Maryland that completes assembly, maintenance, and human EVA assistance tasks in a neutral buoyancy tank. Ranger has demonstrated robotic replacement of an Orbital Replacement Unit (ORU), complete end-to-end electrical connector mate/demate, and two-arm coordinated control.

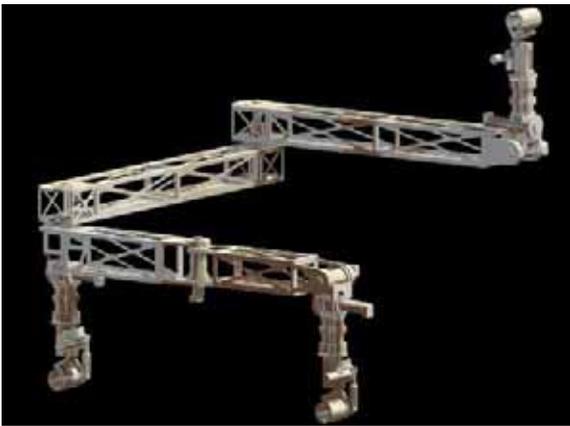


Figure 3. CMU's Skyworker prototype



Figure 4. NASA's Robonaut

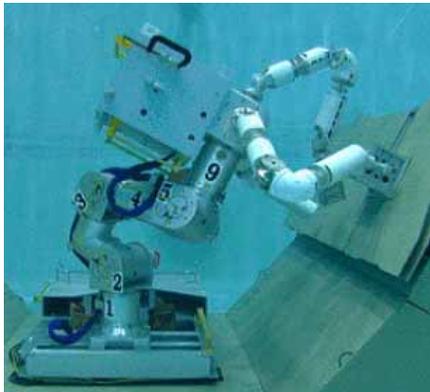


Figure 5. University of Maryland's Ranger

According to “Space Robotics Technology Assessment Report”¹⁵ published by NASA in December of 2002, the expectation for in-space assembly under nominal research efforts is: “Robots that can autonomously mate components and do fine assembly, including making connections under careful human supervision.”¹⁶ It is perceived, given the current state of the art of robots like Robonaut, that robots will possess the mechanical capabilities equivalent to a space-suited human, but barring a breakthrough in communications architectures, more aggressive expectation of space robotic assembly cannot be met for teleoperation, due to current low bandwidth/high latency communication. On the other hand, automating Robonaut and other highly dexterous robots is possible, but to make highly dexterous robots effective under autonomous operation, a system level design needs to be considered that designs the small components specifically for assembly by autonomous robotic systems. This requires significant redesign in current infrastructure, which was perceived to be financially impractical at the time of the “Space Robotics Technology Assessment Report.” However, the new NASA mandate outlines strategic allocation of funding to achieve its aggressive goal set, and as a result it now may be possible to alter the current infrastructure with the intent to simplify automated robotic assembly.

In-space inspection consists of examining space structures for anomaly detection. The robots performing the inspection can be either free-flyers or manipulators capable of performing the subtasks of moving, such as navigation and obstacle avoidance, to examine the entire exterior of the structure and anomaly detection via sensor interpretation. Currently, there are no operational space inspection systems. However, there have been on-orbit investigations of subsystems, such as AERCam Sprint¹⁰ (see

Fig. 6), which flew aboard Columbia (STS-87) in 1997. AERCam is a teleoperated EVA camera whose purpose was to merely investigate the feasibility of an autonomous, free-flying EVA camera system.

The state of the art in the area of inspection is the mini AERCam¹⁷ (see Fig. 7), Orbiter Boom Sensor System,¹⁸ OBSS (see Fig. 8), and the Supplemental Camera and Maneuvering Platform,¹⁹ SCAMP (see Fig. 9), at the University of Maryland. The mini AERCam project is a second generation version of the Sprint aimed at adding more complex capabilities while reducing the overall size of the prototype. A nanosatellite-class spherical free-flyer, the mini AERCam is only 7.5 inches in diameter and weighs a mere 10 pounds. Even though mini AERCam cutting edge hardware is not flight certified, making it far from implementable, the results obtained from orbital simulations and 5 DOF experimental test bed validations are invaluable to advancing the field of autonomous inspection agents.

The Orbiter Boom Sensor System (OBSS) is a manipulator based concept for inspection of the thermal protection surface (TPS) of the Shuttle. Consisting of a camera and a laser range sensor mounted on a 50' boom, the OBSS attaches directly to the SRMS, providing teleoperational inspection of nearly 75% of the Shuttle's TPS. In the Space Systems Laboratory at the University of Maryland, the Supplemental Camera and Maneuvering Platform (SCAMP) allows for investigations in free-flying camera applications. Operated in a neutral buoyancy tank, SCAMP provides near micro-gravity conditions for research into free-flying applications. SCAMP has demonstrated effective stereo video data interface and 3D navigation in the neutral buoyancy test bed.



Figure 6. AERCam Sprint



Figure 7. Mini AERCam

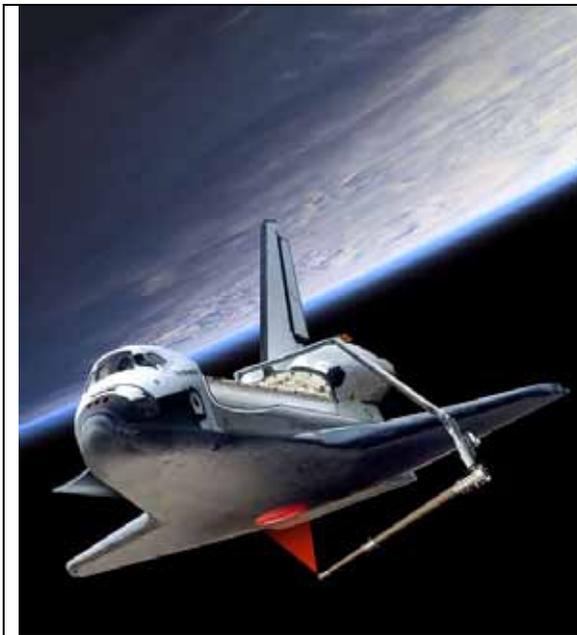


Figure 8. Artist's depiction of the Orbiter Boom Sensor System



Figure 9. The SCAMP in the neutral buoyancy tank

For in-space inspection, NASA perceives the expectation for the next decade to be summarized as “autonomous robotic inspection of some of the exterior surfaces with

sensory data filtered for potential anomaly before being stored or sent.”²⁰ With more aggressive research effort, autonomous inspection and anomaly detection of most exterior surfaces of a target are realizable.

The addition of autonomous capabilities to ISO systems is a timely issue. The new NASA mandate² sets aggressive goals for space robotics missions, and autonomous capabilities for robotic systems are vital to the achievement of these goals. Furthermore, the international community, initiated by Deutsches Zentrum für Luft- und Raumfahrt (DLR), also known as the German Aerospace Center, has been proactive in forming a community whose sole focus is on-orbit servicing (OOS). In their own words, the spirit of the OOS initiative “aims at developing a sound understanding of OOS and its issues, and at creating a dedicated community on global level”²¹ with the overall objective being “to develop a structured long-term approach of how to deal with OOS by providing high-quality information and latest insights to this new, complex and innovative field of space activities, and to initiate new collaborations.”²¹ OOS has organized two workshops, in 2002 and 2004, dedicated solely to all aspects of on-orbit servicing, and numerous sessions for on-orbit servicing in other conferences worldwide, e.g., the 8th International Symposium on Artificial Intelligence, Robotics and Automation in Space (I-sairas 2005), providing further evidence to the point that ISO systems development is a timely, global issue.

With the state of the art in ISO systems, as outlined above, reflecting few autonomous capabilities, and the absence of a mature knowledge base in free-flying robotics theory, adding autonomy to ISO systems is a monumental task as evidenced by the requests for information^{22,23} (RFI) issued from NASA in the months directly

following the issuing of the new mandate, the recent formation of the Institute for Space Robotics, and the OOS initiative. To this end, this dissertation explores strategies for on-board guidance and develops a guidance algorithm for proximity operations for autonomous ISO robotics systems in an attempt to improve the current state of the art of autonomous, in-space robotic systems.

Research Scope

For most ISO missions (assembly, repair, refueling) the target location and its orbit parameters are known a priori, with the exception being maneuvering non-cooperative targets which we will not consider in the present work. For instance, in the case of an assembly mission, the location of the construction site and the parts depot are known parameters from mission planning. Utilizing this known information, we propose a novel approach to collision free operations of the spacecraft based on artificial potential function (APF) guidance.

In order to utilize APF based guidance schemes, two prominent deficiencies inherent to this method need conquering: premature termination due to local minima in the operational space, and suboptimal performance. It is widely known that utilization of harmonic potential functions eliminates the introduction of local minima in the operational space; thus we employ harmonic functions to eliminate the introduction of local minima. In order to overcome the suboptimal performance, a sum of harmonic function primitives (simple functions such as sinks and vortices in the hydrodynamic analog) is employed to shape the velocity field of the operational space to harness the underlying dynamics of the system (i.e., orbit mechanics). Furthermore, given the location and orbit parameters of the target site, a random heuristic search algorithm can be employed to optimize the velocity field shaping for maximizing the performance gains

achieved from harnessing these underlying dynamics. This dissertation presents this novel approach for proximity operations of multiple space assets operating in relatively unstructured environments.

Dissertation Outline

This dissertation is organized as follows. Chapter 1 provides an introduction to the current trends in the space industry, details the state of the art for ISO systems in both real world implementations and test bed environments, and concludes with the research scope that precisely defines the problem and solution presented in this work. Chapter 2 explores the orbit mechanics based approaches to trajectory generation for spacecraft, and illustrates the deficiencies that eliminate these algorithms for the specified application considered in this dissertation.

In Chapter 3 the motion planning approaches from the robotics field are surveyed as alternatives to the classical orbit mechanics based approaches. From the robotics literature, the artificial potential function (APF) guidance approach is singled out as a promising solution approach for the problem considered here, and examples of APF guidance algorithms applied to the classical orbital rendezvous with and without obstacles are presented.

Chapter 4 develops the novel alterations that overcome the two predominant deficiencies of APF algorithms that would prohibit their application for ISO missions. The first part of this chapter develops an original, truly general dynamic obstacle avoidance algorithm that is a critical subsystem for the trajectory generator for ISO missions. The second part of Chapter 4 deals with the mitigation of local minima and suboptimal performance by utilization of harmonic function primitives. The chapter is

concluded with the application of the APF guidance scheme to a representative ISO mission scenario.

Eventually the spacecraft will not be considered to be a point mass and attitude considerations must be included. In Chapter 5 the classical APF framework is utilized to develop an attitude guidance algorithm that can be shown to be a subclass of controllers that can be derived using a Lyapunov analysis. Chapter 6 will present conclusions and future work.

CHAPTER 2 TRAJECTORY DETERMINATION: ORBIT MECHANICS APPROACH

The general mission scenario for all autonomous in-space operations (ISO) systems requires translating from an initial position to a goal position where the system fulfills its designed functional purpose, e.g., assembling a structure. It is assumed that the initial position of the system is either a parking orbit provided by the launch bus or an orbiting space platform (e.g., ISS), both of which imply the need for midcourse and endgame trajectory design. Midcourse trajectory determination refers to the transfer from initial location to close proximity, and endgame trajectory determination refers to the proximity maneuvering. In the sections that follow, classical orbit mechanics theory for the midcourse and endgame trajectory determination phases will be presented along with more recently developed methods based on orbit mechanics. Numerical examples illustrating the key aspects are included, and the limitations on each for autonomous ISO applications are illustrated.

Midcourse Trajectory Determination

The most general framework for midcourse trajectory determination is the n-body problem of astrodynamics (see Refs. 24-27 or any text on Astrodynamics or Celestial Mechanics). The solution of which requires $6n$ integrals, of which there are ten known,²⁴⁻²⁷ and a general closed form solution is not obtainable. However, the artificial satellites are well within the sphere of influence (Ref. 24, Chapter 7) of the primary central body, and the mass of the satellites is negligible with respect to the mass of the central body. Therefore, the trajectory of any robot in this context can be approximated effectively by

the solution of a two-body problem, assuming the position of the goal is associated with an orbiting target.

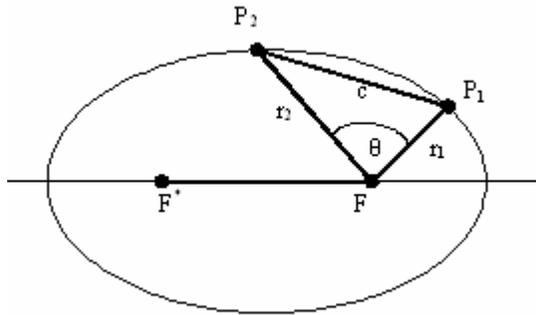


Figure 10. Schematic of Lambert's Problem

Restricting consideration to two-body problems, the determination of the orbital trajectory given two points in space and a time of flight is known as the two-body orbital two-point boundary value problem²⁴ (see Figure 10). The first strides in the solution of this boundary value problem came from Johann Heinrich Lambert^{24,25} (1728-1779). Lambert's Theorem states that the time $(t_2 - t_1)$ required to traverse an elliptical arc between specified endpoints, P_1 and P_2 , depends only on the semi-major axis a and two geometric properties of the space triangle FP_1P_2 , the chord c and the sum of the radii $r_1 + r_2$. Since Lambert arrived at this conclusion from strictly geometric means, the mathematical statement of Lambert's Theorem, Lambert's Equation, actually comes from Joseph-Louis Lagrange²⁴ (1736-1813) and is given by Eq. (1) in terms of Lagrange's variables α and β . In order to show that Eq. (1) indeed proves Lambert's Theorem, α and β can be expressed as Eq. (2) and (3), where $s = \frac{1}{2}(r_1 + r_2 + c)$ is the semi-

perimeter of the space triangle. Hence, the transfer time is a function of the semi-major axis, the radii sum $r_1 + r_2$, and the chord c . Obtaining solutions of Eq. (1) became known as Lambert's problem. Historically its solution was essential in determining the orbits of celestial bodies from observation, and today it is applicable for artificial satellite intercept and rendezvous problems.

$$\sqrt{\mu}(t_2 - t_1) = a^{\frac{3}{2}} [\alpha - \beta - (\sin \alpha - \sin \beta)] \quad (1)$$

$$\sin \frac{\alpha}{2} = \left(\frac{s}{2a} \right)^{\frac{1}{2}} \quad (2)$$

$$\sin \frac{\beta}{2} = \left(\frac{s-c}{2a} \right)^{\frac{1}{2}} \quad (3)$$

It is important to note that the solution of Lambert's Problem only defines the geometry of the intercept problem between points P_1 and P_2 . To effectively execute the trajectory specified by the solution geometry, the terminal velocity vectors need to be calculated in order to obtain the necessary guidance command to initiate the transfer. Therefore, determination of the maneuver command is the essential quantity.

Lambert's Problem

The determination of an orbit from a specified transfer time $(t_2 - t_1)$ connecting two position vectors, \underline{r}_1 and \underline{r}_2 , is called Lambert's Problem²⁴⁻²⁷ (see Figure 10). A variety of techniques for the solution of Lambert's Problem have been developed over the years^{24,28,29} with each being classified by the particular form of the transfer-time equation and the independent variable chosen in an iterative solution technique. Classically two transfer-time equations²⁴ were used, one from Lagrange's equations used in his proof of Lambert's Theorem and the other derived from Gauss' work. Each of these methods has

their advantages and disadvantages, and a third transfer-time equation that utilizes these advantages while mitigating the disadvantages can be developed as in Battin.²⁴ Battin's combined equations technique is utilized for the present work. For a complete history and development of Lambert's Problem and the two-body orbital boundary value problem see Battin.²⁴

Before attempting to solve Lambert's Problem it is instructive to consider inherent properties of Eq. (1) that will influence the techniques employed for its solution.

Lambert's Equation is:

1. Transcendental equation
2. Double valued function of a (allows for conjugate orbits with same semi-major axis)
3. Derivative with respect to a is infinite for the minimum energy orbit $a = \frac{s}{2}$
(the minimum energy orbit corresponds to the smallest possible value of semimajor axis a)
4. Four possible solution for each case due to quadrant ambiguities in α and β

Property 1 requires the use of an iterative solution approach to find the semi-major axis a . From properties 2 and 3 it is obvious that the semi-major axis is not amenable to iterative solution techniques. Furthermore, Property 4 mandates the use of a method for selecting the unique solution from the multiple solution possibilities. These difficulties can be mitigated by utilizing a transformation of Eq. (1) by Battin²⁴ that replaces the semi-major axis with an independent variable that produces monotonically decreasing and single valued solution curves, which is more appropriate for an iterative solution approach. Furthermore, utilization of Prussing's geometric interpretation²⁵ provides the framework necessary to determine a unique solution.

Battin's combined equations technique²⁴ will now be presented. This procedure first transforms the independent variable in Lagrange's Equation to one with properties

amenable to iterative solution techniques. Then, utilizing another form of the transfer-time equation attributed to Carl Friedrich Gauss (1777-1855), Battin improves the computational efficiency by reducing the number of hypergeometric function evaluations from two to one, however the independent variable in this transfer-time equation lacks the attractive properties obtained previously. By developing a relationship between the two independent variables, Battin combines the two transfer-time equations in order to obtain the combined equation form utilized in the current work.

Battin's combined equations transfer-time equation is given by Eqs. (4a) through (4f), where $F[\alpha, \beta; \gamma; x]$ is the hypergeometric function. This form of the transfer-time equation is devoid of the difficulties encountered from using the semi-major axis as the independent variable, which renders it readily adaptable to an iterative solution method, such as Newton's Method. The parameter λ is a function of the known geometry and y is an intermediate variable expressible in terms of x as Eq. (4c). Furthermore, η is essentially a function of x only since y can be given by Eq. (4c), and S_1 is a function of x only, which implies that $Q = Q(x)$ only. Therefore Eq. (4a) is a function of the independent variable x only. It is also important to point out that the solution obtained for x is unique, a property not apparent by inspection. The reason for the uniqueness can be attributed to the use of variable x and y , which can be given by alternate expressions $x = \cos \frac{1}{2} \alpha$ and $y = \cos \frac{1}{2} \beta$. Therefore, ambiguities arising from the trigonometric functions involving α and β are avoided.

$$\sqrt{\frac{\mu}{a_m^3}} (t_2 - t_1) = \eta^3 Q + 4\lambda \eta \quad (4a)$$

$$\lambda = \frac{\sqrt{r_1 r_2}}{s} \cos \frac{\theta}{2} \quad (4b)$$

$$y = \sqrt{1 - \lambda^2 (1 - x^2)} \quad (4c)$$

$$\eta = y - \lambda x \quad (4d)$$

$$S_1 = \frac{1}{2} (1 - \lambda - \eta x) \quad (4e)$$

$$Q = \frac{4}{3} F \left[3, 1; \frac{5}{2}; S_1 \right] \quad (4f)$$

Maneuver Command Determination

In order to calculate the maneuver command the solution x from the Battin algorithm must be converted to geometric properties a , α , and β . The equations utilized for this purpose are given by Eqs. (2), (3), and (5), where $a_m = \frac{s}{2}$ is the value of the semi-major axis for the minimum energy transfer between P_1 and P_2 . From the inverse trigonometric operations the conversion always yields the principal values α_0 and β_0 , characterized by $0 \leq \beta_0 \leq \alpha_0 \leq \pi$. However, four solutions for α and β are possible arising from quadrant ambiguities from inverse trigonometric operations. In order to obtain the correct solution from the four possibilities, Prussing geometric interpretation²⁵ is employed.

$$x^2 = 1 - \frac{a_m}{a} \quad (5)$$

Prussing geometric interpretation involves transforming the geometry of Figure 10 to a rectilinear ellipse, thereby introducing geometric significance to the angles α and β . The results of this procedure provides a framework for the logical determination of the

appropriate values of α and β from the principle values α_0 and β_0 . The assignment procedure is given below by Eqs. (6) and (7), where $(t_2 - t_1)$ is the transfer time, t_m is the transfer time on the minimum energy transfer, and θ is the transfer angle.

$$\alpha = \begin{cases} \alpha_0, & (t_2 - t_1) \leq t_m \\ 2\pi - \alpha_0, & (t_2 - t_1) > t_m \end{cases} \quad (6)$$

$$\beta = \begin{cases} \beta_0, & 0 \leq \theta < \pi \\ -\beta_0, & \pi \leq \theta \leq 2\pi \end{cases} \quad (7)$$

Once α and β have been obtained from the above procedure the terminal velocity vector \underline{v}_1 at \underline{r}_1 (and \underline{v}_2 at \underline{r}_2 for a rendezvous) is calculated. To do so, introduce a skewed set of unit vectors, defined in Eqs. (8a) through (8c), where $\hat{\underline{u}}_i$, $i = 1, 2$, are unit vectors in the direction of \underline{r}_i , and $\hat{\underline{u}}_c$ is a unit vector parallel to the chord. It can be shown²⁵ that in this coordinate bases \underline{v}_1 has the elegantly simple expression given by Eq. (9), with A and B given by Eqs. (10) and (11). Therefore, the final objective of obtaining the maneuver command necessary to initiate the trajectory between the start and goal configuration can be calculated by taking the vector difference of the satellite's current velocity with \underline{v}_1 .

$$\hat{\underline{u}}_1 = \frac{\underline{r}_1}{r_1} \quad (8a)$$

$$\hat{\underline{u}}_2 = \frac{\underline{r}_2}{r_2} \quad (8b)$$

$$\hat{\underline{u}}_c = \frac{(\underline{r}_2 - \underline{r}_1)}{c} \quad (8c)$$

$$\underline{v}_1 = (B + A)\hat{\underline{u}}_c + (B - A)\hat{\underline{u}}_1 \quad (9)$$

$$A = \left(\frac{\mu}{4a} \right)^{\frac{1}{2}} \cot \frac{\alpha}{2} \quad (10)$$

$$B = \left(\frac{\mu}{4a} \right)^{\frac{1}{2}} \cot \frac{\beta}{2} \quad (11)$$

Numerical Example: Midcourse

To illustrate the midcourse trajectory determination and maneuver command algorithms, a numerical example is presented. The problem is as follows: given position vectors $\underline{r}_1 = [9567 \ 0 \ 0]^T$ and $\underline{r}_2 = [-37394 \ 10783 \ 0]^T$ (units are kilometers) coordinatized in the Perifocal (PQW) frame (x-axis pointed in the direction of the eccentricity vector) and a time of flight equal to four hours, determine the solution trajectory and the maneuver command. The initial position was chosen to be the periapse which allows for simple analytical verification of the terminal velocity vector produced by the midcourse maneuver command algorithm.

Using algorithms based on the theory of the previous sections and developed by the author the solution is obtained. The iterative solution for x in Lambert's Problem using Newton's method required 19 iterations to converge with a relative error less than 10^{-8} .

The terminal velocity at P_1 was found as $\underline{v}_1 = [-8.73 \times 10^{-10} \ 8.228 \ 0]^T$ kilometers per second. The CPU runtime on a dual 2.4 GHz Xeon processor workstation for the midcourse maneuver command determination algorithm was approximately 0.15 ± 0.05 seconds. The terminal velocity \underline{v}_1 produces by the algorithm can be verified by checking

against the analytical expression for the velocity at periapse passage $v_p = \sqrt{\frac{\mu(1+e)}{a(1-e)}}$.

Performing this calculation produces the analytical terminal velocity vector as

$\underline{v}_p = [0 \quad 8.228 \quad 0]^T$ km/s, which shows good agreement with the results from the algorithm. Figure 11 depicts the transfer trajectory, obtained from a Keplerian motion simulator using the initial conditions given by the position and terminal velocity vector solutions.

It is important to note a caveat for the numerical example above. The position vectors were given in the Perifocal (PQW) frame, which is an orbit based coordinate frame. However, it is unlikely that the position information will be known in the PQW frame, instead position information is more likely to be obtained in the Earth-Center Inertial (ECI) frame or the Local-Vertical Local-Horizontal (LVLH) frame. The implication of this is, unlike the example above, the initial value of the true anomaly is unknown and cannot be determined by inspection from the position vectors. For instance, if the position vectors in the example above were given in the ECI frame (with $\Omega = 10^\circ, i = 28.5^\circ, \omega = 45^\circ$) the position vector would be $\underline{r}_1 = [5629.4 \quad -7694.7 \quad 792.9]^T$ and $\underline{r}_2 = [-14080.1 \quad 35350.3 \quad -8166.2]^T$ km, which does not exhibit any evidence that the true anomaly is zero at P_1 . Therefore, in most cases only the difference in true anomaly is known from geometry and the elegant calculations used to verify the terminal velocity vector here are not applicable.

Endgame Trajectory Determination

Endgame conditions for this work will be defined such that the dynamics of relative motion between two bodies in orbit can be accurately approximated by linear orbit theory. The limit on the applicability of linear orbit theory is dependent on both the desired level of fidelity and the class of orbit operation, e.g., Low-Earth orbit (LEO) or Geostationary orbit (GEO). For an in-depth analysis on the applicability of linear orbit

theory see Refs. 30 and 31. For the analysis that follows it is assumed that endgame conditions are met.

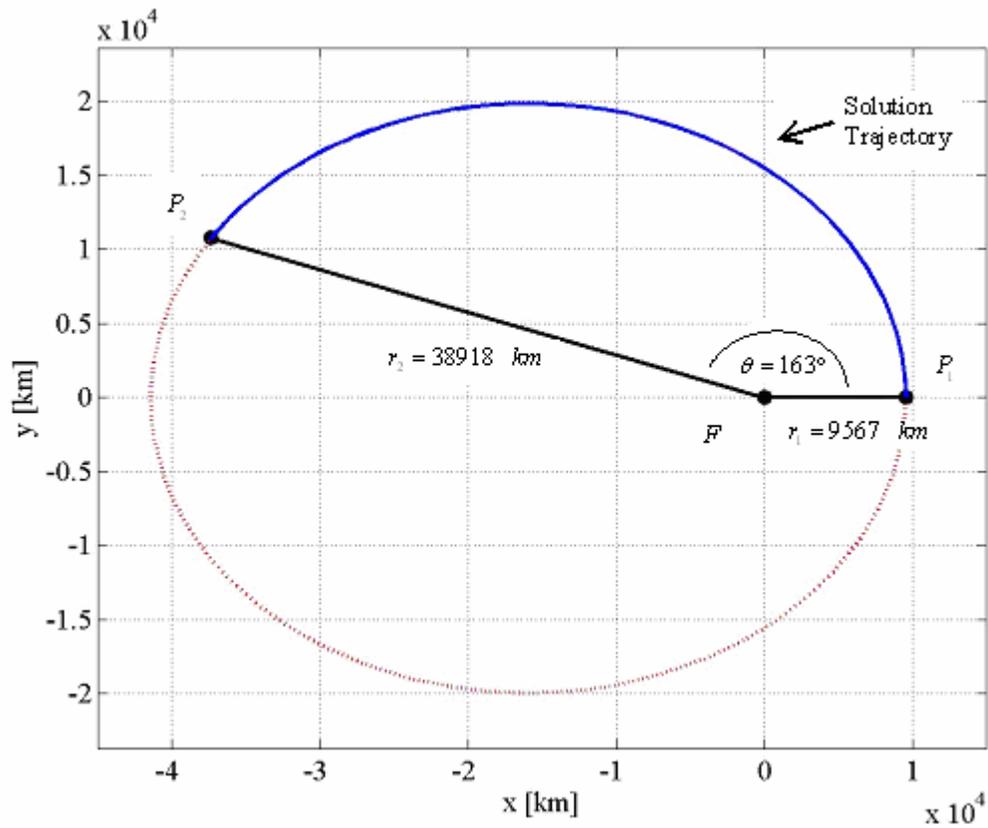


Figure 11. Transfer trajectory from midcourse trajectory determination algorithm

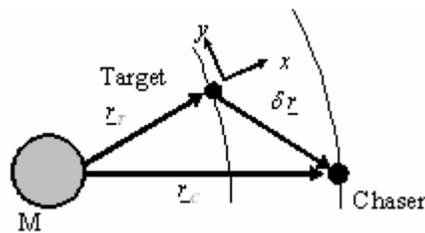


Figure 12. Depiction of orbital rendezvous

The basic scenario for the development of linear orbit theory is depicted in Figure 12, where both the Target and Chaser are assumed to be in two-body motion about the

central body, denoted by M. The most familiar equations from linear orbit theory and the ones employed in the present work are the Clohessy-Wiltshire (CW) equations,³² originally developed for the first orbital rendezvous flight demonstration programs. The CW equations are given in state transition matrix form by Eqs. (12) - (14), where x, y, z (z out of the page) are the coordinates of the Chaser with respect to the Target in the local frame depicted in Figure 12 and n is the orbital mean motion of the Target. It is vitally important to understand the implicit assumption of the CW equations that ultimately limits their applicability and fidelity. The reference (Target) orbit is assumed to be circular. Therefore, for any scenario where the reference orbit is not circular errors in addition to the ones from linearization are introduced, thus, further degrading the fidelity of the model.

$$\delta \underline{r}(t) = \Phi(t) \delta \underline{r}(0) \quad (12)$$

$$\delta \underline{r}(t) = [x(t) \quad y(t) \quad z(t) \quad \dot{x}(t) \quad \dot{y}(t) \quad \dot{z}(t)]^T \quad (13)$$

$$\Phi(t) = \begin{bmatrix} 4 - 3 \cos nt & 0 & 0 & \frac{s}{n} & \frac{2}{n}(1 - \cos nt) & 0 \\ 6(\sin nt - nt) & 1 & 0 & -\frac{2}{n}(1 - \cos nt) & \frac{4 \sin nt - 3nt}{n} & 0 \\ 0 & 0 & \cos nt & 0 & 0 & \frac{s}{n} \\ 3n \sin nt & 0 & 0 & \cos nt & 2 \sin nt & 0 \\ -6n(1 - \cos nt) & 0 & 0 & -2 \sin nt & 4 \cos nt - 3 & 0 \\ 0 & 0 & -n \sin nt & 0 & 0 & \cos nt \end{bmatrix} \quad (14)$$

CW Maneuver Command Determination

The primary advantage of the CW equations is the computational simplicity of determining the transfer velocity necessary for performing a maneuver. The objective of the maneuver is to bring the Chaser with initial conditions $\delta \underline{r}(0)$ and $\delta \dot{\underline{r}}(0)$ to the origin

of the CW frame (depicted in Figure 12) given a time of flight t^* . Partitioning the state transition matrix Φ into four 3 x 3 matrices as given by Eq. (15), the top partition leads to the equation for the relative position $\delta \underline{r}(t) = \mathbf{M}(t)\delta \underline{r}(0) + \mathbf{N}(t)\delta \dot{\underline{r}}(0)$. Substitute $t = t^*$ and $\delta \underline{r}(t^*) = \underline{0}$, giving the transfer velocity $\delta \dot{\underline{r}}_{0+}$ the expression shown by Eq. (16).

$$\Phi(t) = \begin{bmatrix} \mathbf{M}(t) & \mathbf{N}(t) \\ \mathbf{S}(t) & \mathbf{T}(t) \end{bmatrix} \quad (15)$$

$$\delta \dot{\underline{r}}_{0+} = -\mathbf{N}(t^*)^{-1} \mathbf{M}(t^*) \delta \underline{r}(0) \quad (16)$$

Numerical Example: Endgame

Continuing the maneuver begun in the numerical example of the midcourse section, the problem scenario is as follows. The robot completes the midcourse trajectory to find the goal in a circular orbit with radius $r_G = 38916.7$ km. The discrepancy in the positions of the goal and robot at the end of the midcourse maneuver can be viewed as errors incurred from unmodeled perturbations such as drag, oblateness, or solar radiation pressure effects. The objective of this example is to use the endgame algorithm outlined in the previous section to generate the maneuver command that will bring the Chaser to the Target's position with a time of flight arbitrarily specified as one-quarter of period of the Target orbit. All vector quantities are coordinatized in an R-bar V-bar frame attached to the Target, with x in the direction of R-bar and y in the direction of V-bar.

The endgame maneuver command generated by the author's algorithm is given by $\Delta \underline{v} = [0.0001 \quad 1.4028 \quad 0]^T$ km/s with a CPU time less than a few milliseconds on the workstation used previously. The maneuver was simulated, as depicted by Figure 13, and the absolute value of the errors in position at the end of the maneuver were obtained as

$|\varepsilon| = [1.059 \times 10^{-9} \quad 1.206 \times 10^{-9} \quad 0]^T$, which illustrates the convergence of the robot to the goal position in the specified time.

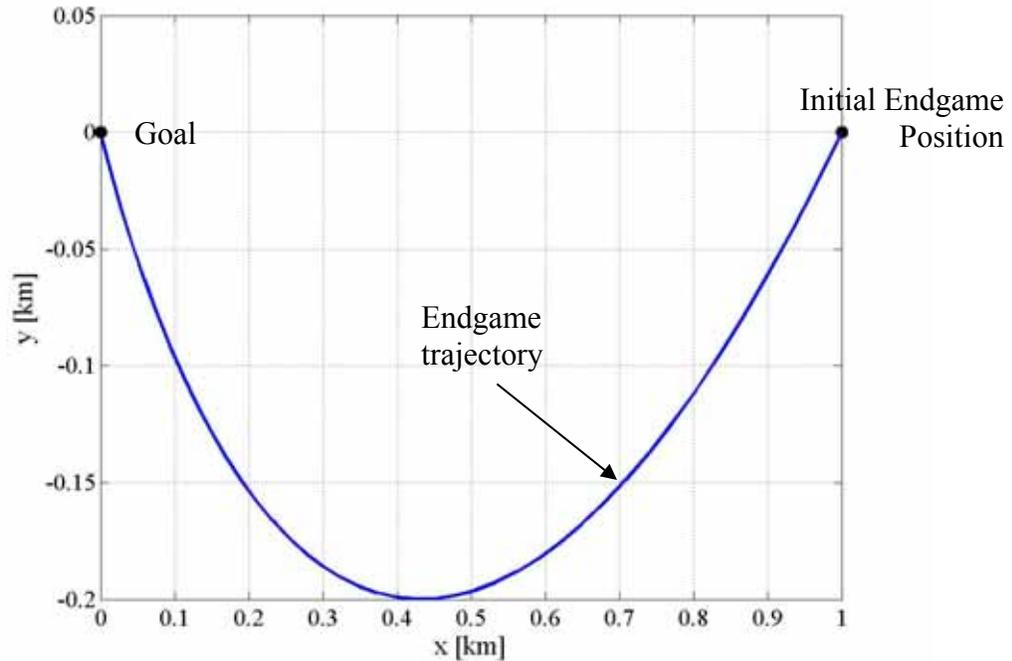


Figure 13. Endgame maneuver trajectory in the xy-plane (R-bar V-bar plane)

Alternative Orbit Mechanics Based Methods

Having illustrated the classical methods of orbit mechanics for trajectory determination, we turn to a more recent algorithm³³ based on an adaptation of glideslope guidance that has been utilized for rendezvous and proximity operations for the space shuttle and other vehicles with astronauts in the loop. To date this is the most widely used approach from a real world implementation perspective.

The glideslope approach utilizes the linearized equations of motion Eqs. (12) - (14) but completes the maneuver with N thruster firings. The algorithm divides the range-to-go into N equally spaced segments along the chord connecting the initial and final

relative positions. At equally spaced time intervals an impulse is calculated using the CW equations that drive the current position to, the next intermediate position r_m . An example of the algorithm is shown in Figure 14.

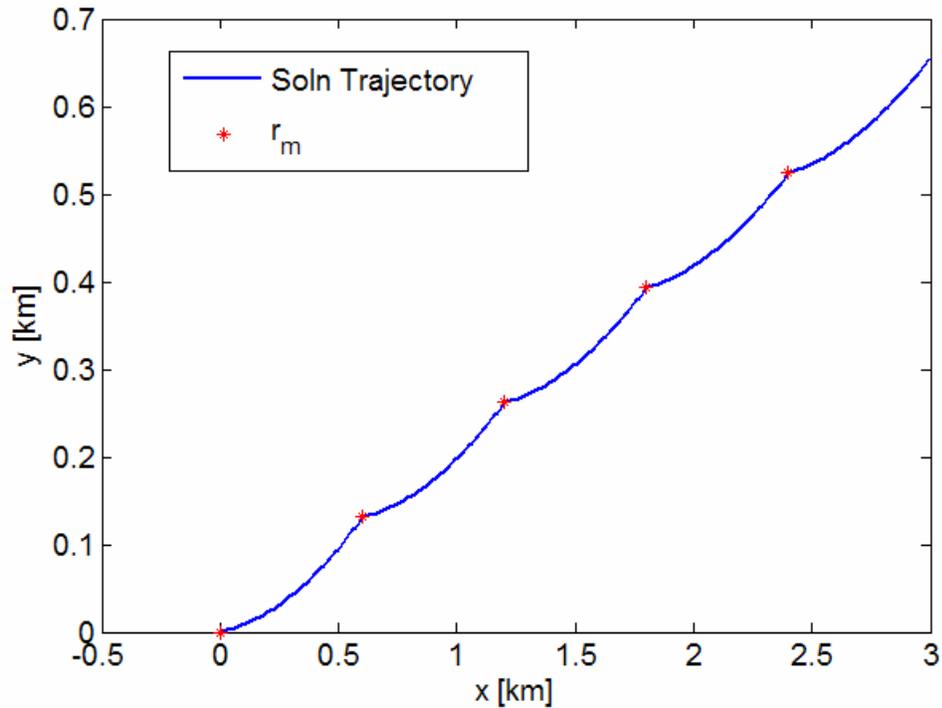


Figure 14. Glideslope solution for 5 impulse rendezvous.

The computational simplicity of the glideslope algorithm lends itself readily to application to ISO mission. However, this computational simplicity is sacrificed for situations where obstacle avoidance is necessary. In order to use a glideslope approach with obstacle avoidance, a chord must be drawn between the centers of mass of all the spacecraft in the operating environment. Therefore, $(n-1)!$ chords, each with N impulse to be calculated are necessary. For instant for a five spacecraft mission, requires $24N$

online impulse calculations! For this reason alone the glideslope approach is an unacceptable alternative for the ISO missions considered in this dissertation.

Conclusions

The existence of two trajectory determination algorithms, one for midcourse and one for end-game, implies that a multi-tier architecture is necessary. Therefore, not only are two algorithms necessary but the added complexity of switching algorithms to determine when the linear control is valid is introduced. To mitigate a multi-tier control structure, the midcourse maneuver commands could be utilized for both midcourse and endgame scenarios. However, the midcourse maneuver command determination is a computationally intensive algorithm that requires an iterative solution, which does not lend itself to on-orbit implementation due to the limited computational power that exists for current space systems. Another alternative is to modify the linear orbit theory in order to expand its range of applicability to midcourse trajectory determination as well. This subject has received intense research,³⁴⁻³⁷ including investigations by the author, with minimal achievements in expanding the range of applicability.

An additional disadvantage of classical orbit mechanics theory was obvious to Stern and Fowler³⁰ twenty years ago; classical orbit mechanics theory does not admit the possibility of the incorporation of physical trajectory constraints, introduced by either goal structure or obstacles in the path. Since the ability to dynamically incorporate obstacles and trajectory constraints is vital for all autonomous ISO systems, and given the additional disadvantages mentioned above, other sources of trajectory planning theory for autonomous ISO systems must be explored. Chapter 3 will survey the robotics literature for motion planning algorithms and will motivate and develop artificial potential function guidance algorithms for autonomous ISO applications.

CHAPTER 3 ARTIFICIAL POTENTIAL FUNCTION POSITION GUIDANCE

In the preceding chapter it was demonstrated that trajectory determination for autonomous in-space operation (ISO) systems cannot be easily handled by classical orbit mechanics theory. A logical avenue to explore for trajectory determination is the robotics literature on robot motion planning (MP). A brief outline of the robot MP problem is discussed in the first section of this chapter with an emphasis on motion planning approaches. Given that space-born robotics applications require motion planning subject to properties distinct from classical robot MP problems, for instance the difference in local operator mentioned in Chapter 1, only some of the common motion planning approaches are feasible for in-space applications. Based on the unique needs of in-space applications, a potential field approach to MP is perceived to be most appropriate for autonomous ISO systems, and this chapter concludes with a presentation of classical artificial potential field (APF) guidance algorithm for ISO applications.

Robot Motion Planning

In order to precisely define the robot MP problem, it is first necessary to define two terms, configuration and configuration space (Cspace). A *configuration* of a robot is defined as the set of independent parameters, degrees of freedom (DOF), necessary to specify the location of every point of that robot. Cspace, a concept introduced in an influential paper by Lozano-Perez and Wesley³⁸ in 1979, is defined as the set of all possible configurations of a robot, i.e., the Cspace represents all possible motions of a robot. Essentially all MP problems are equivalent when cast in the Cspace, and can be

stated as follows. Find a connected sequence of point in the Cspace between an initial configuration and goal configuration. Configurations that result in collision with the environment, known as configuration obstacles,³⁹ must be computed, and the connected sequence of points must not contain any points associated with configuration obstacles. Computation of the configuration obstacles is difficult^{39,40} and the dimension of the Cspace is equivalent to the DOF of the system, both of which makes motion planning challenging for robots with multiple degrees of freedom.

Classically the MP problem has been broken down into the steps below.

Concentration on the approaches in the literature for Steps 3 and 4 are the focus here for determining an applicable approach for ISO systems. For a complete investigation on methodologies for the steps in the MP problem see the excellent surveys in Refs. 39-43.

1. Parameterize the configuration of the robot(s)
2. Choose a representation scheme for robot(s) and the environment
3. Select a motion planning approach
4. Select a search method to find feasible path through environment
5. Optimize the solution path based on various constraints (e.g., smoothness)

Citing the landmark survey by Hwang and Ahuja,³⁹ the most common motion planning approaches found in the literature are; cell decomposition, mathematical programming, skeletons, and potential field methods. Cell decomposition techniques involve computation of the entire Cspace and all configuration obstacles in addition to requiring large memory allocations. Therefore, given that on-orbit computational power is limited, this approach is not feasible for ISO systems. Mathematical programming methods represent the requirements for obstacle avoidance as a set of inequalities on the configuration parameters. The motion planning problem is then formulated as an optimization problem that finds a solution path between start and goal configurations

based on minimization of a scalar constraint. This motion planning approach is essentially a nonlinear optimization problem subject to multiple inequality constraints, which generally requires a numerical method for its solution. Again this process is computationally intensive and therefore does not lend itself to on-orbit implementation.

Skeletons, otherwise known as roadmap or highway approaches, contract the environment into a network of 1-D lines and the motion planning problem becomes a graph search problem restricted to the network. Common examples of the skeleton method are Voronoi diagrams and subgoal networks. The skeleton methods have low memory requirements, and if an accurate representation of the environment and obstacles is known a priori, this method is feasible for ISO applications. In fact a Voronoi diagram approach is currently being investigated for the AERCam project.⁴⁴

Potential field methods for motion planning are the simplest, computationally speaking, of all the motion planning approaches presented here. This method creates a scalar function called the potential that has the following properties.

- Potential is at a minimum when the robot is at the goal configuration
- Potential is high on the surface of obstacles
- Everywhere in the environment away from the obstacles, the potential slopes toward the goal configuration

Typically the potential field method is combined with a specified local operator, such as “go-straight” and the path taken to the goal is recorded. Potential field methods also allow for simple consideration of non-point robots by evaluating the potential at representative points on the robot. Additionally, no explicit Cspace computation is necessary. It is the opinion of the author that the utilization of an artificial potential field

motion planning approach with the local operator being the dynamics of motion is the most effective motion planning approach for ISO applications.

Artificial Potential Function Position Guidance

Artificial potential function (APF) guidance for satellite systems⁴⁵⁻⁴⁹ has appeared in the literature recently. However, the application of real-time proximity operations of an autonomous ISO system has yet to be investigated. In this section the robot will be considered to be a point-mass moving in three dimensions. This restriction will be removed for attitude considerations in Chapter 5. The objective of the APF position guidance algorithm is to drive the robot to the origin of the state space through a series of impulsive maneuvers.

It is instructive at this point to state the equation of motion of the dynamical system considered in this section. Since this section is focused on the relative motion between a robot and target in a two-body orbit, the motion of each is governed by Eq. (17), where $\underline{g}(\underline{r})$ is a vector function describing the gravitational field and \underline{a}_D is the disturbing accelerations due to perturbations such as atmospheric drag or oblateness. It will be assumed that the perturbations are zero and the both satellites are in an inverse gravitational field given by $\underline{g}(\underline{r}) = \frac{-\mu}{r^3}\underline{r}$ for the present analysis. Therefore, the relative motion between the robot and target $\delta\underline{r} = \underline{r}_R - \underline{r}_T$ is given by the vector difference of the governing equation of the robot and the target, as given by Eq. (18). However, these are the equations of motion in a Newtonian reference frame. It is preferable to transform these equations into a local vertical local horizontal (LVLH) frame attached to the target. This is accomplished by the classical transformation given in Eq. (19) where primes

denote differentiation with respect to the local coordinate frame and $\underline{\omega}$ and $\underline{\omega}'$ are the angular velocity and acceleration of the local frame, respectively. Thus, the equations of motion for the relative motion of the robot and target are given by Eq. (20). Furthermore, it is obvious that the origin of the LVLH frame $\delta \underline{r} = \underline{0}$ is indeed an equilibrium point of the dynamical system.

$$\ddot{\underline{r}} = \underline{g}(\underline{r}) + \underline{a}_D \quad (17)$$

$$\delta \ddot{\underline{r}} = \ddot{\underline{r}}_R - \ddot{\underline{r}}_T = \underline{g}(\underline{r}_R) - \underline{g}(\underline{r}_T) \quad (18)$$

$$\delta \ddot{\underline{r}} = \delta \underline{r}'' + 2\underline{\omega} \times \delta \underline{r}' + \underline{\omega}' \times \delta \underline{r} + \underline{\omega} \times \underline{\omega} \times \delta \underline{r} \quad (19)$$

$$\delta \underline{r}'' = \frac{-\mu}{r_R^3} \underline{r}_R + \frac{\mu}{r_T^3} \underline{r}_T - \left(2\underline{\omega} \times \delta \underline{r}' + \underline{\omega}' \times \delta \underline{r} + \underline{\omega} \times \underline{\omega} \times \delta \underline{r} \right) \quad (20)$$

The general theory for APF guidance is based on the 2nd method of Lyapunov⁵⁰ which can be described as follows: Consider the dynamical system $\dot{\underline{x}} = f(\underline{x}, t)$ with $\underline{x} \in \mathfrak{R}_n$. An equilibrium point of the system \underline{x}_e is such that $f(\underline{x}_e, t) = \underline{0}$. For simplicity it will be assumed that through a change of coordinates any equilibrium point can be shifted to the origin, and therefore, Lyapunov's theorem will be presented for equilibrium states at the origin only.

Essentially, Lyapunov's 2nd Method is a means of assessing the stability of the equilibrium points of the dynamical system.⁵⁰ The theorem states that if there is a scalar function $V(\underline{x}, t)$ such that Eqs. (21) - (23) are satisfied, where $\dot{V}\Big|_{\dot{\underline{x}}=f(\underline{x}, t)}$ is the derivative evaluated along the solution trajectories, then the origin of the state space is a globally attractive point.

$$V(\underline{x}, t) > 0 \quad \forall \underline{x} \neq \underline{0} \quad (21)$$

$$V(\underline{x}, t) \rightarrow \infty \text{ as } \|\underline{x}\| \rightarrow \infty \quad (22)$$

$$\dot{V}\Big|_{\dot{\underline{x}}=f(\underline{x}, t)} < 0 \quad \forall \underline{x} \neq \underline{0} \quad (23)$$

The first two conditions are satisfied by choosing V to be an appropriate positive definite function of \underline{x} (it will be assumed for now that the potential function is not an explicit function of time). The third condition in Lyapunov's 2nd theorem is enforced by an appropriate choice of control action.⁵¹ The total derivative of the potential function is given by $\dot{V} = (\nabla_{\underline{x}} V)^T \dot{\underline{x}}$, and the control is defined by Eq. (24) where δ is such that the velocity after a control impulse $\dot{\underline{x}}_{0+}$ is directly opposite to the local gradient of the potential $\dot{\underline{x}}_{0+} = -k \nabla_{\underline{x}} V$. This proper⁵¹ control constrains the motion of the robot after an impulse to be tangent to the local gradient of the potential. Defining the control in this manner ensures the derivative of the potential after a control impulse is negative definite, as given by Eq. (25), and therefore Eq. (23) is satisfied. Thus the origin is a globally attractive point under the control action specified.

$$Control = \begin{cases} 0, & \dot{V} < 0 \\ \delta, & \dot{V} \geq 0 \end{cases} \quad (24)$$

$$V'(\underline{x}) = -k (\nabla_{\underline{x}} V)^T (\nabla_{\underline{x}} V) < 0 \quad (25)$$

A pseudo-code for the APF guidance algorithm is given below, and an example of APF guidance for an unconstrained (no obstacles) rendezvous is given in the next section to demonstrate the implementation of the algorithm.

Algorithm

1. Obtain the current relative position and velocity between robot and target
2. Check the switching condition $\dot{V} = (\nabla_{\underline{x}} V)^T \dot{\underline{x}}$

3. If $\dot{V} \geq 0$, give an impulsive velocity input such that $\dot{\underline{x}}_{0+} = -k\nabla_{\underline{x}}V$
4. If $\dot{V} < 0$, coast under the influence of the dynamics of motion
5. Sample position and velocities at desired frequencies and apply control based on switching conditions
6. Repeat until terminal error bound is satisfied

Example: Spacecraft Unconstrained Rendezvous

Define the potential function $V(\underline{x}) = \frac{1}{2}\underline{x}^T \mathbf{P}\underline{x}$, where \mathbf{P} is assumed to be a positive

definite, diagonal matrix for simplicity and $\underline{x} = \delta\underline{r} = [x \ y \ z]^T$ are the coordinates of the

relative position in the LVLH frame. This gives $\nabla_{\underline{x}}V = \mathbf{P}\underline{x}$ and the total derivative

$\dot{V} = (\mathbf{P}\underline{x})^T \dot{\underline{x}}$. For the rendezvous, the relative velocities of the spacecraft $\dot{\underline{x}}$ must also

converge to zero. The convergence is actually guaranteed by the choice of control

presented above. Consider the norm of the velocity after an impulse given by Eq. (26),

thus $\|\dot{\underline{x}}\| \rightarrow 0$ as $\|\underline{x}\| \rightarrow 0$ and the rendezvous is guaranteed.

$$\|\dot{\underline{x}}\| = k [\mathbf{P}\underline{x}]^{\frac{1}{2}} \quad (26)$$

The numerical example considers the rendezvous between a robot in a circular orbit with radius $r_R = 6705$ km and a target satellite also in a circular orbit with radius

$r_T = 6700$. The robot leads the target radius vector by a phase of 0.001 degrees. The

APF guidance parameters used were $k = 1$ and $\mathbf{P} = \mathbf{I}$, where \mathbf{I} equal to the 3 x 3 identity matrix. Figure 15 depicts the trajectory taken by the robot in the xy-plane, where x and y are the coordinates of the LVLH frame in the orbit plane with x in the radial direction.

The sharp corners on the trajectory mark where an impulsive control was applied. Figure

16 shows the impulses in the x and y directions, respectively, applied by the APF

guidance algorithm as a function of time. In all a total $\Delta v = 27.10$ m/s was necessary to

bring the robot to within a meter of the target. The duration of the maneuver was 86.2 minutes. It is important to note that the APF guidance algorithm ensures asymptotic convergence to the origin, thus a majority of the duration of the maneuver came from closing the final meters.

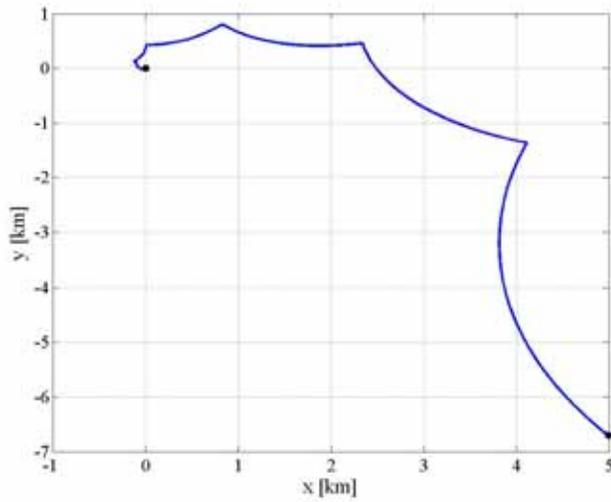


Figure 15. Planar rendezvous in the CW frame under APF guidance

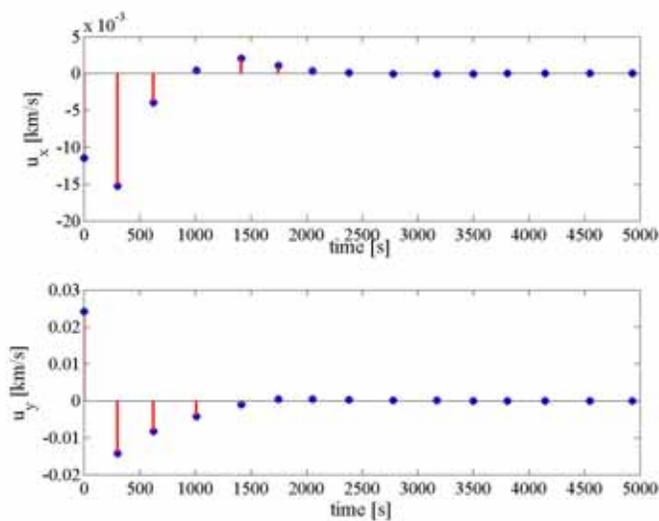


Figure 16. Control efforts in x and y directions for the rendezvous

Obstructions

In proximity operating conditions, instances exist where certain regions of the state space need to be avoided; for example, solid obstructions in the flight path. Therefore, the potential function needs to be altered in order to ensure the robot avoids these forbidden areas. For the current work, all obstructions will be approximated by spheres (or circles in the planar case). This is a reasonable assumption for two reasons. Firstly, for now the objective of the guidance algorithm is to provide safe trajectories between a start to goal location, not to generate trajectories for close proximity fly-bys, yet. Therefore, an approximate map of objects is acceptable. Secondly, the big picture for autonomous ISO systems is perceived to start with a rough estimate of the working environment, with perhaps this estimated environment approximating objects initially as spheres and updating the map through sensor data obtained from fly-bys.

The obstructions will be represented as regions of high artificial potential in the potential field in the form of Gaussian function as in Eq. (27). The parameters ψ_k and σ_k define the width and skewness of the Gaussian function whose center is located at $\tilde{\mathbf{x}}_k$, and the matrix \mathbf{M} defines the shape (since all obstacles are assumed spherical $\mathbf{M} = \mathbf{I}$). Representing obstacles by Gaussian functions ensures that controls remain bounded, no singularities are introduced into the potential field, and there are no local minima. It is instructive to note here the units of the width and skewness are $[L^2]$ which will be important due to the fact that all simulations are run in nondimensional units for completeness.

$$\Psi_k(\underline{\mathbf{x}}, \tilde{\mathbf{x}}_k) = \psi_k \exp\left\{-\sigma_k^{-1}(\underline{\mathbf{x}} - \tilde{\mathbf{x}}_k)^T \mathbf{M}(\underline{\mathbf{x}} - \tilde{\mathbf{x}}_k)\right\} \quad (27)$$

Once the APF for the obstacles has been selected, the parameters of the obstacle APF must be determined such that the robot cannot cross into the forbidden regions. At first glance it would appear that constraining the potential everywhere on the surface of the obstruction to be equal to the potential at the initial conditions, stated mathematically in Eq. (28), would be sufficient. However, this requires the width and skewness to be function of \underline{x} , which would degrade the most attractive property of the APF approach, simplicity. An alternative constraint such that the $V(\underline{x}_{surf}) \geq V(\underline{x}_0)$ still guarantees the robot cannot cross the obstruction surface, and constant values of ψ_k and σ_k can be found to satisfy this constraint.

$$V(\underline{x})|_{surface} \equiv V(\underline{x}_{surf}) = V(\underline{x}_0) \quad (28)$$

Utilizing the assumption of spherical obstructions only, a simple analytical method for the determination of the obstruction APF parameters ψ_k and σ_k can be obtained. The potential on the surface of the obstruction is given by Eq. (29). For spherical obstruction

$$(\underline{x}_{surf} - \tilde{\underline{x}}_k)^T (\underline{x}_{surf} - \tilde{\underline{x}}_k) = R_k^2, \text{ where } R_k \text{ is the radius of the spherical obstruction.}$$

Effectively there are two parameters to be chosen based on one constraint equation, thus only one independent parameter. The prescription for obtaining ψ_k is to choose a value for σ_k and vary \underline{x}_{surf} over all the points on the surface of the obstruction calculating a ψ_k at each value via Eq. (30). Then looking at the resulting potential fields for the various ψ_k , select the ones that do not violate $V(\underline{x}_{surf}) \geq V(\underline{x}_0)$. A simulation was implemented to investigate which values of \underline{x}_{surf} produced ψ_k such that the constraint holds, the results of which are illustrated by Figure 17 through Figure 20.

$$V(\underline{x}_{surf}) = \underline{x}_{surf}^T \mathbf{P} \underline{x}_{surf} + \psi_k \exp \left\{ -\sigma_k^{-1} (\underline{x}_{surf} - \tilde{\underline{x}}_k)^T (\underline{x}_{surf} - \tilde{\underline{x}}_k) \right\} \quad (29)$$

$$\psi_k = \exp \left\{ \frac{-R_k^2}{\sigma_k} \right\} \left(V(\underline{x}_0) - \underline{x}_{surf}^T \mathbf{P} \underline{x}_{surf} \right) \quad (30)$$

Figure 17 shows percent error between the potential at the initial conditions and the potential on the surface of the obstacle. The black line represents the percent error incurred when using the value of ψ_k calculated using \underline{x}_{surf} for which $\|\underline{x}_{surf}\|$ is a maximum over all points on the surface, and the red line corresponds to percent error using the value of ψ_k computed from \underline{x}_{surf} such that $\|\underline{x}_{surf}\|$ is a minimum over all points on the surface. It is clear from this diagram that the latter value of \underline{x}_{surf} must be used to compute ψ_k in order for $V(\underline{x}_{surf}) \geq V(\underline{x}_0)$ to be satisfied. For spherical obstructions finding \underline{x}_{surf} whose norm is a minimum over all points on the surface \underline{x}_{surf}^* is a simple exercise in geometry, and can be computed by Eq. (31) where $\tilde{\underline{x}}$ is the vector connecting the center of the obstruction to the origin.

$$\underline{x}_{surf}^* = \left(1 - \frac{R_{obs}}{\|\tilde{\underline{x}}\|} \right) \tilde{\underline{x}} \quad (31)$$

Figure 18 and Figure 19 depict contour plots of the potential in the xy-plane using ψ_k calculated by Eq. (31). Figure 18 shows the level of contour at the initial conditions as labeled. Figure 19 is a zoomed view of the previous figure in the region of the obstacle (for this case the obstacle center was given by $\tilde{\underline{x}}_k = [2 \ 0 \ 0]^T$). The blue dotted line corresponds to the positions on the surface of the obstacle, and it is evident that the potential is greater than the potential at the initial conditions at all the points on the

boundary. Figure 20 shows the percentage error between the potential at the initial conditions and potential on the surface of the obstruction for ψ_k as calculated by Eq. (31).

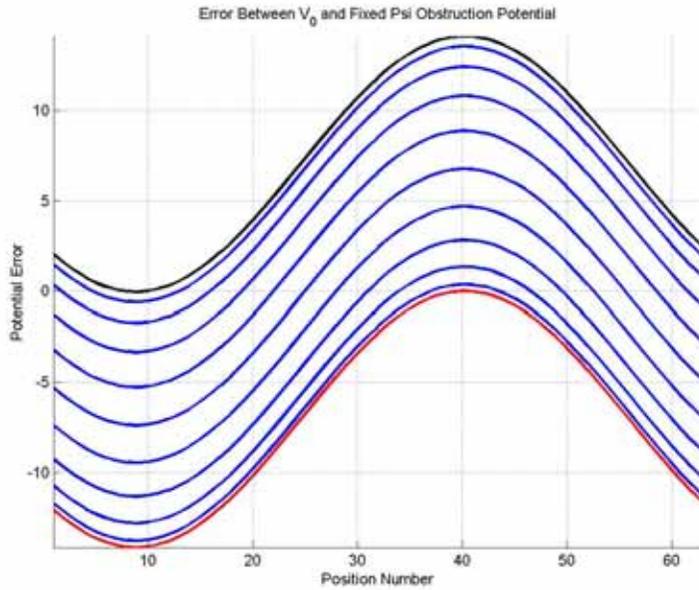


Figure 17. Difference between IC potential and potential on surface of obstructions

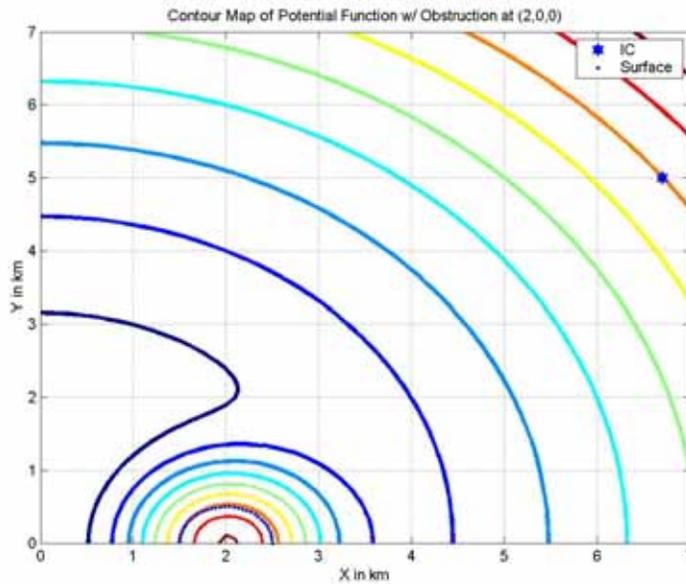


Figure 18. Contour plot of the potential function

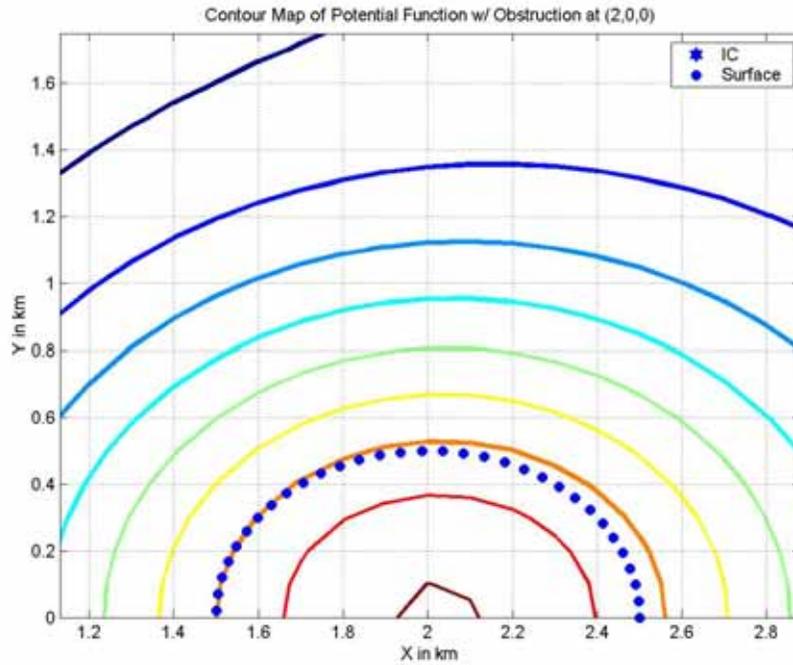


Figure 19. Zoomed view of contour plot

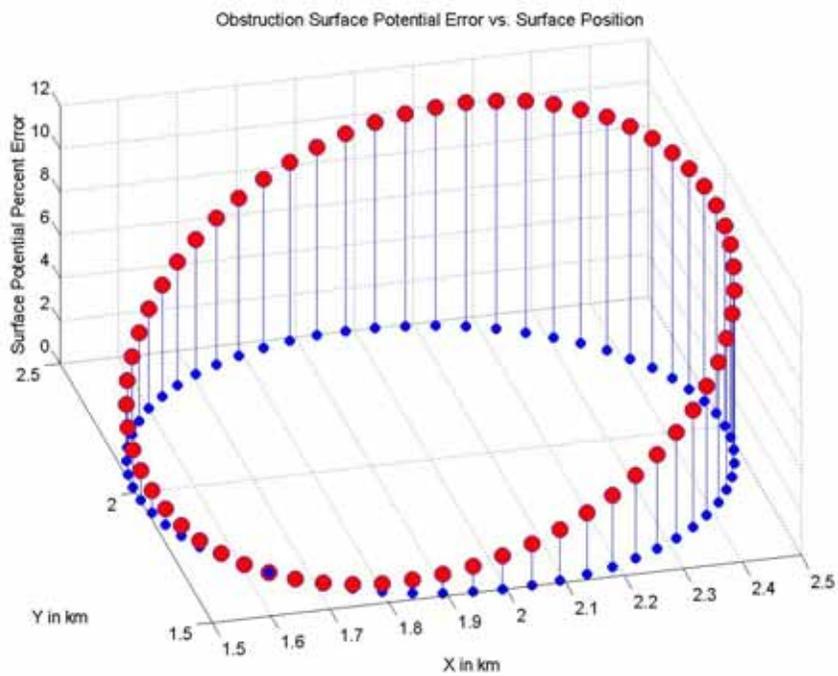


Figure 20. Stem plot of obstruction surface potential error

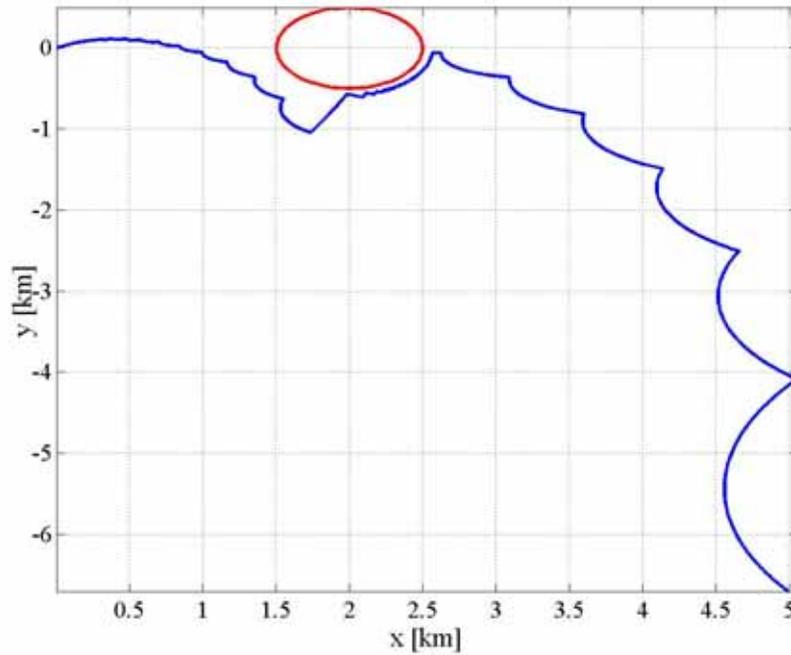


Figure 21. Trajectory traversed by robot for the path-constrained rendezvous example

Numerical Example: Path-Constrained Rendezvous

The initial conditions for the previous example will be used again in this example. The APF guidance algorithm parameters used were $k = 0.25$ and $\mathbf{P} = \mathbf{I}_3$, the different value of k was selected to enhance the illustration of the behavior of the robot around the obstacle. The obstacle is a circle in the plane with center $\tilde{\mathbf{x}} = [2 \ 0 \ 0]^T$ and radius $R_o = 0.5$ km. The parameter $\sigma_k = 0.01$ which leads to $a\psi_k = 1.0855 \times 10^5$, both given in nondimensional units (the nondimensionalization employed the radius of the target as the length scale factor and the mean motion of the target as the time scale factor). Figure 21 shows the trajectory traversed by the robot during the constrained rendezvous maneuver. The duration of the maneuver was approximately 245 minutes and a total $\Delta y = 61.52$ m/s to bring the robot to within a meter of the target. Figure 22 (a) and (b) and Figure 23 (a)

and (b) depict the impulses in the x and y directions, respectively, applied by the APF guidance algorithm as a function of time. The second plot in both figures is zoomed around the times the robot encounters the obstacle.

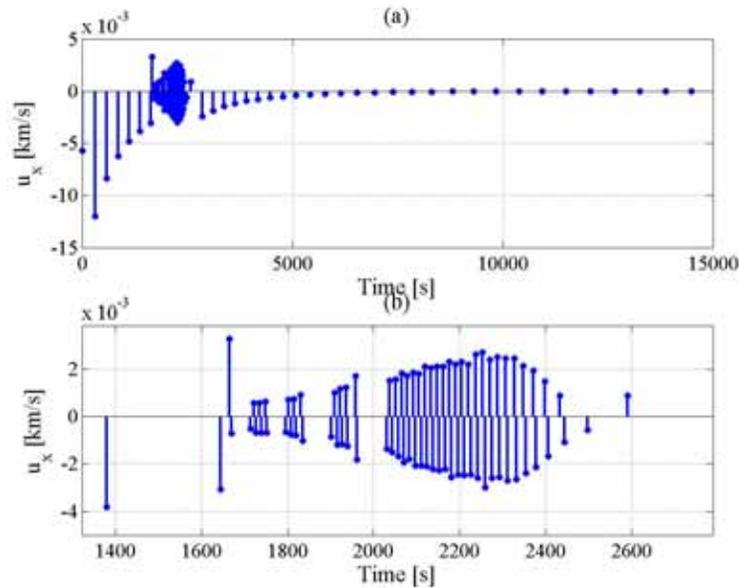


Figure 22. Control action in the x-direction for the constrained rendezvous (a) entire simulation (b) control actions in vicinity of obstacle

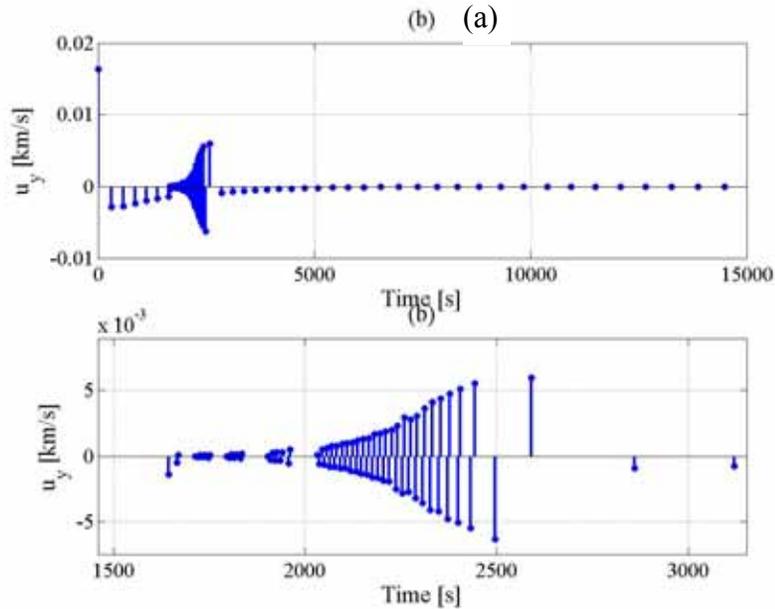


Figure 23. Control action in the y-direction for the constrained rendezvous (a) entire simulation (b) control actions in vicinity of obstacle

Conclusions

In this chapter the motivation for employing artificial potential function (APF) guidance for autonomous ISO applications was presented. Furthermore, position guidance based on APF was developed for unconstrained rendezvous and path-constrained rendezvous, with a numerical example presented for each. Through the numerical examples it was illustrated that APF position guidance can be an effective tool for autonomous proximity operations, a vital element for all ISO systems.

Even though the classical APF examples from above were shown to be effective, they suffer from two predominant drawbacks of APF methods: local minima causing premature termination of algorithm, and sub-optimal performance. Furthermore, the absence of a truly general APF-based obstacle avoidance algorithm needs to be addressed. In the next chapter, the APF guidance framework will be augmented with novel alterations that both conquer these drawbacks and extrapolate the obstacle avoidance algorithm to the most general dynamic case.

CHAPTER 4 ARTIFICIAL POTENTIAL FUNCTION GUIDANCE: A NEW APPROACH

The earliest implementation of APF algorithms is attributed to Khatib and Mampey⁵² in 1978, who used the approach for force control applications. Since then, APF methods have been applied to a wide variety of research areas, most predominantly robot motion planning^{53–55} and computer graphics.⁵⁶ Recently, APF guidance has been utilized in on-orbit applications such as autonomous rendezvous,⁵⁷ position,⁵⁸ and attitude^{59,60} guidance for inspection, maintenance, and assembly operations; and formation control.

Despite the proliferation of APF methods in the robotic literature and their diverse utilization in robotics research, the two primary drawbacks of these methods, spurious local minima and suboptimal performance, have received sparse attention. Furthermore, in the vast body of literature on APF methods a void exists for algorithms capable of handling a general dynamic environment. The first section of this chapter will address the vacancy in the literature pertaining to APF based obstacle avoidance algorithms for truly general dynamic environments. Section two of this chapter will be charged with overcoming the two predominant shortcomings by utilizing a sum of harmonic function primitives to compose the attractive potential function.

Dynamic Artificial Potential Function Obstacle Avoidance Algorithm

Despite the numerous applications of APF methods, most of these instances deal with static environments (potentials). However, a few examples of published works^{59–63} dealing with an APF approach in dynamic environments exist with the most prominent

being the work of Ge and Cui.⁶³ Nevertheless, these works introduce prohibitive restriction as to the nature of the motions of the target and obstacles (e.g., constant velocities for both). Furthermore, they only consider the situation were the robot is required to take obstacle avoidance measures for one obstacle at a time. To the author's knowledge, a truly general APF algorithm for dynamic environments without these restrictions has not been addressed.

The dynamic artificial potential function guidance scheme employed in this section consists of two components: an attractive and a repulsive potential. The attractive potential is a familiar form seen in previous studies on artificial potential field applications. However, the repulsive potential has been augmented with a priority weighting scheme that implements simultaneous obstacle avoidance for all obstacles requiring such. This weighting scheme enables APF guidance to be extended to dynamic environments, with the caveats that the robot possesses sufficient actuation to complete the maneuver, and the refresh rate for updating environment data is sufficiently fast.

Attractive Potential

The attractive potential, U_{att} , given by Eq. (1) includes both the position, $\mathbf{r}(t)$, and velocity, $\mathbf{v}(t)$ of the chaser and position and velocity of the target, $\mathbf{r}_{tar}(t)$, $\mathbf{v}_{tar}(t)$ respectively. The constants $m, n > 1$ and α_p, α_v are design parameters.

$$U_{att} = \alpha_p \|\mathbf{r}_{tar}(t) - \mathbf{r}(t)\|^m + \alpha_v \|\mathbf{v}_{tar}(t) - \mathbf{v}(t)\|^n \quad (1)$$

Similarly to the classical APF approach, we define the virtual attractive force, \mathbf{F}_{att} , as the sum of the gradients of the attractive potential with respect to the position and velocity of the chaser, as given in Eq. (2). Evaluation of two gradient operations yields the expression for the virtual attractive force shown by Eq. (3), where \mathbf{n}_{RT} is a unit vector

pointing from chaser (robot) to the target (goal) and \mathbf{n}_{RT}^v is a unit vector in the direction of relative velocity between chaser and target. It is clear to see that the virtual attractive force goes to zero as the chaser simultaneously obtains the same position and velocity as the target.

$$\mathbf{F}_{att} = -\nabla_p U_{att} - \nabla_v U_{att} \quad (2)$$

$$\mathbf{F}_{att} = m\alpha_p \|\mathbf{r}_{tar}(t) - \mathbf{r}(t)\|^{m-1} \mathbf{n}_{RT} + n\alpha_v \|\mathbf{v}_{tar}(t) - \mathbf{v}(t)\|^{n-1} \mathbf{n}_{RT}^v \quad (3)$$

Repulsive Potential

In order to define the repulsive potential, parameters need to be introduced to quantify the necessity for obstacle avoidance as it pertains to each obstacle in the environment. Define the parameters given by Eqs. (4) – (8) for each obstacle $i = 1 \dots n$. With these parameters in mind, we define the repulsive potential as given by Eq. (9). Qualitatively, the potential is zero when the chaser is moving away from the obstacle, as indicated by ${}^i v_{RO}(t) \leq 0$, while the potential is nonzero for situations when the chase is approaching an obstacle and is within that obstacles range of influence. The repulsive potential is not defined when minimum distance between chaser and the obstacle is less than ρ_m , meaning a collision cannot be avoided. The total repulsive potential is obtained by summing each repulsive potential over all the obstacles, $U_{rep} = \sum_i U_{rep}$.

$$a_{\max} \triangleq \text{maximum acceleration of chaser} \quad (4)$$

$${}^i \rho_s \triangleq \text{min distance between robot and obstacle} = \|\mathbf{r}(t) - {}^i \mathbf{r}_{obs}(t)\| - {}^i r \quad (5)$$

$${}^i \rho_m \triangleq \text{distance under } a_{\max} \mid {}^i v_{RO} \rightarrow 0 = \frac{{}^i v_{RO}^2}{2a_{\max}} \quad (6)$$

$${}^i \rho_0 \triangleq \text{influence range of obstacle} = {}^i \rho_0({}^i v_{RO}(t), T) \quad (7)$$

$${}^i v_{RO}(t) = [\mathbf{v}(t) - {}^i \mathbf{v}_{obs}(t)]^T {}^i \mathbf{n}_{RO}, \quad i = 1 \dots n \quad (8)$$

$${}^iU_{rep} = \begin{cases} 0, & \text{if } {}^i\rho_s \geq {}^i\rho_0 \text{ or } {}^iv_{RO}(t) \leq 0 \\ \eta \left(\frac{1}{{}^i\rho_s - {}^i\rho_m} - \frac{1}{{}^i\rho_0} \right), & \text{if } 0 < {}^i\rho_s - {}^i\rho_m < {}^i\rho_0 \text{ and } {}^iv_{RO}(t) > 0 \\ \text{not defined,} & \text{if } {}^iv_{RO}(t) > 0 \text{ and } {}^i\rho_s < {}^i\rho_m \end{cases} \quad (9)$$

Just as in the attractive force case, the virtual repulsive force is defined as the sum of the gradients of the repulsive potential with respect to the position and velocity of the chaser. After performing the operations and simplifying, the virtual repulsive force, which invokes the necessary obstacle avoidance, is given by Eqs. (10) – (12), where ${}^iv_{RO\perp}$ is the magnitude of the component of the relative velocity of chaser with respect to the i^{th} obstacle in the direction perpendicular to the line connecting the chaser and that obstacle.

$${}^i\mathbf{F}_{rep} = \begin{cases} 0 & , \text{ if } {}^i\rho_s \geq {}^i\rho_0 \text{ or } {}^iv_{RO}(t) \leq 0 \\ {}^i\mathbf{F}_{rep1} + {}^i\mathbf{F}_{rep2} & , \text{ if } 0 < {}^i\rho_s - {}^i\rho_m < {}^i\rho_0 \text{ and } {}^iv_{RO}(t) > 0 \\ \text{undefined} & , \text{ if } {}^i\rho_s < {}^i\rho_m \text{ and } {}^iv_{RO}(t) > 0 \end{cases} \quad (10)$$

$${}^i\mathbf{F}_{rep1} = \eta {}^iU_{rep}^2 \left(1 + \frac{{}^iv_{RO}(t)}{a_{\max}} \right) {}^i\mathbf{n}_{OR} \quad (11)$$

$${}^i\mathbf{F}_{rep2} = \frac{\eta {}^iU_{rep}^2 {}^iv_{RO}(t) {}^iv_{RO\perp}}{a_{\max} {}^i\rho_s} {}^i\mathbf{n}_{OR\perp} \quad (12)$$

Given the repulsive force definition above, the priority weighted scheme is elucidated. Introduce the priority index for the i^{th} obstacle (iPI), calculated as shown by Eq. (13). Again, qualitatively the priority index is a measure of how close are the minimum distance under action of maximum acceleration and the shortest distance between the robot and obstacle. As any one iPI approaches some predefined factor of safety, FOS, (one would prefer not to tempt fate by waiting until the $PI \rightarrow 1$ and the robot

was forced to do a critical avoidance maneuver) the obstacle avoidance algorithm switches from the simultaneous mode to a single obstacle avoidance mode for the obstacle with critical valued PI . Note that we did not consider the case of simultaneous obstacles with critical valued PI s. This logic guarantees a collision free trajectory through the operational space. It is important to note at this juncture that the priority index *implicitly depends on relative velocity* along the chord connecting the two centers of mass of the robot and obstacle. This implicit dependence is crucial to the situation with non-constant velocity targets and obstacles.

$${}^iPI = \begin{cases} 0 & , \text{ if } {}^i\rho_s - {}^i\rho_m > {}^i\rho_0 \text{ or } {}^iv_{RO} < 0 \\ \frac{{}^i\rho_m}{{}^i\rho_s} & , \text{ otherwise} \end{cases} \quad (13)$$

Under normal operating conditions, i.e., no ${}^iPI \geq FOS$, the total repulsive force felt by the robot is calculated as the PI -weighted sum of the individual repulsive forces for each obstacle in the environment, as shown in Eq. (14).

$$\mathbf{F}_{rep} = \sum_i {}^iPI * {}^i\mathbf{F}_{rep} \quad (14)$$

Total Force Interactions

The total virtual force acting on the chaser is the sum of the attractive and repulsive forces, $\mathbf{F}_{total} = \mathbf{F}_{att} + \mathbf{F}_{rep}$. Through Newton's 2nd Law we relate this force to the applied acceleration of the chaser, given by Eq. (15). Notice that if the applied acceleration is greater than the maximum acceleration of the chaser, the maximum acceleration is applied in the direction of the total virtual force. With the applied acceleration, the position and velocity can be obtained via integration of the equation of motion for the chaser.

$$\mathbf{a}(t) = \begin{cases} \frac{\mathbf{F}_{total}}{m}, & \text{if } \left\| \frac{\mathbf{F}_{total}}{m} \right\| \leq a_{max} \\ a_{max} \frac{\mathbf{F}_{total}}{\|\mathbf{F}_{total}\|}, & \text{otherwise} \end{cases} \quad (15)$$

Simulation Results

To illustrate the efficacy of the algorithm, we present two simple 2D simulation scenarios. As the first example, we demonstrate the differences generated by the priority weighted avoidance of the new algorithm with that of Ge and Cui,⁶³ whose work is the baseline for the present analysis, which constrains the target and obstacles to constant velocities. In a second simulation we present a scenario more in line with the intended application, that being the Heterogeneous Expert Robots for On-orbit Servicing (HEROS) architecture,⁶⁴ demonstrating a planar, on-orbit rendezvous with obstacle avoidance. This simulation showcases a situation where target and obstacles have non-constant velocities.

Constant velocity scenario

The 2D constant velocity simulation presented below is a comparison between the algorithm present in Ge and Cui⁶³ and the algorithm derived in the present work. In the former work the repulsive potential is restricted to single obstacle interactions. The parameters $(\alpha_p, \alpha_v, m, n, \eta, \rho_0)$ for both algorithms are equal in order to isolate the effect of the *PI*-weighting scheme. The pertinent parameters of the simulation are as follows. The target has initial position $[10 \ 10]^T$ and constant velocity $[0.1 \ -0.05]^T$. Two obstacles are in the operational space with initial positions $[5 \ 0]^T$ and $[21 \ 11]^T$ and constant velocities $[0 \ 0.1]^T$ and $[-0.05 \ -0.065]^T$, respectively. The chase is of unit mass with the maximum acceleration, A_{max} , also equal to unity. The chase is assigned initial position [1

$1]^T$ and velocity $[0.1 \ 0]^T$. A FOS of 0.9 was used for the *PI*-weighting algorithm. The results of the simulation are shown in Figure 24.

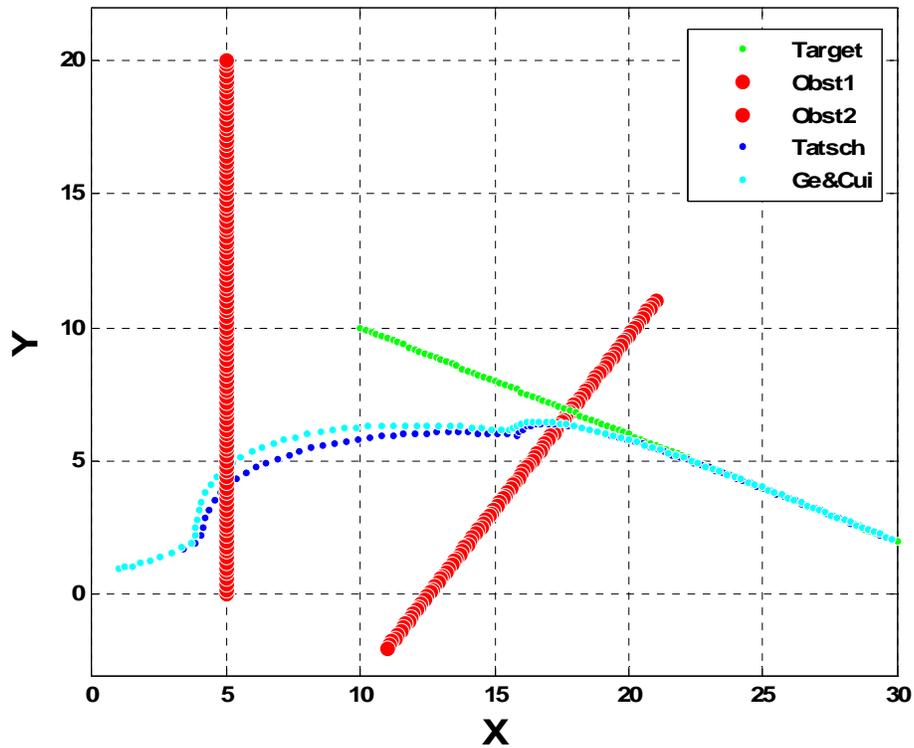


Figure 24. 2D Constant velocity simulation results.

Figure 24 shows the locus of points for all entities in the operational space for the 200 seconds. Most important to note is the more direct route traversed by the chase using the *PI*-weighting scheme when in the influence of both obstacles. As chase approaches the first obstacle in both situations, the differences in the algorithms is illustrated. The chase under Ge's and Cui's algorithm makes a more prominent maneuver with respect to the *PI*-weighting algorithm because the former does not consider the influence of the second obstacle at all. As the chase encounters the second obstacle, the algorithms converge to the same trajectory; which is to be expected considering that in the single obstacle case, the algorithms only differ by the factor of safety. Finally we see that in the

absence of obstacles the algorithms are equivalent given that they employ identical attractive potential/force interactions.

On-orbit rendezvous scenario

The on-orbit rendezvous simulation consists of four spacecraft in Keplerian motion: target, chase, and two obstacles. The initial conditions for all the vehicles are summarized in Table 1. The objective of this scenario is to drive the chase spacecraft using the APF guidance presented above to the target spacecraft while avoiding the obstacles.

Table 1. Orbit parameters for rendezvous simulation.

Vehicle	a [km]	e	i	f	ω
Target	8000	0.1	0°	0°	0°
Chase	9005	0.1	0°	0°	0°
Obstacle 1	10288	0.3	0°	-0.0718°	0°
Obstacle 2	12001	0.4	0°	-0.7171°	0°

Figure 25 illustrates the result of this simulation again in the form of the locus of points for all entities in the operational space. The chase spacecraft starts radially aligned with the target at a 3.6 kilometers higher altitude. Initially the chase proceeds by thrusting in the radial direction to lower its radial velocity (and altitude), thereby increasing its orbital speed. However, it encounters the first obstacle which requires it to thrust in the negative radial direction (raising the chase's altitude) to avoid a collision. Post-avoidance, the chase again thrusts as before. The second obstacle encounter requires a less drastic trajectory change. Once past all obstacles in the environment, the chase finally achieves an orbit altitude lower than that of the target and thus achieves a

positive relative velocity (meaning it is catching the target without thrusting). However, the dependence on the relative position results in small accelerations still being applied to the chase until the rendezvous is achieved.

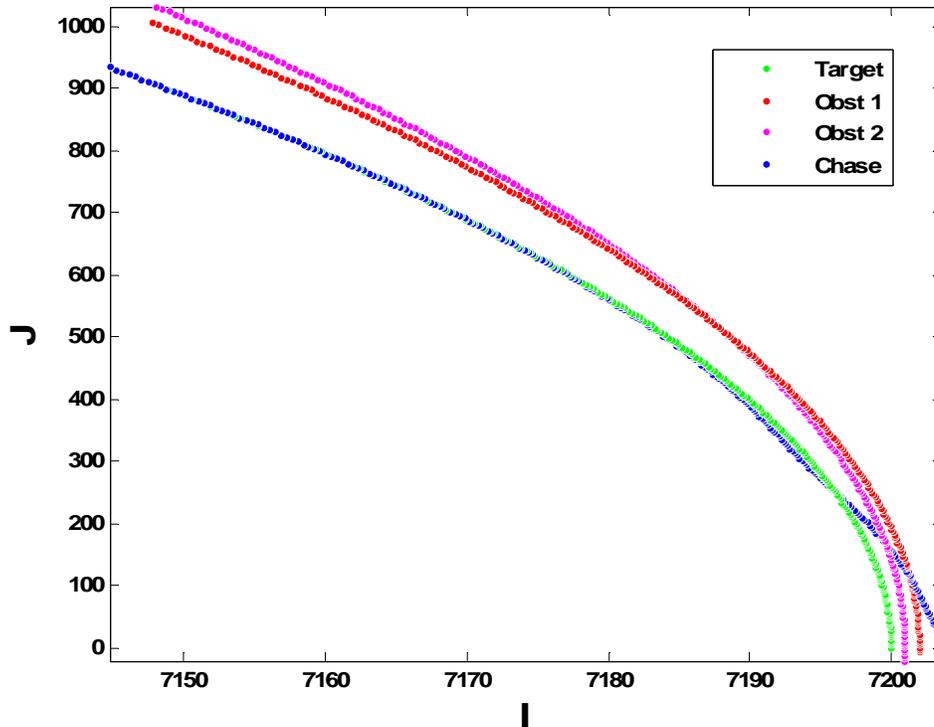


Figure 25. Planar on-orbit rendezvous results.

Figure 26 is a graph of the norm of the applied acceleration of the chase spacecraft. This simulation also took into account the maximum allowable acceleration that could be achieved by the vehicle. We see that initially the chase acceleration is saturated until we encounter the first obstacle (indicated by the dip in the graph). After passing the obstacle we again see a saturation of the acceleration due to increasing lag between the chase and target spacecraft. The second dip in the graph indicates the encounter with the second

obstacle. Beyond the second obstacle encounter the chase acceleration does not saturate again but rather smoothly decays to zero as it approaches the target spacecraft.

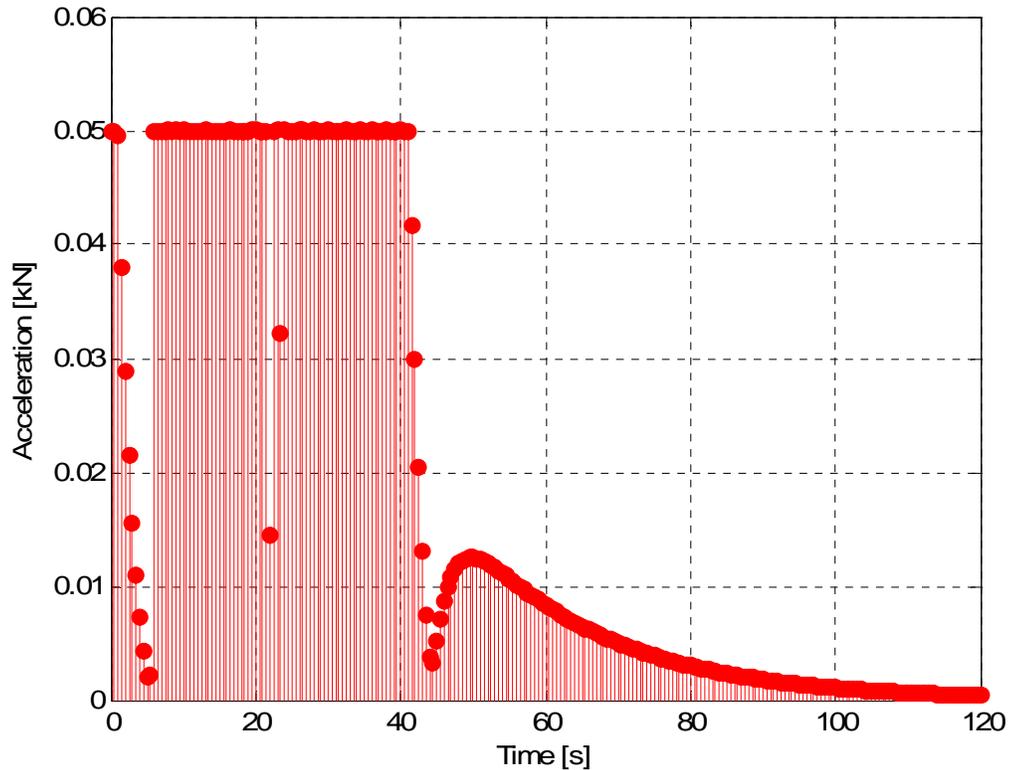


Figure 26.. Norm of applied chaser accelerations.

Optimized Harmonic Attractive Potential

Recall the two prominent flaws mentioned in the introduction: spurious local minima and suboptimal performance. The first of these is easily mitigated by utilization of harmonic potential functions, i.e., functions that satisfy Laplace's equation. By using harmonic functions for construction of the artificial potential field, the resulting potential field is guaranteed to be free of local minima. The first utilization of harmonic potential field for motion control is generally attributed to Satoh⁶⁶ in the mid-1980's. However the work was published in Japanese and did not receive exposure until it was reprinted in

English in 1993. Masoud⁶⁷ has explored both scalar and vector potential field methods using harmonic functions. For an extensive survey on these methods we refer you to the introduction of their work. More recently Waydo⁶⁸ has utilized harmonic potential field methods, in the vein of potential flow analogs in hydrodynamics, for motion planning on the Cornell RoboFlag test bed.

Conquering the suboptimal penalties inherent to APF methodologies without sacrificing its most attractive characteristic, that being computational simplicity, is a daunting task. However we proposed a solution using a library of harmonic function primitives (i.e., simple harmonic functions such as source, sinks, doublets in the hydrodynamic analog) to shape the force/acceleration/velocity field around the spacecraft such that it takes into account the system dynamics.

A high degree of similarity can be achieved by using just two primitives, a sink-vortex pair from the hydrodynamic analog, that have been optimized for sink strength α and vortex circulation Γ to mimic the true velocity field of a dynamic system using a genetic algorithm (GA). The reference velocity field is that for single impulse orbital intercept maneuvers (utilizing the solution of the Clohessy-Wiltshire equations). The parameters α and Γ were optimized over a 50 x 50 grid of points in terms of directional accuracy in the velocity fields. A program that generates the velocity vectors for CW transfers (with time of flight, TOF = $\pi/4$ in this case) at all 2500 points in the grid and the velocity field for the sink-vortex pair was generated by specifying a value for α , Γ and subsequently quantifying the error between the two resulting vector fields. The error between the two velocity fields is quantified by the scalar error measure, ε , given by Eq. (16), where i varies over all points in the grid, and \underline{v} and \underline{v}' are the velocities for the

CW transfer and the sink-vortex field, respectively. In the ideal case (velocity fields are perfectly aligned) the angles θ_i are all zero, and $\cos(\theta_i)$ equal one. Therefore, in the case of perfectly aligned vector fields, ε will equal the number of points in the grid, n , in this case 2500. Using this value of ε as the reference value, ε_{REF} , we define the cost function, F , employed in the GA as a simple difference $\varepsilon_{REF} - \varepsilon$, i.e., the difference between ε_{REF} and the actual value of the summation of the cosines of the angle between the velocity vectors at each point in the grid, as given by Eq. (17). The GA then minimized the value of the cost functional F for all values of α and Γ in their specified ranges. The genetic algorithm parameters used for the optimization are summarized in Table 2.

$$\varepsilon = \sum_i \cos(\theta_i) = \sum_i \left(\frac{\underline{v} \cdot \underline{v}'}{\|\underline{v}\| \|\underline{v}'\|} \right)_i \quad (16)$$

$$F = 2500 - \sum_i \cos(\theta_i) \quad (17)$$

The two parameters of α and Γ were restricted to range between 0.0001 and 5.0.

The output of the GA for eleven runs is shown in

Table 3. As is illustrated by the small standard deviations in

Table 3 (highlighted in blue), both parameters converged to a unique value in their respective ranges. The best fitness score achieved was 43.1029 with corresponding values of $\alpha = 0.1992$ and $\Gamma = 0.2941$ (highlighted in red in

Table 3). Figure 27 plots a comparison of the velocity fields for both the optimized sink-vortex pair (blue) and the CW field (red). Figure 27 illustrates the fidelity in the velocity field that can be achieved with just two harmonic function primitives.

Table 2. Genetic algorithm statistics

Parameter Name	PARAMETER VALUE
Population size	25
Numberof Generations	25
Probability of Mutation	0.05
Probability of Crossover	0.9
Crossover type	Uniform
Fitness Function	Rosenbrock
Number of Bits	20

Table 3. Mean and standard deviations for the two parameters in the GA optimization

Run	α	Γ	Cost Function Score
1	0.397	0.5309	55.2659
2	0.397	0.5309	55.2659
3	0.397	0.5309	55.2659
4	0.397	0.5309	55.2659
5	0.298	0.4451	48.8113
6	0.9852	1.3229	98.0802
7	0.1878	0.2784	43.2809
8	0.3011	0.4162	47.5251
9	0.2099	0.3171	43.5354
10	0.2838	0.3799	47.0728
11	0.1992	0.2941	43.1029
MEAN	0.3685	0.507	53.86110909
STD	0.2206	0.2881	15.52175451

Conclusions

To summarize, with the results of the previous section, the guidance algorithm presented in this work is the combination of a fully dynamic obstacle avoidance algorithm coupled with an attractive potential based on harmonic functions. This form of the attractive potential is guaranteed to be local minima free. Furthermore, by using a sum of the harmonic function primitives to shape the velocity/force/acceleration field, with the intent to harness the underlying dynamics of the system, the suboptimal penalties of artificial potential field methods are mitigated.

The new artificial potential function-based guidance algorithm for dynamic environments introduced a priority-based queuing approach to obstacle avoidance called

the *PI*-weighting scheme. The *PI*-weighting scheme gives the ability to do multiple obstacle avoidance simultaneously, and given its implicit dependence on the relative velocity between the chase (robot) and the obstacles, extends the APF guidance algorithm to general dynamic environments (non-constant velocity targets and obstacles).

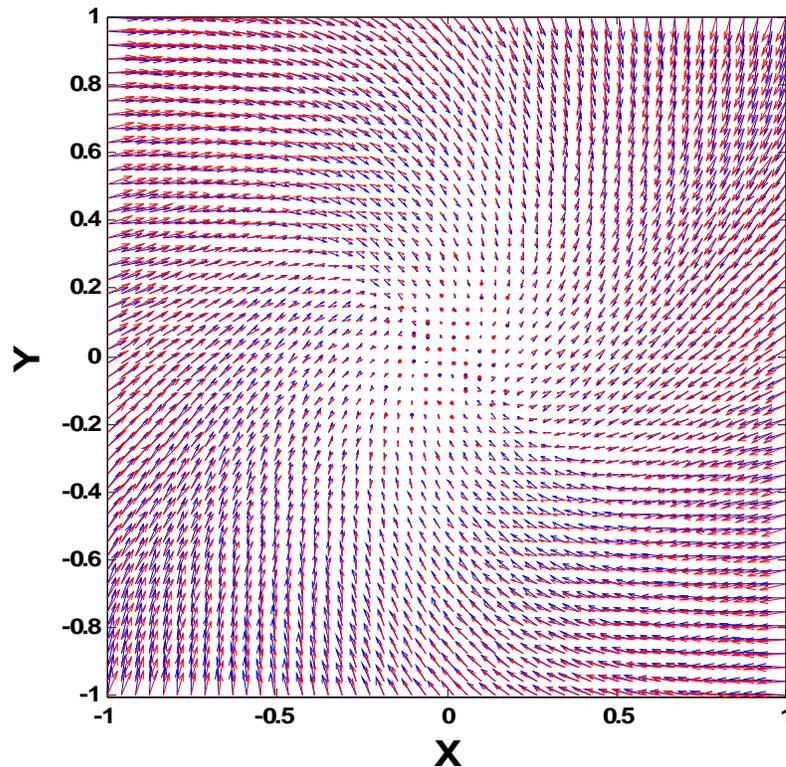


Figure 27. Optimized sink-vortex pair vs. CW based vector field

Through a simple 2D simulation we isolated the effects of the *PI*-weighting scheme and demonstrated the effectiveness of the simultaneous avoidance in situations involving multiple obstacles. Furthermore, a second simulation was presented to demonstrate the efficacy of the overall APF guidance algorithm in terms of the ultimate application, the HEROS architecture. This simulation demonstrated the effectiveness of the algorithm to

handle an on-orbit rendezvous with multiple obstacles, all of which were experiencing Keplerian motion.

CHAPTER 5 ARTIFICIAL POTENTIAL FUNCTION ATTITUDE GUIDANCE

Thus far the spacecraft have been modeled as point masses, however at some stage we must admit the vehicles as rigid bodies (due to directionality constraints, e.g., docking axes alignment) and account for both attitude and position dynamics of the true system. Even though the attitude and position dynamics are highly coupled in the most general case, there are situations where decoupled dynamics are acceptable, and we will deal with uncoupled cases in this work, which allows independent development of an attitude guidance design. In this vein, Chapter 5 develops a novel attitude guidance algorithm utilizing the APF.

The motivation for employing artificial potential functions (APF) for guidance algorithms came from the robotics literature. The literature contains volumes of position (translational) guidance algorithms, and implementations of them, but very few examples of algorithms for rigid body (6 DOF, position and attitude) guidance. Primarily the scarcity of literature on the rigid body motion planning can be attributed to two factors: the configuration space (Cspace) has dimension equal to the DOF of the robot, and configuration obstacles, configurations that result in collisions with the environment, do not have compact representation for high dimensional Cspace. Thus for all but the simplest cases, motion planning is challenging.

Classically, in order to circumvent the complexities of dealing with a high dimensional Cspace, for robots with multiple DOF the attitude, or rotational, DOF have been approximated or constrained. The simplest solution for mitigating high

dimensionality of the Cspace is to ignore the attitude states all together, and merely grow the robot to a sphere with radius equal to its maximum dimension. Although this does reduce the dimensionality of the Cspace, it also reduces the capabilities of the robot; for example, for in-space operations (ISO) systems, the ability to point at objects is crucial, and this approximation eliminates this possibility. An alternative employed in the literature is to hold all but one of the attitude states fixed. Again, this alleviates the issues with dimensionality, but still significantly limits the capabilities of the robot. Therefore, a new approach that allows for incorporation of all attitude states is necessary for ISO systems, preferably one that avoids calculating the Cspace and configuration obstacles. APF guidance for attitude is utilized to accomplish this task.

At this point one might interject that the two deficiencies of APF methods that were discussed in the introduction to Chapter 4 would prohibit the use of the classical approach. However, in the case of attitude guidance, as opposed to position guidance, these deficiencies do not arise. The introduction of local minima which causes premature termination of the algorithm is not an issue for the attitude case because the equations of motion, as will be shown later, possess only two extrema in the state space (which are identical points in the physical space) at the equilibrium points. Suboptimal performance concerns can be dismissed by showing that the guidance law obtained from this approach belongs to a subclass of controllers obtained from a more general Lyapunov analysis. Most obviously an obstacle avoidance algorithm, dynamic or static, is no longer necessary in the attitude guidance case.

In this chapter the common attitude representations and their properties are presented in order to justify the selection of the quaternions of rotation as the attitude

representation scheme. Then, paralleling the procedure for developing the APF position guidance algorithm in Chapter 3, APF guidance for attitude is developed.

Attitude Representation

Before proceeding with the development of the APF guidance algorithms for attitude, an attitude representation scheme must be selected. For attitude representations, it is commonly known that there are no globally nonsingular 3-parameter representations of the rotation group.^{69,70} Therefore, we look to the 4-parameter sets in order to obtain global nonsingularity, and in the present work the quaternions are chosen as the attitude representation. More precisely, we are using the term quaternion as a quaternion of rotation, i.e., the norm is constrained to unity, which is precisely the column vector of the Euler-Rodrigues symmetric parameters.⁶⁹ The primary advantage of the quaternions is that successive rotations result in successive multiplication of 4 x 4 matrices which are commutative.^{69,70} As a result of this computational simplicity, the quaternion attitude representation has become commonplace in computer animation⁷¹ and spacecraft control.^{72,73} Furthermore, the kinematic relationship between the attitude states and the angular velocity can be written as a linear relationship,⁶⁹ as given by Eq. (33). For a comprehensive survey on attitude representations see Shuster.⁶⁹

The development of the attitude APF guidance is identical to the position APF guidance algorithm. However, hidden in the details is a critical difference in state variables between the two cases. For the position case, the generalized coordinates were \underline{x} and the generalized velocities were $\dot{\underline{x}}$, representing the relative position and velocities in the LVLH frame, respectively. This yields the relationship between the derivative of the generalized coordinates and the generalized velocities as unity, shown by Eq. (32).

For the attitude case the generalized coordinates are the elements of the quaternion \bar{q} , and the generalized velocities are the components of the angular velocity of the robot $\underline{\omega}$. The kinematic equation relating the coordinates to the velocities is no longer unity, as illustrated by Eqs. (33) and (34). Although not immediately obvious at this point what complication to the development of the guidance algorithm for attitude this presents, it will be made apparent in the subsequent development.

$$\frac{d\underline{x}}{dt} = \dot{\underline{x}} \quad (32)$$

$$\dot{\bar{q}} = \frac{1}{2} \begin{bmatrix} \tilde{\underline{\omega}} & \underline{\omega} \\ -\underline{\omega}^T & 0 \end{bmatrix} \bar{q} \quad (33)$$

$$\tilde{\underline{\omega}} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \quad (34)$$

Attitude Equations of Motion

The rotational dynamics of a rigid body are governed by the Euler Equations, Eq. (35) where $\underline{\tau}$ is the external torque and \underline{u} is the vector of control torques. For the present work it will be assumed that the external torque is zero.

$$\mathbf{J}\dot{\underline{\omega}} + \underline{\omega} \times \mathbf{J}\underline{\omega} = \underline{\tau} + \underline{u} \quad (35)$$

The quaternion \bar{q} was selected to represent the attitude, which is given in terms of the unit vector in the direction of the eigen-axis of rotation, \mathbf{a} , and angle of rotation, φ , in Eqs. (36) - (38). The kinematic equations relating the angular velocity and the attitude are given by Eqs. (33) and (34) above.

$$\bar{q} = \begin{bmatrix} \underline{q}^T & q_4 \end{bmatrix}^T \quad (36)$$

$$\underline{q} = \sin\left(\frac{\varphi}{2}\right) * \mathbf{a} \quad (37)$$

$$q_4 = \cos\left(\frac{\varphi}{2}\right) \quad (38)$$

Again since APF guidance is based on enforcing stability about the equilibrium of the dynamical system, it is instructive to find the equilibrium solutions for the uncontrolled ($\underline{u} = \underline{0}$) attitude equations of motion. The equilibrium states can be found by solving the following system of equations, derived from Eqs. (33) and (35):

$$-\underline{\omega} \times (\mathbf{J}\underline{\omega}) + \underline{u} = \underline{0} \quad (39)$$

$$\underline{q} \times \underline{\omega} + q_4 \underline{\omega} = \underline{0} \quad (40)$$

$$\underline{\omega}^T \underline{q} = 0 \quad (41)$$

From Eqs. (39) – (41) we observe that the uncontrolled equilibrium state is defined as $\underline{\omega}_e = \underline{0}$ regardless of the steady state values of $q_{4,e}$ and \underline{q}_e .

Continuous APF Attitude Guidance

The development of continuous attitude guidance algorithms applies to situations where the actuation on board is from control moment gyros (CMG) clusters, reaction wheels, or similar continuous input devices.

The control is derived utilizing the general procedure for APF methods discussed in Chapter 3. This general procedure will be repeated here for completeness. Choose a scalar function $V(\underline{x})$ of the states \underline{x} , such that V is positive definite for all $\underline{x} \neq \underline{x}_e$, where \underline{x}_e is the equilibrium state. Furthermore, we constrain the control \underline{u} such that its action

is always in the direction directly opposite the gradient of $V(\underline{x})$. Employing this control ensures the total derivative of V is negative definite for all $\underline{x} \neq \underline{x}_e$, thereby we are guaranteed by Lyapunov's 2nd theorem that $\underline{x} \rightarrow \underline{x}_e$ as $t \rightarrow \infty$.

Following the above procedure, consider the positive definite scalar potential function $V(\underline{q}, \underline{\omega}) = \frac{1}{2}(\underline{q}^T \mathbf{P}\underline{q} + \underline{\omega}^T \mathbf{Q}\underline{\omega})$, where \underline{q} is the vector component of the quaternion and \mathbf{P} and \mathbf{Q} are positive definite and diagonal (for simplicity) matrices. The utilization of the vector component of the quaternion only is justified by the existence of the unity norm constraint, by which the scalar part is computable from knowledge of the vector component. Implementing a continuous control by constraining the attitude rates, under the action of the control ($\underline{\dot{q}}_{0+}$) to be proportional and directly opposite to the resultant of the gradient of V with respect to \underline{q} and $\underline{\omega}$, this can be thought of as a pseudo-steepest decent approach, given by

$$\underline{\dot{q}}_{0+} = -k(\nabla_{\underline{q}}V + \nabla_{\underline{\omega}}V) \quad (42)$$

where $\nabla_{\underline{q}}V = \mathbf{P}\underline{q}$ and $\nabla_{\underline{\omega}}V = \mathbf{Q}\underline{\omega}$. In order to obtain the control input $\Delta\underline{\dot{q}}$ necessary to achieve this condition, bring the attitude rates before control application, $\underline{\dot{q}}_{0-}$, to the right hand side of Eq. (42) resulting in Eq. (43). Since the attitude rates $\underline{\dot{q}}$ are non-causal, the control must be implemented via the angular velocity. Using the kinematic equation to relate $\underline{\dot{q}}_{0-}$ and $\underline{\omega}_{0-}$ and assuming the control input to the attitude rates is equivalent to the control input in the angular velocity states, i.e., $\Delta\underline{\dot{q}} = \mathbf{I}\Delta\underline{\omega}$, yields the control \underline{u} in Eq. (71), where \mathbf{G}_v is given by Eq. (45).

$$\Delta \dot{\underline{q}} = -k\mathbf{P}\underline{q} - k\mathbf{Q}\underline{\omega} - \dot{\underline{q}}_0 \quad (43)$$

$$\underline{u} = -k\mathbf{P}\underline{q} - \mathbf{G}_v \underline{\omega} \quad (44)$$

$$\mathbf{G}_v = k\mathbf{Q} + \frac{1}{2}(q_4 \mathbf{1} + \tilde{\underline{q}}) \quad (45)$$

It is instructive at this point to address the issue of suboptimality for APF methods. The control defined in Eq. (45) is a special case of a more general quaternion feedback regulator, shown in Eq. (46), developed in a seminal paper by Bong Wie,⁷⁵ et al. This larger class of quaternion feedback regulators was shown, with proper gain selection, to provide near-eigenaxis (near-optimal) rotations with guaranteed global stability. For a precise treatment of the general problem we refer you to Ref. 75. We will prove the stability of the special case in the next section.

$$\underline{u} = -(\mathbf{\Omega J} + \mathbf{D})\underline{\omega} - \mathbf{K}\underline{q} \quad (46)$$

Before proceeding with the stability analysis, it is instructive at this point to evaluate the closed-loop equilibrium states now that we have defined a control law. Returning to the equilibrium equations in Eqs.(39)–(41), we find that for $\underline{u} = \underline{0}$ to be satisfied at the closed-loop equilibrium state, both the angular velocity and the vector part of the quaternion must be zero (i.e., $\underline{\omega}_e = \underline{q}_e = \underline{0}$). Therefore, the closed-loop system has two equilibrium states as shown below; however, they represent the same state in the physical space.

$$\underline{\omega}_e = \underline{0} \quad (47)$$

$$\underline{q}_e = \underline{0} \quad (48)$$

$$q_{4_e} = \pm 1 \quad (49)$$

Stability Analysis

The following theorems, along with conditions set forth in the subsequent lemmas, establish the global asymptotic stability of the equilibrium state, as defined previously.

THEOREM 1: The closed-loop system defined by Eqs.(35) and (44) is globally asymptotically stable if \mathbf{M} is positive definite, where \mathbf{M} is defined by Eq. (50).

$$\mathbf{M} = \begin{bmatrix} \mathbf{G}_v & \frac{k}{2}(\mathbf{Q} + \mathbf{1})\mathbf{P} \\ \frac{k}{2}(\mathbf{Q} + \mathbf{1})\mathbf{P} & k\mathbf{P}^2 \end{bmatrix} \quad (50)$$

PROOF: Consider the candidate Lyapunov function $V(\underline{q}, \underline{\omega}) = \frac{1}{2}(\underline{q}^T \mathbf{P} \underline{q} + \underline{\omega}^T \mathbf{J} \underline{\omega})$,

where \mathbf{J} is the inertia dyadic of the rigid body. It is important to note that the inertia matrix is used in the place of \mathbf{Q} due to an elegant simplification that results during the stability proof. The expression for the total derivative of V is shown in Eq. (51).

Substituting for $\mathbf{J}\dot{\underline{\omega}}$ from Eq. (35), with the external torque $\underline{\tau} = 0$, and for $\dot{\underline{q}}$ from the control constraint given by Eq. (42) we obtain Eq. (52). Substituting the control given by Eq. (44) and simplifying yields Eq. (53) (NOTE: the simplification resulting from the use of \mathbf{J} as the derivative matrix gain is the elimination of the cross product term in Eq. (52)).

$$\dot{V} = \underline{q}^T \mathbf{P} \dot{\underline{q}} + \underline{\omega}^T \mathbf{J} \dot{\underline{\omega}} \quad (51)$$

$$\dot{V} = \underline{q}^T \mathbf{P} (-k\mathbf{P}\underline{q} - k\mathbf{Q}\underline{\omega}) + \underline{\omega}^T (-\tilde{\omega}\mathbf{J}\underline{\omega} + \underline{u}) \quad (52)$$

$$\dot{V} = -k\underline{q}^T \mathbf{P}^2 \underline{q} - k\underline{q}^T \mathbf{P}\mathbf{Q}\underline{\omega} - \underline{\omega}^T k\mathbf{P}\underline{q} - \underline{\omega}^T \mathbf{G}_v \underline{\omega} \quad (53)$$

Since \dot{V} is a scalar function, we may transpose any term in the above expression while maintaining its value. Therefore, transposing the second term in Eq. (53) and

combining with the third term leads to the expression in Eq. (54). Rearranging yields a compact expression for the derivative in a quadratic form given by Eq. (55).

$$\dot{V} = -k\underline{q}^T \mathbf{P}^2 \underline{q} - \underline{\omega}^T (k\mathbf{Q}\mathbf{P} + \mathbf{P}) \underline{q} - \underline{\omega}^T \mathbf{G}_v \underline{\omega} \quad (54)$$

$$\dot{V} = \begin{bmatrix} \underline{\omega} \\ \underline{q} \end{bmatrix}^T \begin{bmatrix} -\mathbf{G}_v & -\frac{k}{2}(\mathbf{Q}+\mathbf{1})\mathbf{P} \\ -\frac{k}{2}(\mathbf{Q}+\mathbf{1})\mathbf{P} & -k\mathbf{P}^2 \end{bmatrix} \begin{bmatrix} \underline{\omega} \\ \underline{q} \end{bmatrix} \quad (55)$$

Now, observe that the matrix in Eq. (55) is precisely $-\mathbf{M}$. Therefore, if \mathbf{M} is shown to be positive definite, then \dot{V} is negative definite and by Lyapunov's 2nd theorem global asymptotic stability is guaranteed. Note, \mathbf{G}_v being positive definite is a necessary condition for the positive definiteness of \mathbf{M} .

LEMMA 1: The derivative gain matrix, \mathbf{G}_v , is positive definite if and only if

$$kQ_i > \frac{1}{2}.$$

PROOF: The derivative gain matrix is precisely given by Eq. (56) below. Positive definiteness requires that the principle minors and the determinant are positive. The i^{th} principle minor of \mathbf{G}_v is shown in Eq. (57) with $i=1,2,3$ and $i \neq j \neq k$. The principle minors are positive if Eq. (58) holds. Given that $-1 \leq q_4 \leq 1$, Eq. (58) implies that the necessary condition for the positive definiteness of \mathbf{G}_v is $kQ_i > \frac{1}{2}$.

$$\mathbf{G}_v = \begin{bmatrix} kQ_1 + \frac{q_4}{2} & -\frac{q_3}{2} & \frac{q_2}{2} \\ \frac{q_3}{2} & kQ_2 + \frac{q_4}{2} & -\frac{q_1}{2} \\ -\frac{q_2}{2} & \frac{q_1}{2} & kQ_3 + \frac{q_4}{2} \end{bmatrix} \quad (56)$$

$$PM_i = \left(kQ_j + \frac{q_4}{2}\right) \left(kQ_k + \frac{q_4}{2}\right) + \left(\frac{q_i}{2}\right)^2 \quad (57)$$

$$kQ_i + \frac{q_4}{2} > 0 \quad (58)$$

The determinant of \mathbf{G}_v , given by Eq. (59), is always positive for positive diagonal elements of \mathbf{G}_v . Therefore, $kQ_i > \frac{1}{2}$ is a necessary and sufficient condition for the positive definiteness of \mathbf{G}_v .

$$\det(\mathbf{G}_v) = \prod_{i=1}^3 G_{v,ii} + \sum_{j=1}^3 \left[\left(\frac{q_j}{2}\right)^2 G_{v,jj} \right] \quad (59)$$

It is instructive to point out that, in general, that all diagonal elements $kQ_i + \frac{q_4}{2}$ could be negative and still satisfy the positivity of the principal minors as required by Eq. (57). Since \mathbf{Q} was assume diagonal and positive definite, this implies that $k < 0$. It will be shown that $k > 0$ is required for \mathbf{M} to be positive definite, and therefore, the condition $kQ_i > \frac{1}{2}$ will be used.

LEMMA 2: A block matrix \mathbf{S} , given by Eq. (60) is positive definite if and only if the matrices \mathbf{A} and $\mathbf{R} = \mathbf{D} - \mathbf{B}\mathbf{A}^{-1}\mathbf{B}$ are positive definite.

$$\mathbf{S} = \begin{bmatrix} \mathbf{D} & \mathbf{B} \\ \mathbf{B} & \mathbf{A} \end{bmatrix} \quad (60)$$

PROOF: Start by executing block elementary column operation on \mathbf{S} in order to make the resulting matrix, \mathbf{T} shown in Eq. (61), block upper triangular. From Eq. (61) we see that the upper left block entry is exactly \mathbf{R} . Also, from its upper triangular form, \mathbf{T} is positive definite if and only if \mathbf{A} and $\mathbf{R} = \mathbf{D} - \mathbf{B}\mathbf{A}^{-1}\mathbf{B}$ are positive definite. Since

elementary operations, similarity transformations in general, do not alter the eigenvalues, it is implied that if \mathbf{T} is positive definite only if \mathbf{S} is positive definite and vice versa.

$$\mathbf{T} = \begin{bmatrix} \mathbf{D} - \mathbf{B}\mathbf{A}^{-1}\mathbf{B} & \mathbf{B} \\ \mathbf{0} & \mathbf{A} \end{bmatrix} \quad (61)$$

THEOREM 2: For $kQ_i > \frac{1}{2}$, \mathbf{M} is positive definite.

PROOF: In *Lemma 2* let $\mathbf{S} = \mathbf{M}$ (Note: \mathbf{M} has the necessary structure) which gives the matrices \mathbf{A} , \mathbf{B} , and \mathbf{D} from Eq. (60) as

$$\begin{aligned} \mathbf{A} &= k\mathbf{P}^2 \\ \mathbf{B} &= \frac{k}{2}(\mathbf{Q} + \mathbf{1})\mathbf{P} \\ \mathbf{D} &= \mathbf{G}_v \end{aligned} \quad (62)$$

Using the result of *Lemma 2*, if \mathbf{A} and $\mathbf{R} = \mathbf{D} - \mathbf{B}\mathbf{A}^{-1}\mathbf{B}$ are positive definite then \mathbf{M} is positive definite. In this case $\mathbf{A} = k\mathbf{P}^2$ which needs to be positive definite, which implies that $k > 0$. Furthermore, since \mathbf{P} is a diagonal matrix so is \mathbf{A} , which implies that the inverse of \mathbf{A} is just the reciprocal of the diagonal elements

$\mathbf{A}^{-1} = \text{diag}\left(\frac{1}{A_1}, \frac{1}{A_2}, \frac{1}{A_3}\right)$. This gives \mathbf{R} the form shown in Eq. (63).

$$\mathbf{R} = \begin{bmatrix} D_1 - \frac{B_1^2}{A_1} & D_{12} & D_{13} \\ -D_{12} & D_2 - \frac{B_2^2}{A_2} & D_{23} \\ -D_{13} & -D_{23} & D_3 - \frac{B_3^2}{A_3} \end{bmatrix} \quad (63)$$

\mathbf{R} has the same structure as \mathbf{G}_v , hence it follows from *Lemma 1* that \mathbf{R} will be positive definite if the condition $D_i - \frac{B_i^2}{A_i} > 0$ is satisfied. Assuming that $A_i > 1$ (i.e.,

$P_i^2 > 1/k$) and substituting for **A**, **B**, and **D** the condition for positive definiteness of **R** becomes the expression in Eq. (64).

$$\left(kQ_i + \frac{q_4}{2}\right) - \frac{k^2}{4}(Q_i + 1)^2 P_i^2 > 0 \quad (64)$$

Let the left hand side of the inequality in Eq. (64) be represented by

$$W(P_i) = c - aP_i^2 \text{ as shown in Figure 1, where } 2c = 2kQ_i + q_4 \text{ and } 4a = k^2(Q_i + 1)^2.$$

Therefore, **R** is positive definite for all values of P_i such that W is positive. W is the equation of a parabola, with c characterizing the W -intercept and a governing the width. Examining the range of the two constants a and c , we find that $a > 0$ always, which implies W is always opening downward, and $c > 0$ from $kQ_i > 1/2$ and $-1 \leq q_4 \leq 1$. This implies that for all choices of $kQ_i > 1/2$ a portion of W lies above the P_i axis as can be seen in Figure 1 below. Therefore, for $kQ_i > 1/2$ we can find a range of P_i such that **R** is positive definite and therefore **M** is positive definite. Furthermore, the range on P_i is given by $-\sqrt{c/a} < P_i < \sqrt{c/a}$, however **P** was assumed positive definite previously, meaning P_i must be selected from the positive side of this range.

It is important to note that k is a free parameter in the sense that it is not explicitly specified in the selection of the control gains. Consequently, the gains may be chosen within their range of stability specified by lemmas and theorems and a k found after the fact to guarantee the stability.

The restriction that $0 < P_i < \sqrt{c/a}$ along with the other constraints imposed on the controller parameters from Theorems 1 and 2 and Lemmas 1 and 2 make selection of the control gains **G_p** and **G_d** cumbersome. To circumvent this difficulty we restrict our

control gains to a region of interest and employ an optimization algorithm, specifically a genetic algorithm discussed in the next section, to determine the control gains G_p and G_D .

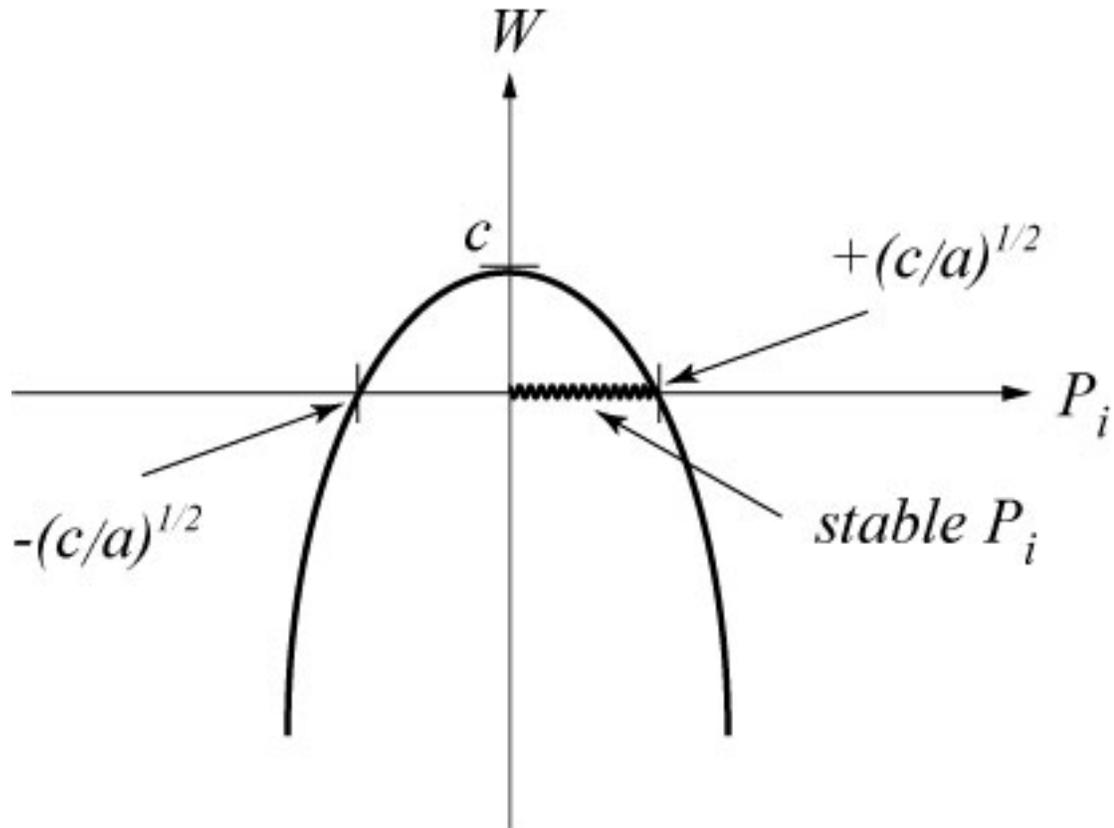


Figure 28. Inequality of Eq. (64)

Stability of Dynamical System

The convergence of the vector component of the quaternion and the angular velocity to zero for the controller derived in the present work is guaranteed from the analysis of the previous section. One would ask at this point as to the behavior of q_4 in the dynamical system. For a general perspective on the nature of the equilibrium states given by Eqs. (39) – (41), look at the linearization of the closed loop dynamical system,

given by Eqs. (35) and (44), around both equilibrium states. The behavior of the dynamical system in the neighborhood of these states is elucidated by the eigenvalues of these linearizations. It can be shown that for $q_4 = +1$ the linearization is stable, implying that the spectrum of the eigenvalues, $\sigma(\lambda) \leq 0$. For $q_4 = -1$ the dynamical system is unstable, meaning there exist $\lambda > 0$. Therefore, regardless of the initial value of q_4 , it will converge to $q_4 = +1$, which depending on the application may be viewed as a blemish on the efficiency of the controller. A plot illustrating the behavior of q_4 when the initial states are very close to desired (equilibrium) state in the physical space, but the unstable equilibrium in the state space is shown below.

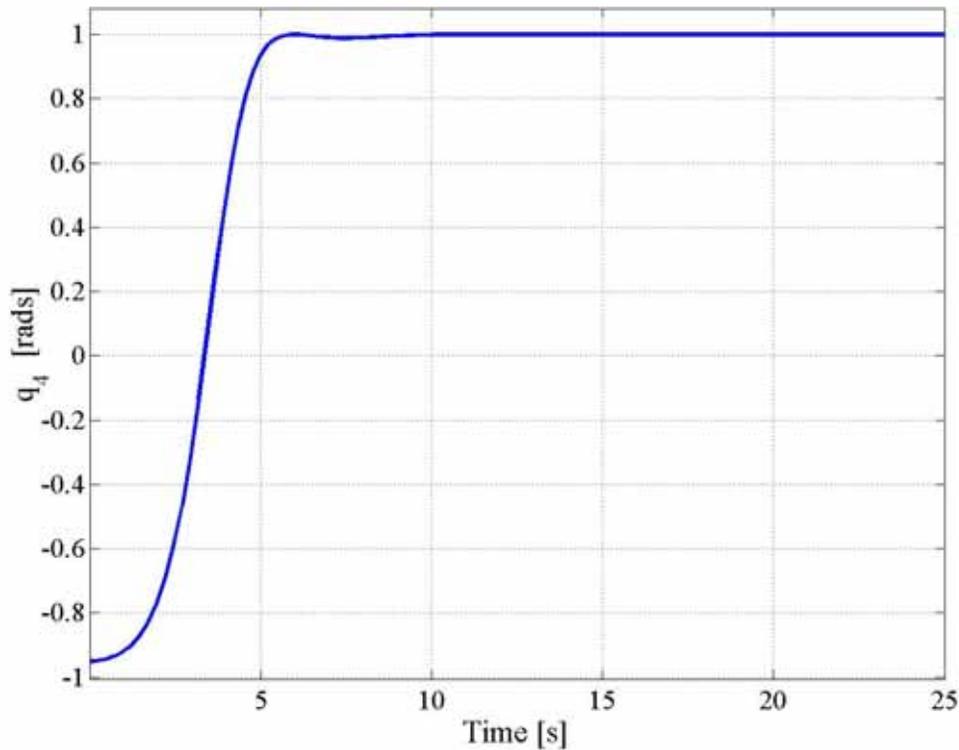


Figure 29. Behavior of q_4

Optimization

To aid in the selection of the control gains \mathbf{G}_p and \mathbf{G}_d a genetic algorithm was used to optimize the gains with respect to a cost function, F , given by Eq. (17) that is a function of the error in the states $\underline{X} = [\underline{\bar{q}}^T \quad \underline{\omega}^T]^T$ and the control effort \underline{u} . The matrices \mathbf{H}_x and \mathbf{H}_d are weight functions for the state error and control states respectively, and for the optimization performed here both were assigned the appropriated dimensioned identity matrix, meaning equal weighting was used for the control and state errors. Since the objective of the control is to bring the state vector to the equilibrium state defined by Eq. (75) – (76), the error states are equivalent to states themselves

$$[\underline{\bar{q}}_{err}^T \quad \underline{\omega}_{err}^T]^T = [\underline{\bar{q}}^T \quad \underline{\omega}^T]^T.$$

$$F = \frac{1}{2} \underline{X}_{err}^T \mathbf{H}_x \underline{X}_{err} + \frac{1}{2} \underline{u}^T \mathbf{H}_u \underline{u} \quad (65)$$

The genetic algorithm parameters used for the optimization are summarized in Table 4.

Table 4. Genetic algorithm statistics

Parameter Name	PARAMETER VALUE
Population size	25
Numberof Generations	25
Probability of Mutation	0.05
Probability of Crossover	0.9
Crossover type	Uniform
Fitness Function	Rosenbrock
Number of Bits	20

The fitness function evaluated the contribution of the error in the states, \underline{q} , q_4 and $\underline{\omega}$ and the control effort over the first 50 seconds of the simulation given by the numerical example in the next section. The control matrices \mathbf{G}_p and \mathbf{G}_D are 3x3 diagonal matrices, and therefore six parameters were optimized. The mean and standard deviation for these six parameters for ten runs of the genetic algorithm are summarized in Table 5.

Table 5. Mean and standard deviation for the six parameters in the GA optimization

Parameter	Mean	Std. Deviation
G_{p1}	1.1083	0.1083
G_{p2}	1.0111	0.0101
G_{p3}	1.1703	0.1293
G_{D1}	0.5253	0.0218
G_{D2}	0.5527	0.0433
G_{D3}	0.8038	0.1094
Cost Function	49.5967	0.4197

The three parameters of \mathbf{G}_p were restricted to range between 1.0 and 2.0, and those of \mathbf{G}_D were restricted to 0.5 to 1.0. As is illustrated by the small standard deviations in Table 5, in general all parameters converged to a unique value in their respective ranges.

Numerical Example

In this section, we use the numerical example given in Wen and Kreuz-Delgado⁷⁰ to compare our controller to their classical PD controller. The objective of the numerical example is to maneuver a spacecraft, represented by its inertia matrix, \mathbf{J} , from an initial orientation and angular velocity to the equilibrium state given by Eqs. (75) – (76), with $q_4 = +1$. The controller in Ref. 70 is defined as shown in Eq. (66), where $k_p = 4$ and $k_v = 8$.

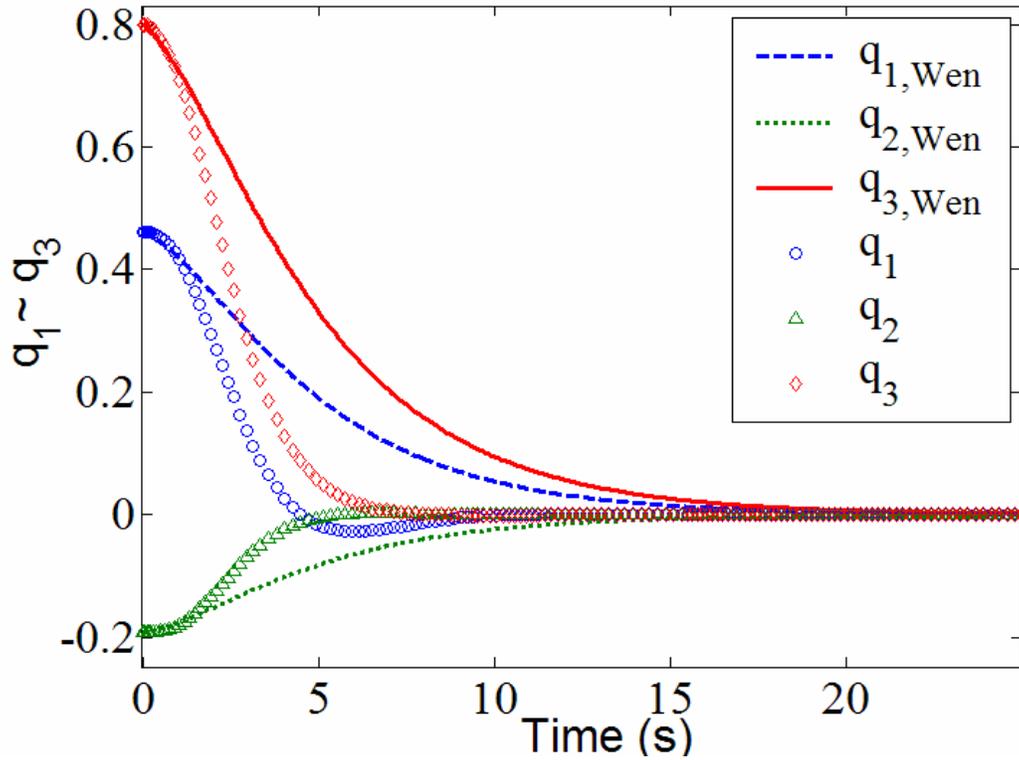


Figure 30. Time response of vector part of quaternion

From Ref. 70, the spacecraft's inertia matrix is given by Eq. (77) and the initial orientation in axis-angle representation is $\underline{a} = [0.4896 \ 0.2052 \ 0.8480]^T$ and $\theta = 2.4648$ radians. The spacecraft's initial angular velocity is $\underline{\omega} = [-0.1 \ 0.3 \ -0.5]^T$. The gains for the controller developed in the present work are listed in Eqs. (68) – (69).

$$\underline{u} = -k_p \underline{q} - k_v \underline{\omega} \quad (66)$$

$$\mathbf{J} = \begin{bmatrix} 1.0 & 0 & 0 \\ 0 & 0.63 & 0 \\ 0 & 0 & 0.85 \end{bmatrix} \quad (67)$$

$$\mathbf{G}_p = \begin{bmatrix} 1.1083 & 0 & 0 \\ 0 & 1.0111 & 0 \\ 0 & 0 & 1.1703 \end{bmatrix} \quad (68)$$

$$\mathbf{G}_D = \begin{bmatrix} 0.5253 & 0 & 0 \\ 0 & 0.5527 & 0 \\ 0 & 0 & 0.8038 \end{bmatrix} \quad (69)$$

Figure 30 illustrates the response of the three components of the vector part of the quaternion, q , and Figure 31 shows the response of the scalar part of the quaternion, q_4 .

Figure 32 and Figure 33 show the response of the three components of the angular velocity and the control input, respectively. As demonstrated by these plots, the controller developed in this paper is capable of performing the stated maneuver. In fact, the maneuver was accomplished at a significant reduction in control effort. It is important to note however, that the gains were arbitrarily selected in Ref. 70 and no optimization was investigated.

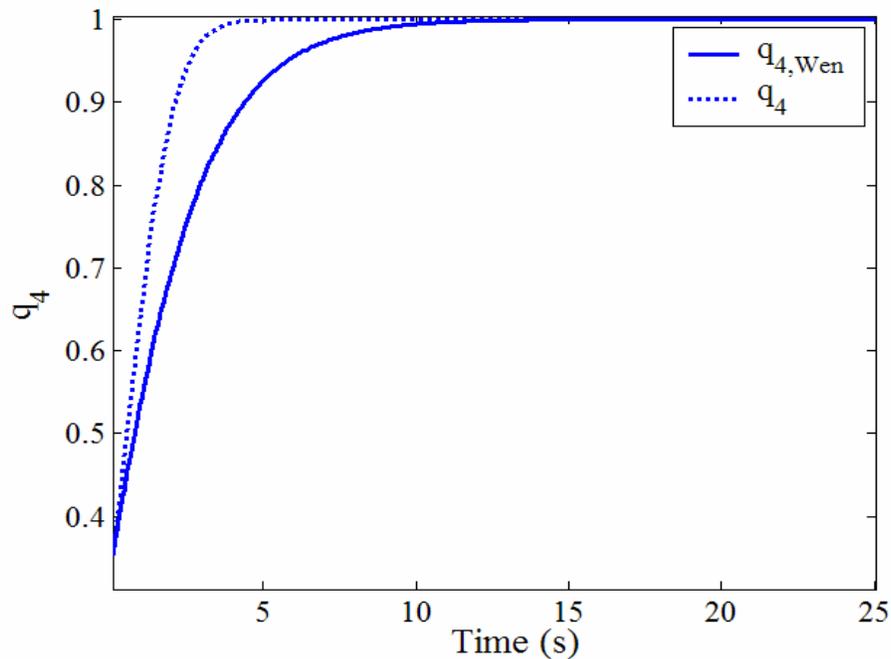


Figure 31. Time response of scalar part of quaternion

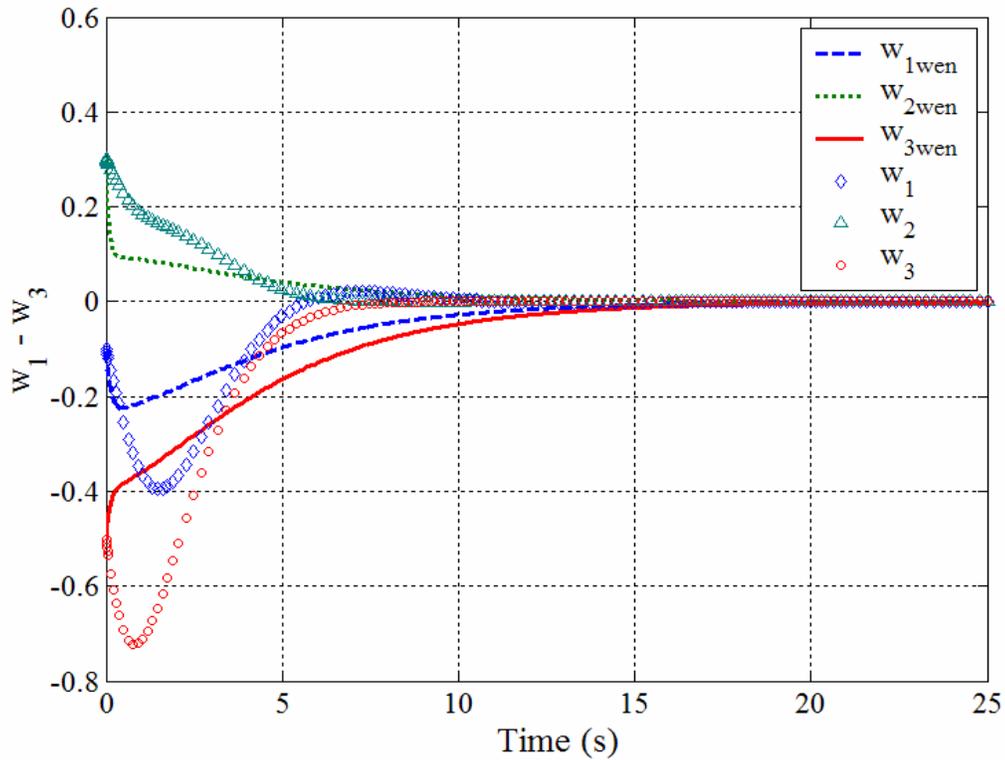


Figure 32. Time response of angular velocity

The effective performance of the controller was demonstrated through comparison with a standard PD controller by a numerical example. It is important to note that even though the gain matrices components were in general smaller than the gains used in the PD controller simulation, the performance achieved was superior, as is illustrated by Figure 30 through Figure 32. Furthermore, the control effort to achieve this improved performance was considerably less, as can be seen in Figure 33. It should be stated here that the convergence of q_4 to +1 introduces a possible inefficiency for attitude maneuvers. If initially q_4 is closer to -1 than +1, then it would be more efficient to drive the attitude toward the equilibrium state at $q_4 = -1$. However, our controller drives the attitude to the equilibrium state with $q_4 = +1$, regardless of the initial value of the scalar part of the

quaternion. One possibility to mitigate this inefficiency is to make the gain matrix \mathbf{G}_p in Eq. (71) proportional to the $\text{signum}(q_4)$, as done in Wie, et. al.⁷³

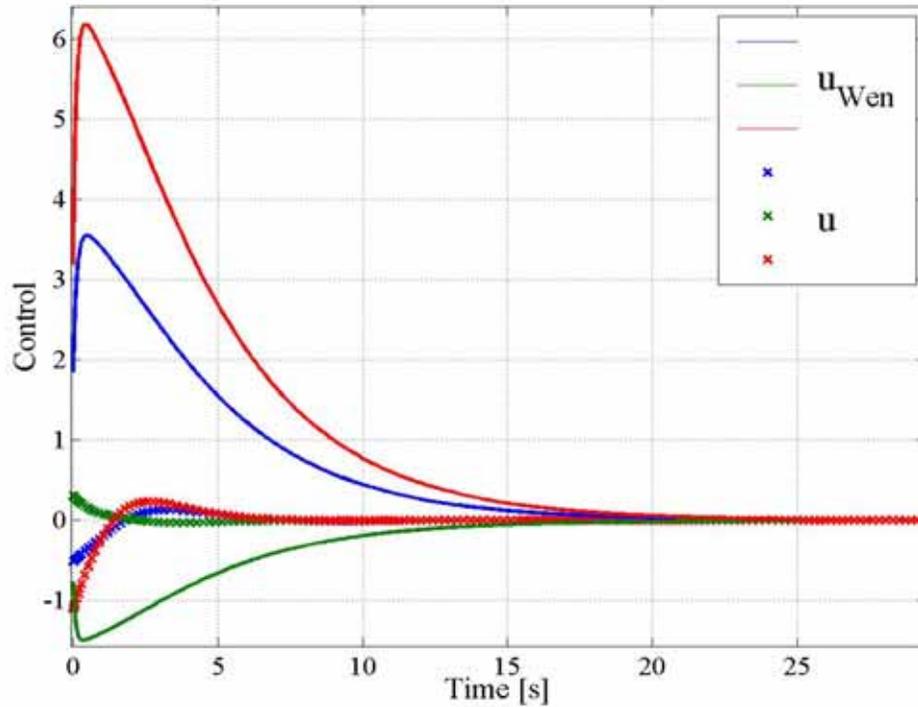


Figure 33. Time response of control effort

Impulsive APF Attitude Guidance

Impulsive attitude guidance is applicable in those situations where the actuation is provided by thrusters as opposed to the devices used in the continuous case, which with the growing sophistication and maturity of micro-Newton capable micro-thrusters is becoming a popular option for attitude control systems (ACS) on board micro and nano class satellites.

Paralleling the development in previous section, let the artificial potential function for this case be $V(\underline{q}) = \frac{1}{2} \underline{q}^T \mathbf{Q} \underline{q}$, where \underline{q} is the vector component of the quaternion \bar{q} . Note, it is acceptable to use the vector portion of \bar{q} only due to the constraint on quaternions of rotation, i.e., $\|\bar{q}\| = 1$. The total derivative of the potential function is given by $\dot{V} = (\nabla_{\underline{q}} V)^T \dot{\underline{q}}$, and the control is defined by Eq. (71) where δ is such that the rate of the vector part of the quaternion after a control impulse $\dot{\underline{q}}_{0+}$ is directly opposite to the local gradient of the potential $\dot{\underline{q}}_{0+} = -k \nabla_{\underline{q}} V$. Defining the control in this manner ensures the derivative of the potential after a control impulse is negative definite, as given by Eq. (70), and therefore, by Lyapunov's 2nd method \underline{q} is guaranteed to converge to its equilibrium value.

$$V'(\underline{x}) = -k (\nabla_{\underline{q}} V)^T (\nabla_{\underline{q}} V) < 0 \quad (70)$$

$$Control = \begin{cases} 0, & V' < 0 \\ \delta, & V' \geq 0 \end{cases} \quad (71)$$

$$\dot{\underline{q}}_{0+} = \begin{bmatrix} \tilde{\underline{q}} & \underline{q} \\ -\underline{q}^T & 0 \end{bmatrix} \underline{\omega}_{0+} = -k \nabla_{\underline{q}} V \quad (72)$$

At this point, the difference in the relationship between the generalized coordinates and velocities become apparent. Since \underline{q} are not causal states, the control must be applied to the angular velocity states. Utilizing the kinematic relationship given by Eq. (33) the control constraint can be transformed to a form in terms of the angular velocity $\underline{\omega}$, given by Eq. (72), while maintaining the constraint on $\dot{\underline{q}}_{0+}$. Substituting the gradient

$\nabla_{\underline{q}} V = \mathbf{Q}\underline{q}$ and premultiplying both sides of Eq. (72) by the inverse of the coefficient matrix of $\underline{\omega}_{0+}$ yields Eq. (73). Thus the control is applied to the angular velocity states such that an impulsive angular velocity correction $\Delta\underline{\omega} = \underline{\omega}_{0+} - \underline{\omega}_{0-}$ that constrains the rate of the vector component of the quaternion to be tangent to the local gradient of the potential.

$$\underline{\omega}_{0+} = -k \begin{bmatrix} \tilde{\underline{q}} & \underline{q} \\ -\underline{q}^T & 0 \end{bmatrix}^{-1} \mathbf{Q}\underline{q} \quad (73)$$

Another difference for APF guidance development between the position and attitude cases is the proof of the stability of the equilibrium states. The equilibrium was the origin in the position case, with the equilibrium derived from the orbital relative motion equations of motion. For the attitude we must return to Eqs. (39) - (41) now that the control has been defined in order to obtain the controlled ($\underline{u} \neq \underline{0}$) equilibria. Just as before, the equilibrium equations show that for $\underline{u} = \underline{0}$ to be satisfied at the equilibrium solutions both the angular velocity and the vector part of the quaternion must be zero (i.e., $\underline{\omega}_e = \underline{q}_e = \underline{0}$). Again there are two equilibrium states (both the same point in the physical space) for the controlled system, restated in Eqs. (74) - (76). Since no differentiation between the equilibrium solutions is necessary in the convergence analysis above, the states will converge to the equilibrium solution to which it is initially closest.

$$\underline{\omega}_e = \underline{0} \quad (74)$$

$$\underline{q}_e = \underline{0} \quad (75)$$

$$q_{4,e} = \pm 1 \quad (76)$$

Numerical Example: Impulse-based APF Attitude Guidance

To illustrate the efficacy of the impulse based APF guidance algorithm for attitude, a numerical example is considered. A satellite in orbit has an initial orientation given in axis/angle representation as $\underline{a} = [0.4896 \quad -0.2032 \quad 0.8480]^T$ and $\theta = 2.4648$ radians, and its initial angular velocity is given by $\underline{\omega} = [0.5 \quad -0.8 \quad 0.9]^T$ radians per second. The central principle inertia matrix representing the spacecraft is given by Eq. (77). The APF parameters used were $k = 0.1$ and $\mathbf{Q} = \mathbf{I}$, where \mathbf{I} is the 3 x 3 identity matrix. The objective of the APF guidance algorithm is to bring the spacecraft with its initial parameters to the equilibrium state $[\underline{q}^T \quad q_4 \quad \underline{\omega}^T]^T = [\underline{0}^T \quad \pm 1 \quad \underline{0}^T]^T$.

$$\mathbf{J} = \begin{bmatrix} 1.0 & 0 & 0 \\ 0 & 0.63 & 0 \\ 0 & 0 & 0.85 \end{bmatrix} \quad (77)$$

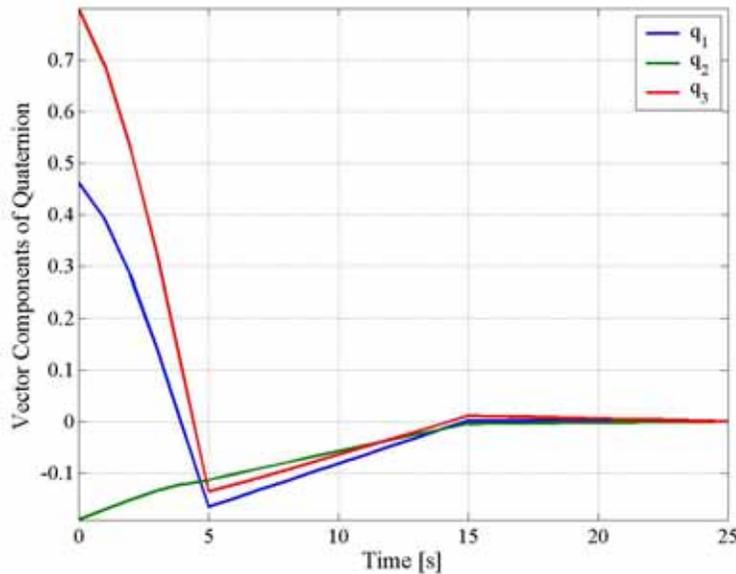


Figure 34. Components of the vector part of the quaternion vs. time

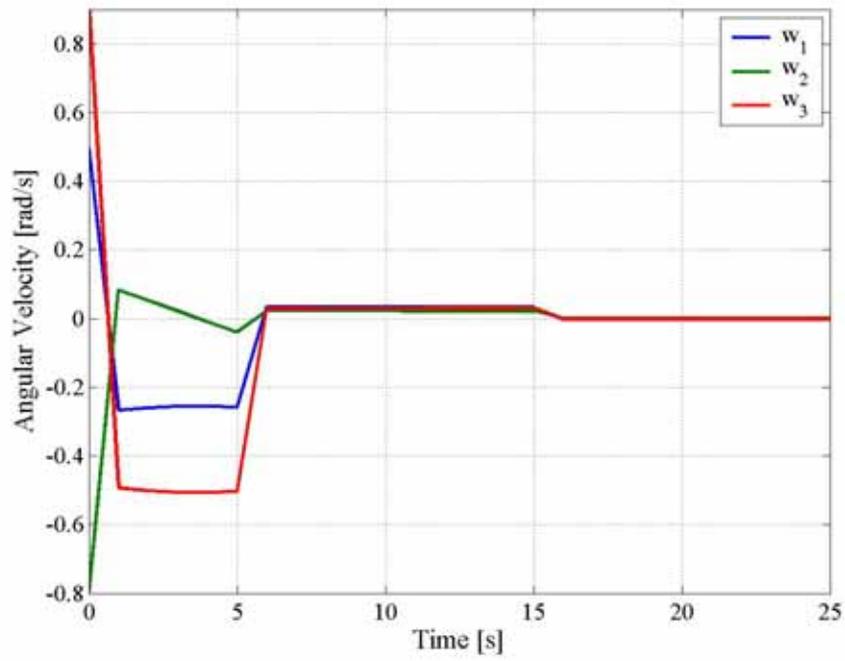


Figure 35. Components of the angular velocity vs. time

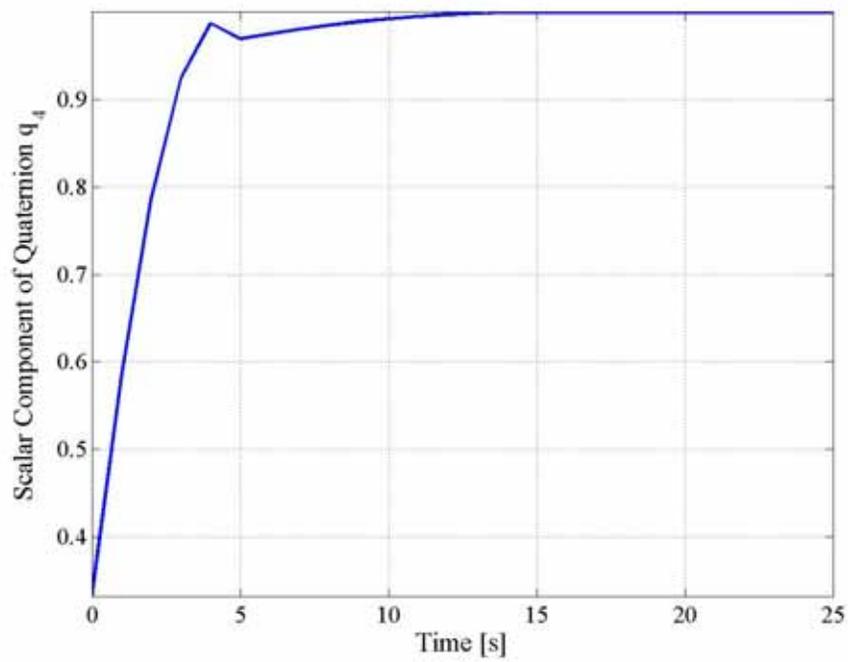


Figure 36. Scalar part of the quaternion vs. time

Figure 34 and Figure 35 show \underline{q} and $\underline{\omega}$ versus time, respectively. Figure 36 illustrates the trajectory followed by q_4 as a function of time. The maneuver duration was 25 seconds and a total impulsive angular velocity change $\Delta\omega_{total} = 1.4034$ radian per second was necessary to bring the $\|\underline{q}\| < 1 \times 10^{-3}$. The impulsive angular velocity inputs are illustrated by Figure 37, and it can be observed that only three impulses totaling 1.4034 radians per second were necessary for the maneuver. Solely for the purpose of illustrating the convergence to the equilibrium point initially closer, the numerical example was repeated with identical parameters, except the initial value of $q_4 = 4.4648$ radians. Figure 38 depicts the trajectory followed by the scalar part of the quaternion in this case, and it is observed that indeed it converges to -1.

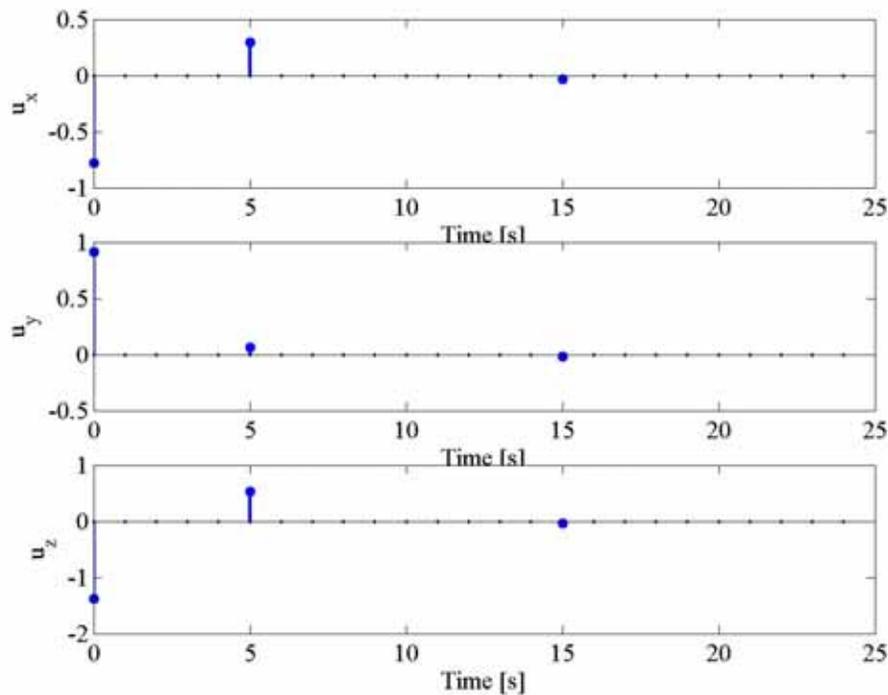


Figure 37. Angular velocity impulses vs. time for the numerical example

An important caveat to the results presented here - the relatively fast convergence of the attitude and angular velocities are direct consequences of a small inertia matrix in Eq. (77) and small initial angular velocities. However, the results here are for illustration of the efficacy of the algorithms to bring the states to their equilibrium values. A more precise analysis of the performance of these algorithms is necessary utilizing more practical inertias and initial parameters.

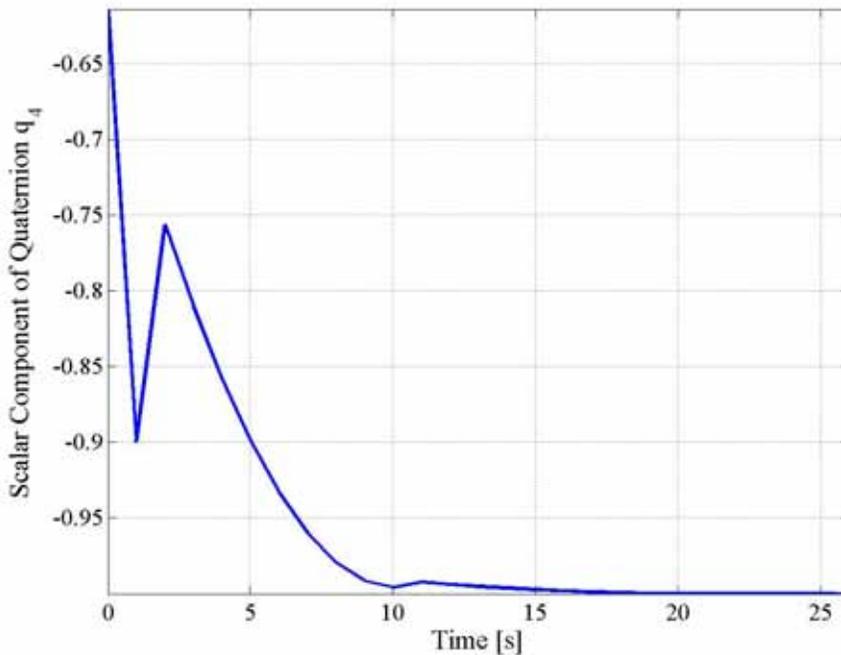


Figure 38. Scalar part of the quaternion vs. time for minus one convergence case

Conclusions

In this chapter two original attitude guidance algorithms were derived from an artificial potential function approach. In the continuous guidance case, the algorithm was shown to belong to a more general class of quaternion feedback regulator to eliminate concerns about suboptimal performance inherent in APF methods, and the algorithm was

proven to be globally asymptotically stable. The controller parameters were obtained using a genetic algorithm in order to mitigate the difficulty in gain selection and to obtain a degree of optimization with respect to state error and control effort. The GA obtained convergence for all six parameters in their respective ranges, as was illustrated by the small standard deviation of each parameter. Furthermore, an impulsive guidance approach was developed and shown to be effective for spacecraft with thruster based ACS.

With these algorithms in place, a more complete development of APF guidance that synthesized both position and attitude APF algorithms for rigid bodies is pertinent. These are the topics under current investigation and will be the directions pursued in the future. In the next chapter, we conclude this dissertation with some recommendations for future work and our concluding remarks.

CHAPTER 6 CONCLUSION AND FUTURE WORK

Conclusion

The current direction of the space industry is toward utilization of autonomous space robotic systems for in-space operations (ISO) and planetary surface exploration (PSE). The state of the art outlined in Chapter 1 illustrates few achievements in real world implementations and test bed environments for ISO space robotics. As a result research in autonomous space robotic systems is both timely and necessitated. With this as the stimulus we explore autonomous guidance techniques for advancing the state of the art ISO space robotic systems.

As a first attempt the classical orbit mechanics methods were explored in Chapter 2. These methods were shown to be insufficient, primarily due to the inability to incorporate obstacle avoidance, for a large class of ISO mission. In Chapter 3 we turned to the robotics literature for alternative approaches and discovered artificial potential function (APF) guidance to be a feasible solution, provided the deficiencies inherent in these methods are overcome; the lack of dynamic obstacle avoidance, spurious local minima, and suboptimal performance.

Chapter 4 developed the key elements that allow APF guidance to be an effective solution for ISO missions. A novel dynamic obstacle avoidance algorithm is generated that is capable of handling general dynamic environments. Furthermore, through the use of a sum of harmonic function primitives we can guarantee no introduction of local minima and mitigate the suboptimal penalties of APF methods.

At some point the point mass assumption for the spacecraft becomes unacceptable and instead must be admitted as rigid bodies. As a result, consideration of the attitude is introduced. In Chapter 5 the classical APF framework is utilized to develop two novel attitude guidance algorithms, one for continuous and one for impulsive type actuation. From characteristics of the attitude control problem and proper potential function selection eliminates the inherent deficiencies of an APF approach. Both controllers were shown to perform effectively for their appropriate application.

Future Work

The solution method presented in this dissertation is an effective solution for autonomous guidance of in-space operations mission, such as assembly, inspection, and servicing. However, further development of this approach is possible in several areas, as outlined below.

- Harmonic Functions
 - Utilization of more complex primitives
 - Use additional primitives in sum
 - Eliminate the use of offline optimization so algorithm could be used for real time guidance
- Reference Velocity Field
 - Generate full nonlinear velocity field (Lambert solution field)
 - Include perturbing effects in reference field
 - Drag
 - J_2
 - Solar radiation

- System Dynamics
 - Fuse attitude and position guidance algorithms for couple attitude/position dynamics

Research in these areas is the focus of ongoing post graduate study.

LIST OF REFERENCES

1. Kennedy, J.F., "Urgent National Needs," Speech to US Congress, Washington, DC, May 25, 1961.
2. Bush, G.W., "Vision for US Space Exploration," Speech to US Congress, Washington, DC, January 14, 2004.
3. Stathopoulos, V., "Lunakhod Spacecraft," <http://www.aerospaceguide.net/spacecraft/lunakhod.html>, December 5, 2005 (last accessed).
4. Jet Propulsion Laboratory, "Mars Exploration Rover Missions," <http://marsrovers.jpl.nasa.gov/home/>, December 5, 2005 (last accessed).
5. Polites, M., "Technology of Automated Rendezvous and Capture in Space," Journal of Spacecraft and Rockets, Vol. 36, No. 2, March-April 1999, pp. 280-291.
6. Legostayev, V.P. and Raushenbakh, B.V., "Automatic Rendezvous in Space," Foreign Technology Division, FTD-HT-23-21346-68, December 1968.
7. Stern, S.A., and Fowler, W.T., "Path-Constrained Maneuvering Near Large Space Structures," Journal of Spacecraft and Rockets, Vol. 22, Sept.-Oct. 1985, pp. 548-553.
8. Soileau, K.M., and Stern, S.A., "Path-Constrained Rendezvous: Necessary and Sufficient Conditions," Journal of Spacecraft and Rockets, Vol. 23, No. 5, pp. 492-498.
9. Mitsushige, O., "ETS-VII: Achievements, Troubles and Future," Proceedings of the 6th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS), 2001, pp. 914-919.
10. "AERCam Sprint," http://vesuvius.jsc.nasa.gov/er_er/html/sprint/index.htm, December 5, 2005 (last accessed).
11. "ROTEX," <http://www.cosimir.com/VR/English/rotex.htm>, January 11, 2005 (last accessed).
12. Skyworker," <http://www.frc.ri.cmu.edu/projects/skyworker/>, December 5, 2005 (last accessed).

13. Necessary, R., Curator, "Robonaut," http://vesuvius.jsc.nasa.gov/er_er/html/robonaut/robonaut.html, December 5, 2005 (last accessed).
14. Space Systems Laboratory, University of Maryland, "Dexterous Robotics at the Space Systems Laboratory," <http://robotics.ssl.umd.edu/ranger/index.shtml>, December 5, 2005 (last accessed).
15. Pedersen, L., Kortenkamp, D., Wettergreen, D., and Nourbakhsh, I., editors, NASA Exploration Team (NEXT), "Space Robotics Technology Assessment Report," Moffet Field, CA, December 2002.
16. Pedersen, L., Kortenkamp, D., Wettergreen, D., and Nourbakhsh, I., editors, NASA Exploration Team (NEXT), "Space Robotics Technology Assessment Report," Moffet Field, CA, December 2002, pp. 17.
17. "Mini AERCam," <http://aercam.nasa.gov/>, December 5, 2005 (last accessed).
18. "NASA's Implementation Plan for Space Shuttle Return to Flight and Beyond," http://www.nasa.gov/pdf/58541main_RTF_rev2.pdf, Vol. 1, Revision 2, April 26, 2004, pp. 1-21 – 1-31.
19. Space Systems Laboratory, University of Maryland, "The Supplemental Camera and Maneuvering Platform (SCAMP)," http://www.ssl.umd.edu/projects/SCAMP/Project_overview.html, December 5, 2005 (last accessed).
20. Pedersen, L., Kortenkamp, D., Wettergreen, D., and Nourbakhsh, I., editors, NASA Exploration Team (NEXT), "Space Robotics Technology Assessment Report," Moffet Field, CA, December 2002, pp. 31.
21. "On-Orbit Servicing (OOS)," <http://www.on-orbit-servicing.com/index.html>, December 5, 2005 (last accessed).
22. RFI 2004-500, Intramural Call for Human and Robotic (H&RT) Proposals, <http://www.moontoday.net/news/viewsr.html?pid=12890> , December 5, 2005 (last accessed).
23. RFI04212004, Intramural Call for Human and Robotic (H&RT) Proposals, <http://www.moontoday.net/news/viewsr.html?pid=12890> , December 5, 2005 (last accessed).
24. Battin, R.H., *An Introduction to the Mathematics and Methods of Astrodynamics*, Revised Edition, AIAA, Reston, VA, 1999.
25. Prussian, J.E. and Conway, Bruce A., *Orbit Mechanics*, Oxford University Press, New York, 1993, pp.139-152.

26. Bate, R., Mueller, D., and White, J., *Fundamentals of Astrodynamics*, Dover Publications Inc., New York, 1971
27. Vallado, D., *Fundamentals of Astrodynamics and Applications*, The McGraw-Hill Companies, Inc., New York, 1997.
28. Battin, R.H., and Vaughan, R.M., "An Elegant Lambert Algorithm," *Journal of Guidance, Control, and Dynamics*, Vol. 7, No. 6, Nov-Dec 1984, pp. 662-670.
29. Gooding, R.H., "A Procedure for the Solution of Lambert's Orbital Value Problem," *Celestial Mechanics*, Vol. 48, 1990, pp. 145-165.
30. Stern, S.A., and Fowler, W.T., "Path-Constrained Maneuvering Near Large Space Structures," *Journal of Spacecraft and Rockets*, Vol. 22, Sept.-Oct. 1985, pp. 548-553.
31. Soileau, K.M., and Stern, S.A., "Path-Constrained Rendezvous: Necessary and Sufficient Conditions," *Journal of Spacecraft and Rockets*, Vol. 23, No. 5, pp. 492-498.
32. Clohessy, W. H., and Wiltshire, R. S., "Terminal Guidance Systems for Satellite Rendezvous," *Journal of the Aerospace Sciences*, Vol. 27, No. 9, 1960, pp. 653-658.
33. Hablani, H., Tapper, M., and Dana-Bashian, D., "Guidance and Relative Navigation for Autonomous Rendezvous in a Circular Orbit," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 3, May-June 2002, pp. 553 – 562.
34. Carter, T.E., "State Transition Matrices for Terminal Rendezvous Studies: Brief Survey and New Example," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No.1, 1998, pp. 148-155.
35. Carter, T.E., "New Form for the Optimal Rendezvous Equations Near a Keplerian Orbit," *Journal of Guidance*, Vol. 13, No. 1, 1990, pp. 183-186.
36. Tschauner, J., "Elliptic Orbit Rendezvous," *AIAA Journal*, Vol. 5, No. 6, 1967, pp. 1110-1113.
37. Humi, M., and Carter, T., "Rendezvous Equations in a Central-Force Field with Linear Drag," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 1, 2002, pp. 74-79.
38. Lozano-Perez, T., and Wesley, M.A., "An algorithm for planning collision-free paths among polyhedral obstacles," *Communications of the ACM*, October 10, pp. 560-570.

39. Hwang, Y.K. and Ahuja, N., "Gross Motion Planning – A Survey," ACM Computing Surveys, Vol. 24, No. 3; ABI/INFORM Global, Sep. 1992, pp.219 – 280.
40. Rao, N., Hareti, S., Shi, W., and Iyengar, S., "Robot Navigation in Unknown Terrains: Introductory Survey of Non-heuristic Algorithms," Technical Report ORNL/TM-12410, Oak Ridge National Laboratory, 1993.
41. Wise, K.D., and Bowyer, A., "A Survey of Global Configuration-Space Mapping Techniques for a Single Robot in a Static Environment," International Journal of Robotics Research, Vol. 19, No. 8, August 2000, pp. 762-779.
42. Wise, K.D., and Bowyer, A., "A Survey of Configuration-Space Mapping Techniques and Applications, Part II: From Dynamic Obstacles to Mechanisms," Technical Report 09/98, Department of Mechanical Engineering, University of Bath, February, 1997.
43. Johnson, C.G., "Robot Motion Planning Survey," Annual Technical Report, Department of Mathematics, Napier University, Edinburg, May 3, 1995.
44. Choset, H., and Kortenkamp, D., "Path Planning and Control for Free-Flying Inspection Robot in Space," Journal of Aerospace Engineering, Vol. 12, No. 2, April 1999, pp. 74-81.
45. Lopez, I., and McInnes, C., "Autonomous Rendezvous Using Artificial Potential Function Guidance," Journal of Guidance, Control, and Dynamics, Vol. 18, No. 2, Mar.-Apr. 1995, pp. 237-241.
46. McQuade, F., and McInnes, C.R., "Autonomous control for on-orbit assembly using potential function methods," The Aeronautical Journal, June/July 1997, pp. 255-262.
47. Roger, A.B., and McInnes, C.R., "Safety Constrained Free-Flyer Path Planning at the International Space Station," Journal of Guidance Control and Dynamics, Vol. 23, No. 6, November-December 2000, pp. 971-979.
48. McQuade, F., Ward, R., and McInnes, C.R.: "The Autonomous Configuration of Satellite Formations using a Generic Potential Function," 5th International Symposium on Spaceflight Dynamics, Biarritz, France, June 26-30, pp. 255 – 262.
49. Nagarajan, N, Tan, S.H., Lee, B.T., and Tan, B.L., "Suitable strategies for In-Plane Orbit Acquisition Using Microthrusters," Proc-The 15th Annual AIAA/USU Conference on Small Satellites, USA, August 2001, SSC01-X1-4.
50. Murray, R. and Li, Z., *A Mathematical Introduction to Robotic Manipulators*, CRC Press, Boca Raton, FL, March 22, 1994.

51. Grantham, W., and Chingcuanco, A., "Lyapunov Steepest Descent Control of Constrained Linear Systems," *Transactions on Automatic Control*, Vol. AC-29, No. 8, 1984, pp. 740-743.
52. Khatib, O., and Mampey, L. M., "Fonction decision-commande d'un robot manipulateur," Report 2/7'156. DERA/CERT, Toulouse, France, 1978.
53. Khatib, O., "Real-time obstacle avoidance for robot manipulator and mobile robots," *The International Journal of Robotics Research*, Vol. 5, No. 1, 1986, pp. 90 – 98.
54. Rimon, E., and Koditschek, D., "Exact Robot Navigation using Artificial Potential Functions," *IEEE Transactions on Robotics and Automation*, Vol. 8, No. 5, 1992.
55. Planas, R.M., Fuertes, J.M., and Martinez, A.B., "Qualitative Approach for Mobile Robot Path Planning based on Potential Field Methods," 16th International Workshop on Qualitative Reasoning, 2002, CD-ROM.
56. Kiupers, J.B., *Quaternions and Rotation Sequences: A Primer with Applications to Orbits, Aerospace, and Virtual Reality*, Princeton University Press, Princeton, New Jersey, 1999.
57. Lopez, I., and McInnes, C., "Autonomous Rendezvous Using Artificial Potential Function Guidance," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 2, Mar.-Apr. 1995.
58. McQuade, F., and McInnes, C., "Autonomous Control of on-Orbit assembly using Potential Function Methods," *The Aeronautical Journal of the Royal Aeronautical Society*, June-July 1997.
59. Tatsch, A., Xu, Y., and Fitz-Coy, N., "A Nonlinear Controller via Artificial Potential Function for Spacecraft Attitude Maneuvers," 28th AAS Guidance and Control Conference, Breckenridge, CO, Feb. 2005, AAS 05-008.
60. Tatsch, A., Xu, Y., and Fitz-Coy, N., "A Nonlinear Controller via Artificial Potential Functions for Impulsive Spacecraft Maneuvers," 2004 Core Technologies for Space Systems Conference, Colorado Springs, CO, CD-ROM.
61. Ko., N.Y., and Lee, B.H., "Avoidability Measure in Moving Obstacle Avoidance Problem and Its Use for Robot Motion Planning," *Proceedings of the 1996 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 3, 1996, pp. 1296-1303.
62. Hussein, B., *Robot Path Planning and Obstacle Avoidance by Means of Potential Function Method*, Ph.D Dissertation, University of Missouri-Columbia, 1989.

63. Ge, S.S., and Cui, Y.J., "Dynamic Motion Planning for Mobile Robots using Potential Field Method," Proceedings of the 8th IEEE Mediterranean Conference on Control and Automation, Rio, Patras, Greece, 2000, CD-ROM.
64. Tatsch, A., Fitz-Coy, N., and Edmonson, W., "Artificial Potential Function Guidance for On-orbit Inspection," Space Automation and Robotics Symposium, Naval Research Laboratory, March 30 – 31, 2005, CD-ROM.
65. Connolly, C. and Grupen, R., "The application of harmonic functions to robotics," Journal of Robotic Systems, Vol. 10, No. 7, pp. 931–946, 1993.
66. Satoh, K., "Collision avoidance in multi-dimensional space using laplace potential," in Preprint of 5th Annual Conference of the Robotics Society of Japan, 1987, pp. 155-156.
67. Masoud, A., *Techniques in Potential-Based Path Planning*, Ph.D. Dissertation, Electrical and Computer Engineering Department, Queen's University, Kingston, Ontario, Canada, Mar. 1995.
68. Waydo, S., "Vehicle motion planning using stream functions," CDS Technical Report 2003-001, California Institute of Technology, 2003. Available online: <http://caltechcdstr.library.caltech.edu/>.
69. Shuster, M., "A Survey of Attitude Representations," Journal of the Astronautical Sciences, Vol. 41, No. 4, Oct.-Dec. 1993, pp. 439-517.
70. Wen, J., and Kreutz-Delgado, K., "The Attitude Control Problem," IEEE Transactions on Automatic Control, Vol. 36, No. 10, October 1991, pp. 1148-1162.
71. Shoemaker, K., "Animating Rotation with Quaternion Curves," SIGGRAPH 1985, San Francisco, CA, July 22-26, Vol. 19, No. 3, pp. 245-254.
72. Joshi, S.M., Kelkar, A.G., and Wen, J., "Robust Attitude Stabilization of Spacecraft Using Nonlinear Quaternion Feedback," IEEE Transactions on Automatic Control, Vol. 40, No. 10, Oct. 1995, pp. 1800-1803.
73. Wie, B., Weiss, H., and Arapostathis, A., "Quaternion Feedback Regulator for Spacecraft Eigenaxis Rotations," Journal of Guidance, Control, and Dynamics, Vol. 12, No. 3, May-June 1989, pp. 375-380.
74. Grantham, W., and Chingcuanco, A., "Lyapunov Steepest Descent Control of Constrained Linear Systems," Transactions on Automatic Control, Vol. AC-29, No. 8, 1984, pp. 740-743.
75. Wie, B., Weiss, J., and Arapostathis, A., "Quaternion Feedback Regulator for Spacecraft Eigenaxis Rotations," Journal Guidance, Control, and Dynamics, Vol. 12, No. 3, May-June 1989, pp. 375-380.

BIOGRAPHICAL SKETCH

Andrew R Tatsch was born in Fairmount, Indiana, on November 24, 1978 but has resided in the state of Florida since 1981. In August of 1997 he was accepted to the University of Florida's College of Engineering in the Department of Aerospace Engineering. He received his bachelor's degree in aerospace engineering in May of 2001.

In August of 2001 Andrew was awarded the Alumni Fellowship for pursuing a doctoral degree in aerospace engineering at the University of Florida. He completed his master's degree in 2004 and was subsequently admitted to candidacy for his PhD.

His research interests include guidance, navigation, and control of autonomous in-space operational system.