12-2023

Virtual Structures Based Autonomous Formation Flying Control for Small Satellites

Tyson K. Smith
Utah State University, tyson.smith@usu.edu

VIRTUAL STRUCTURES BASED AUTONOMOUS FORMATION FLYING CONTROL FOR SMALL SATELLITES

by

Tyson K. Smith

A dissertation submitted in partial fulfillment
of the requirements for the degree

of

DOCTOR OF PHILOSOPHY

in

Electrical Engineering

Approved:

Greg Droge, Ph.D. Major Professor	David Geller, Ph.D. Committee Member
Stephen Whitmore, Ph.D. Committee Member	Jonathan Phillips, Ph.D. Committee Member
Don Thompson, Ph.D. Committee Member	D. Richard Cutler, Ph.D. Vice Provost of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2023

ABSTRACT

Virtual Structures Based Autonomous Formation Flying Control for Small Satellites

by

Tyson K. Smith, Doctor of Philosophy

Utah State University, 2023

Major Professor: Greg Droge, Ph.D.
Department: Electrical and Computer Engineering

Autonomous formation flying of small satellites is increasingly becoming a desire of many space organizations. This research develops and evaluates a decentralized, optimization-based control that can be used for autonomous formation flight of small satellites. This architecture is based on a virtual structure approach within a model predictive control framework to enable a decentralized control system. The architecture reduces fuel usage by generating an optimized guidance trajectory that the MPC follows. The relative dynamics used in MPC framework are represented by the Hill-Clohessy-Wiltshire equations and the D'Amico relative orbital elements. One advantage to using MPC is the ability to define constraints. As part of this work, switching surfaces are used as boundary constraints. The boundaries are designed such that no two agents have overlapping regions. This allows the vehicles to execute avoidance strategies without continually maintaining the trajectories of other agents. The two types of switching surfaces used are a polytope approximation of a sphere and state constraints based on the D'Amico ROEs. A performance comparison is completed as part of this research, namely the ΔV and computation time is compared for several objective norm functions, namely the L_1, L_2, and L_∞ norms. The framework developed as part of this research allows for the desired formation of satellites to be maintained while also allowing freedom to maneuver within allowable bounds. The virtual structure, MPC based framework combined with the switching surfaces enables a scalable method that can support satellite formations to coordinate safely, while minimizing fuel usage.

(165 pages)

PUBLIC ABSTRACT

Virtual Structures Based Autonomous Formation Flying Control for Small Satellites

Tyson K. Smith

Many space organizations have a growing need to fly several small satellites close together in order to collect and correlate data from different satellite sensors. To do this requires teams of engineers monitoring the satellites orbits and planning maneuvers for the satellites every time the satellite leaves its desired trajectory or formation. This task of maintaining the satellites orbits quickly becomes an arduous and expensive feat for satellite operations centers. This research develops and analyzes algorithms that allow satellites to autonomously control their orbit and formation without human intervention. This goal is accomplished by developing and evaluating a decentralized, optimization-based control that can be used for autonomous formation flight of small satellites. To do this, virtual structures, model predictive control, and switching surfaces are used. An optimized guidance trajectory is also develop to reduce fuel usage of the system. The Hill-Clohessy-Wiltshire equations and the D'Amico relative orbital elements are used to describe the relative motion of the satellites. And a performance comparison of the L_1, L_2, and L_∞ norms is completed as part of this work. The virtual structure, MPC based framework combined with the switching surfaces enables a scalable method that allows satellites to maneuver safely within their formation, while also minimizing fuel usage.

Dedicated to Mary Ann, Olivia, and Jemma

ACKNOWLEDGMENTS

I would like to thank Dr. Greg Droge. Without his advisement and constant encouragement I would not have been able to complete this work. His insight and knowledge into the subject matter steered me through this research. I would also like to give a special thanks to John Akagi, whose analysis support added greatly to this research.

My biggest thanks goes to my family for all the support they have shown me throughout this research. Especially to my wife Mary Ann, thanks for all your support. You are amazing!

Tyson K. Smith

CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Society has had the need to correlate data collected from different satellites since the late 1960's when data from the U.S., Soviet, and European Space Research Organization satellites was correlated to understand how large solar flares interacted with the Earth's magnetic field and ionosphere [1]. A decade later Labeyrie would propose the idea of a stellar interferometer using free flying telescopes [2]. This was the first time the scientific community had realized the advantages of having a group of satellites flying together, or in formation, with the ability to collect and correlate data.

The idea of using multiple spacecraft for interferometer measurements was first studied in 1977 by Sholomitsky [3]. In the 1980s several multiple-spacecraft interferometer mission architectures were proposed. These formation flying studies focused on possible orbits and ΔV analysis for earth orbiting missions [4, 5]. In the late 1980s to early 1990s researchers turned their focus to developing relative trajectories that were more fuel efficiency [6] and aerodynamic drag compensation methods [7, 8]. It was in 1996 that the first study of a formation flying control was presented by Wand and Hadaegh [9].

Today, there are dozens of missions that either have flown or are currently in development that require some form of formation flying between two or more spacecraft [10–20]. The desire for small satellite formation flying missions continues to increase [21]. Of the missions that have flown, varying levels of success were achieved. Of those that did successfully demonstrate formation flying capability, the majority required the maneuvers to be calculated on the ground and uplinked to each of the spacecraft individually. In the context of this work, formation flight is defined as a fully autonomous, networked system with all vehicles working as one entity toward a shared goal. Specifically, we define formation flight as a set of three or more spacecraft whose dynamic states are coupled through a common control law, such that they maintain a desired arraignment and spacing. But within the formation the amount of time the spacecraft dedicates to maintain the formation, i.e. maneuvering, versus performing the primary mission objectives must be considered. The formation control algorithm cannot command maneuvers at such a rate that the spacecraft does not have the time or resources to perform its primary mission.

Manual coordination of satellite maneuvers becomes increasingly arduous with ever rising numbers of satellites due to the many factors that must be considered during planning. These factors include, but are not limited to: fuel consumption, overcoming disturbances or perturbing forces, data distribution between spacecraft, collision avoidance, mission operational constraints, and computation feasibility. This work addresses various aspects of these challenges by developing a novel control strategy to balance mission operations with fuel consumption for maintaining the relative position of the satellite formation. Switching strategies are developed to determine when the satellite can coast, allowing it to focus on mission operations, and when the satellite must maneuver. When a satellite must maneuver, an optimal guidance trajectory is determined that brings the vehicle back to its desired relative position, with a model predictive controller used to move the vehicle along the guidance trajectory. A virtual structure is introduced to reduce communication for relative position maintenance.

Previous works have employed strategies to optimize fuel usage and overcome perturbations [22–26]. While an optimal fuel saving maneuver may be found and is beneficial, it is also important to have controller feedback to overcoming disturbances from erroneous motion models. Model predictive control (MPC) is a method for using optimal control techniques within a feedback loop to balance multiple desired outcomes [27]. An MPC framework solves the optimization problem given the current state of the system, one or more control inputs from the optimal control trajectory are executed, and the process is repeated with the new state used as the initial condition. Several researchers have implemented an MPC framework based on the Hill-Clohessy-Wiltshire (HCW) relative orbital dynamic equations [28–32] and were able to maintain control of a spacecraft within their desired tolerances.

The HCW dynamics are attractive to use with MPC because they are a well understood, linear, time-invariant model. Additionally, the use of Cartesian coordinates provides an intuitive understanding of the spacecraft states. While the HCW equations are attractive to use, using Cartesian coordinates to define a spacecraft's state can result in some challenges. For example, using a spacecrafts Cartesian position to define the distance between the actual and desired spacecraft states may not produce the most meaningful results. A spacecraft may have a position that perfectly matches a desired location but has a velocity perpendicular to the desired velocity. This can cause the controller to believe that its constraints are met and erroneously turn off. While an HCW based architecture is developed and tested as part of this work, the difficulty of matching the full relative orbital elements and velocities lead to the utilization of D'Amico's relative orbital elements (ROEs) in this work. Using an MPC and switching surface based on the ROEs accounts for the interplay between the control inputs and the natural dynamic motion of the system that further reduce fuel usage when compared to a framework based on the HCW equations. Modeling the constraints and the control in ROE spaces allows the system to better match the desired orbit, not just a position state at a given time.

Collision avoidance also needs to be considered when performing autonomous formation control. Lim et al., [24], guaranteed collision avoidance by developing a framework that held the eccentric and inclination vectors of the D'Amico ROEs parallel to each other. Maintaining these vectors parallel to each other guarantees collision avoidance within a formation, but it does not guarantee a collision free transfers nor does it guarantee the formation is maintained. DiMauro et al., [26], used a six-sided box as a keep out volume for the deputy spacecraft, but DiMauro found that the his approach increased the computation time to a degree that it was deemed not suitable for flight. Tillerson et al., [25], also used a six-sided Euclidean box as a switching condition, but postulated that using a higher dimensional shape could reduce fuel costs. This work builds upon the previous work by implementing a switching conditions based on a polytope approximation of a sphere, and switching conditions based on ROE state constraints. This research also allows the ability to ensure the satellites honors safety constraints when following the guidance trajectory. An evaluation of different objectives for linear (L_1 and L_∞) and quadratic (L_2) programming solutions is performed in terms of fuel usage and computation requirements. It is shown that the L_1 or L_∞ maybe be more applicable given a spacecraft with a high thrust, non-throttlable, on/off type propulsion system. The L_2 norm is more applicable for a smaller thrust, longer duration burning system, such as an electric propulsion system.

As previously mentioned, the challenges that exist in spacecraft formation flight are addressed, namely; fuel consumption, overcoming disturbances or perturbing forces, data distribution between spacecraft, collision avoidance, mission operational constraints, and computation feasibility, by developing a decentralized, optimization-based control that can be used for autonomous formation flight of small satellites. Through developing a switched, model predictive control framework based on a virtual structure architecture.

The remainder of this dissertation is organized as follows: First a literature review of the existing approaches and how they apply to spacecraft formation flying is presented in Chapter 2. This is followed by background information on the relevant methods used for this research, namely virtual structures, the relative dynamics, model predictive control, and switching surfaces in Chapter 3. Chapter 4 presents the in-depth technical details of the formulation and evaluation of the virtual structure-based MPC switching strategy that can be implemented during flight of satellite formation. The formulation in Chapter 4 includes the use of the HCW equations to define the relative motion. Chapter 4 describes the development of the polytope sphere approximation that is used as the switching surface. A comparison of the computation time and fuel usage for L_1, L_2, and L_∞ norm formulations of the framework is presented. Chapter 5 builds upon the work presented in Chapter 4 by adding an optimized guidance trajectory to the formation flying framework. Chapter 5 performs an updated comparison of the different norms. Chapter 6 provides technical detail for a an approach similar to what is presented in Chapter 5 but where the D'Amico ROEs are used as relative dynamics, and state constraints based on the D'Amico ROEs are used as the switching surfaces. Chapter 7 then provides some concluding statements and observations. Finally, Appendix A provides a more extensive literature survey of the different formation control approaches in the literature to include those not selected to be part of this research. Appendix B provides a derivation of the HCW equations.

CHAPTER 2

Literature Review

This chapter discusses the relevance of this research and presents research previously performed in this area of study. A more extensive literature survey of the work previously performed in the area of spacecraft formation flying is presented in Appendix A.

2.1 Spacecraft Formation Flying

Formation flying continues to be an area of growing study. Their are several missions that either have flown or are currently in development that require some form of formation flying, between two or more spacecraft [10,11]. A few examples of recent or current missions that require some form of coordination are: Prototype Research Instruments and Space Mission technology Advancement (PRISMA) [12], Canadian Advanced Nanospace eXperiment-4 and 5 (CanX-4,-5) [13], Gravity Recovery and Climate Experiment (GRACE) [15,16], the German Space Center's phased array instramment, TerraSAR-X, and Digital Elevation Measurement (TanDEM-X) [17, 18], and the NASA's Magnetospheric Multi-Scale (MMS) mission [19,20]. The ability to collect data using multiple satellites autonomously, has great benefits to the scientific and defense communities.

Formation flying is a technology that organizations like the European Space Agency (ESA) [12,33], National Aeronautics and Space Administration (NASA) [12,34,35], and the Department of Defense [36], see as essential to the success of their missions. Formation flying technology is critical for a wide rage of missions. Formation flying missions orbiting Earth unlock the ability to perform sparse antenna array missions, distributed sensing missions, advanced communications and internet access, or spatial sampling for applications such as gravitational mapping, and interferometric synthetic aperture radar [11,37].

The idea of formation flying also allows for many lower cost spacecraft to fly multiply different payloads that traditionally would have been flown on a single, much larger, and more expensive spacecraft [38].

A major hindrance for control of satellite constellations and formations largely consists of developing control schedules for a single satellite at a time by a mission planning team working on the ground. There is a growing desire to put hundreds and even thousands of satellites in a formation [39]. Future desires to have hundreds and even thousands of satellites in a constellation make the single-satellite planning approach infeasible. Satellites that have the ability to autonomously fly in formation, can make this future desire a reality.

2.2 Formation Flying Architectures

Their are several formation control approaches that exist in the literature. This section describes the different architectures, with the majority of the discussion being the architecture selected for this work, virtual structures. Due to its close relationship to the virtual structures architecture, the Leader/Follower approach is also discussed in detail in this section. Other architecture types found in the literature are only briefly mentioned, with a deeper discussion left to Appendix A. The Leader/Follower architecture is discussed first, followed by virtual structures, since the virtual structure approach builds upon the Leader/Follower framework.

2.2.1 Leader/Follower

The Leader/Follower architecture defines one spacecraft as the “Leader” and all other spacecraft are considered “Followers”. In some literature the Leader/Follower approach may also be referred to as Chief/Deputy [40], Master/Slave [41], or Target/Chaser [42]. The Leader/Follower architecture uses a hierarchical arrangement of the individual spacecraft controllers that reduces the formation control to an individual tracking problem. The leader regulates its position based on a desired goal or objective. The followers monitor the position of the leader and adjust accordingly to maintain their required relative location as prescribed by the desired formation. There are several variations with regard to a Leader/Follower architecture. A few examples are: assigning multiple leaders, forming a chain, or branch like topologies. A Leader/Follower architecture requires that the followers feed forward the acceleration of the leader. Feeding forward the leaders acceleration guarantees convergence of the follower spacecraft to the desired relative position [9]. One draw back to this approach that is worth noting is that it requires frequent communication of the acceleration data between the leaders and followers. The amount of data that needs to be shared may make this architecture unfeasible for some missions.

The key advantage to a Leader/Follower architecture is the reduction in coordination between the spacecraft with regard to the problems of regulations and tracking. Many tracking techniques have been adapted to Leader/Follower control, such as model predicative control to track the leaders motion [43]. Manikonda also applied feedback linearization and addressed saturation through controller switching [43]. Mesbahi and Hadeagh combined feedback linearization and linear matrix inequalities to design switch controllers to avoid saturation. They also used linear matrix inequality theory to derive strategies for tracking the motion of the leader, and allowing the leader and follower to change roles during a given moment in the mission. [44, 45].

A Leader/Follower architecture uses a control dependency directed graph approach. A directed graph is made up of a set of vertices V, that are connected by edges ε that have a direction associated with them, Figure 2.1. The edges are specified as ordered pairs (i, j), where $i, j \varepsilon V$. In the case of a set of spacecraft flying in formation, the spacecraft (vertex) i is required to maintain a desired position with respect to spacecraft (vertex) j [46] [45]. The arrows (Figure 2.1) show the control dependency, i.e. vertex "1" is not dependent on any other spacecraft and would be considered the leader, "2" is dependent only on "1" and "3" is dependent on "1" and "2".

There are three cases where an edges of the directed graph is added; 1) the control action of spacecraft j is a function of the state of spacecraft i, e.g. the spacecraft j tracks the relative state between i and j, 2) the reference trajectory of spacecraft j is dependent on the state of spacecraft i, or 3) the controller of spacecraft j has a dependency on the feedback control of spacecraft i. This third dependency has only been used in conjunction with the other two [45] . In a Leader/Follower architecture j would be considered the follower

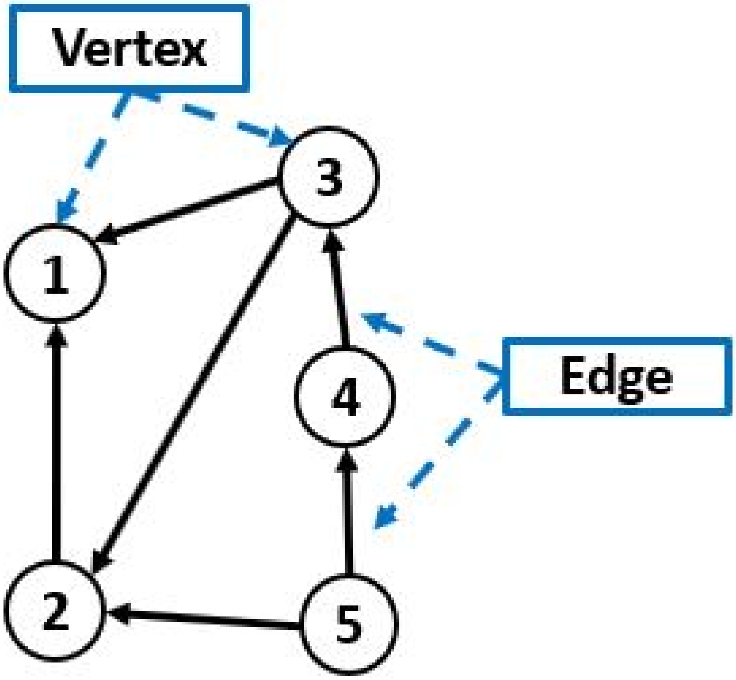


Fig. 2.1: Directed Graph Example

while i would be classified as the leader. In a Leader/Follower architecture there is at least one spacecraft (the leader) that does not have any dependencies on another spacecraft. A spacecraft may have more then one leader, but in these cases it is critical that the tracking problem is well understood [9, 11, 47]. A single-leader architecture is when each individual spacecraft only has a single leader [45].

With regard to directed graphs, a "path" is defined as a sequence of vertices such that each sequential pair is a directed edge, and the length of the path is defined by the number of vertices in it. A "cycle" is a path that contains at least three vertices and where none of the vertices are repeated. A directed graph is considered to be "acyclic" if it does not contain a cycle. A Leader/Follower architecture is considered to be an acyclic control dependency directed graph.

The Leader/Follower architecture was originally studied for applications tied to mobile robot control laws [48]. These same strategies were later applied to spacecraft formations by Wang and Hadaegh and applied to both deep space missions and earth centered missions [9]. Wang and Hadaegh developed control laws for formation maintenance and relative attitude alignment based on nearest neighbor tracking. They discussed several Leader/Follower techniques; leader tracking, nearest neighbor tracking, barycenter tracking, and other tree topologies [49]. They later built upon these ideas to include the implications due to actuator saturation. An adaptive version of this architecture was later added that evaluated the ability to reject space disturbances [50]. They showed that if the tracking control law is stable for each of the followers then the entire formation is stable. They did not address general sufficient conditions for Leader/Follower stability. They also simplified the feedback linearized control laws from [9] and applied them to synchronized transnational and rotational control of deep space missions [49]. [51] also looked at simultaneous rotational and transnational control formation, with the modification of performing the control in the leaders body frame as opposed to an inertial frame. This allows for the whole formation to be rotated by only rotating the attitude of the leader. Further research was done to develop a rule-based control law for synchronizing the rotations of multiple spinning spacecraft [52]. Hadaegh then developed a rule-based controller for synchronizing thruster dead-bands across multiple spacecraft. They showed that it is necessary to synchronize impulsive thrusters for MSIs. They found that vibration from the thruster firings can corrupt the interferometric measurements [53]. Lurie designed a thruster synchronized algorithm for rotational and transnational control using classical control theory with nonlinear dynamic compensation [54]. A sliding mode control was implemented by [55]. LQR and H_∞ methods were applied to a combined translation and rotational controller by [56]. Robertson developed and compared several proportional derivative controllers. He also looked and time optimal control and a mixed fuel-time optimal control [57, 58]. Olfati-Saber and Murray used “node augmentation” to build a mission planner. A user is able to add spacecraft to the simulation formation sequentially through specifying the required distance to two other spacecraft in the formation. A feedback-linearized controller is used to maintain these distances. The controller of the new spacecraft is only dependent on the relative states of the two spacecraft it is anchored to [59]. Desai presents a similar method [46].

Advantages/Disadvantages to the Leader/Follower Architecture: The Leader/Follower architecture reduces the data needs when compared to other architectures, since each spacecraft only needs to track the information of the leader, instead of tracking the entire formation. Because of this, formation coordination is typically simpler then other architectures. A Formation can easily grow or shrink in size since only a locally stabilizing controller and a leader assignment is required. The formation itself is easily specified or changed by commanding the leader. Leader/Follower architecture is also more robust against failures, when compared to other architectures. If a spacecraft fails, then only the followers of that spacecraft are affected. These followers can easily be reassigned to a new leader, reducing the effects to the overall system. For a Leader/Follower architecture optimality is not guaranteed. Connecting individual locally optimal tracking controllers does not guarantee global optimality for the entire formation. Also, if a mesh stable system is required then the data requirements can start to be significant, similar to the requirements of a MIMO architecture [11].

2.2.2 Virtual Structures

In a virtual structure architecture the formation is treated as a ridged body. The spacecraft act as ridged bodies within the larger virtual structure formation. A virtual structure architecture is defined by states that correspond to the dynamics of the formation, for example the dynamics can be made up by the position, orientation, and/or separation distances between spacecraft. These states are allowed to change over time to map the desired trajectories of the formation or each individual spacecraft within the formation. The motion of the virtual structure and the motion of the dynamics within the virtual structure are used to generate trajectories for the formation to follow. In a virtual structure the controller of each individual spacecraft is responsible for generating and tracking a reference trajectory, or virtual leader [11, 60]. In its most basic form the virtual structure architecture is identical to the Leader/Follower architecture. In that if the virtual structure was to simply trace out the position and orientation of a given point within the formation, this would be identical to a Leader/Follower architecture, the only difference would be that their is no actual leader, but merely a *pseudo leader* that would reside only in simulation. It is worth noting that since the virtual structure architecture is similar to the Leader/Follower architecture, the analysis to show convergence is also identical.

Virtual structures architectures are commonly applied to mobile robots and unmanned aerial vehicles (UAV) applications [61–66]. A key advantage to a virtual structure architecture is that the trajectory information of the virtual leader is determined a priori by each of the spacecraft and does not need to be shared, reducing communication loads. Also, since the individual spacecraft follow a reference trajectory, an anomaly of one spacecraft would not impact the entire formation.

The virtual structures architecture enables each individual vehicle to plan maneuvers within a given operational envelope, allowing the planning of the formation to be decomposed from the planning of the individual. Virtual structures enables safe and reliable formation planning. Individual spacecraft exploit the structure to create guaranteed collision-free maneuvers.

One drawback to virtual structures, like Leader/Follower is the requirement to share acceleration information between spacecraft, this can potentially be a costly requirement leveraged on the communication and processing systems of the formation. Because of this, the trajectory of the virtual leader is traditionally centralized, and communicated out to the other members of the formation. Lewis and Tan applied this approach to the formations of mobile robots [66]. Applications to spacecraft formations were described in [67–70]. Tillerson, Berger, and How presented a procedure for calculating a *virtual center* for the formation, that is used to determine the desired states for each of the spacecraft in the formation [71]. Ren and Beard developed a decentralized approach that can be appropriate when a large number of spacecraft are involved in the formation [72].

A less traditional approach to virtual structures that has advantages is to allow each member of the formation to propagate the virtual leaders dynamic motion, thus not limiting the virtual structures to have a centralized virtual leader. A periodic comparison of the virtual leaders trajectory shared between spacecraft could allow for any necessary corrections to be made.

A key advantage to a virtual structure architecture is that the trajectory of the formation can be shaped. In a Leader/Follower architecture the leader's position and attitude move independently. If spacecraft alignment is required, the spacecraft attitude and position must be coupled. A virtual structure architecture allows for the virtual leader's position and attitude motion to be coupled. Each individual spacecraft propagates this coupled motion, thus perfect tracking can guarantee spacecraft alignment. Virtual structures also allow for feedback to the formation trajectory as a function of the current spacecraft states, which allows the formation to not *fall behind*, [66,73], since more current information is available. This can also lead to more up to date health and status of the formation vehicles including fuel consumption [60,74].

For this work a virtual leader is used within the virtual structure to design a reference orbit for each satellite in the formation. From the virtual leader, relative orbits are generated that define the nominal locations of each of the agents within the virtual structure. This work uses two different sets of orbital relative dynamics, the HCW equations and the D'Amico ROEs [42,75–77]. Figure 2.2 gives an example of a virtual structure formation.

2.2.3 Other Architectures

The other architectures studied as part of this research are now briefly presented.

Cyclic

A Cyclic architecture is similar to a Leader/Follower architecture, in that this architecture type consists of connecting individual spacecraft controllers. The difference lies in the fact that for a Cyclic architecture the controller connections are not hierarchical. A Cyclic formation flying control architecture is defined by an interconnection of individual spacecraft controllers that result in a cyclic control dependency directed graph [11]. The

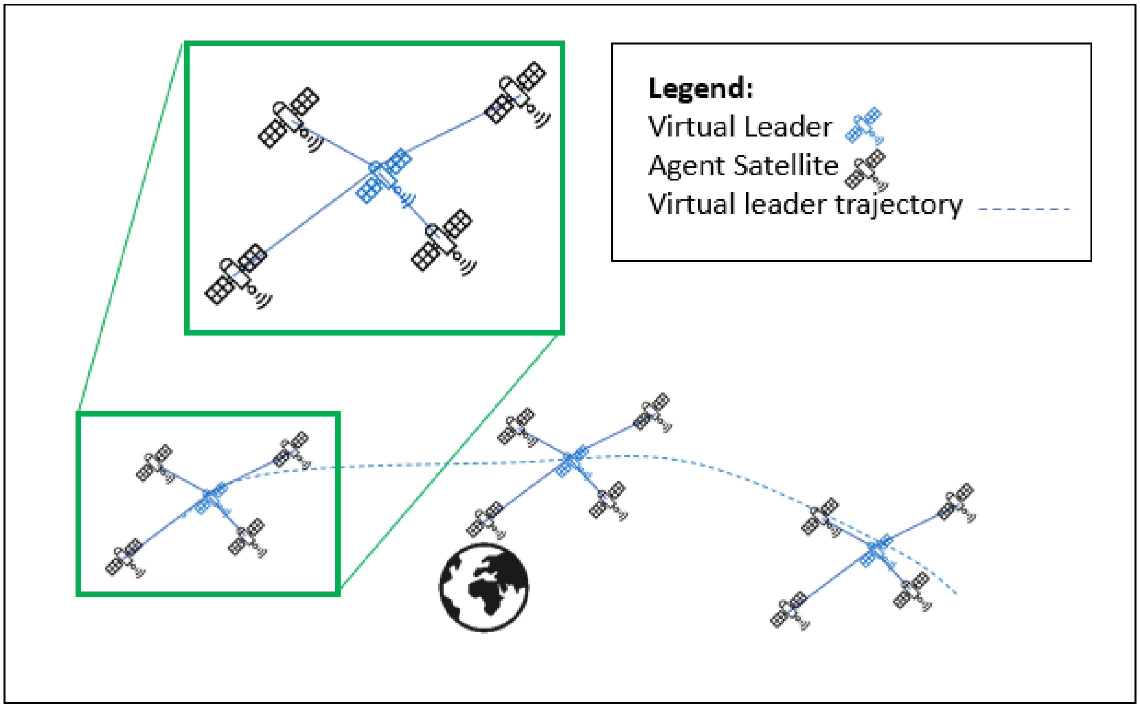


Fig. 2.2: Example virtual structure formation.

dependency graph for a Cyclic architecture is completely connected, i.e. every spacecraft depends on ever other spacecraft [11].

Behavioral

A behavioral architecture is one where each spacecraft has a control law with built in specific objectives or “behaviors”. A behavioral architecture combines the outputs of multiple controllers designed for achieving different and sometimes competing behaviors [70,78]. Examples of possible behaviors are orbital maintenance, formation keeping, collision avoidance, or move-to-goal. These behaviors can either be accomplished individually or collectively.

2.3 Formation Flying Challenges

There are several challenges that arise in formation flying. These challenges include, but are not limited to: fuel consumption, overcoming disturbances or perturbing forces, data distribution between spacecraft, collision avoidance, mission operational constraints, and computation feasibility. Creating a formation flying framework becomes an act of balancing priorities between meeting these different challenges or objectives.

A key constraint to any formation flying mission is the amount of spacecraft mass that can be allocated to the propulsion system. The amount of fuel mass available directly impacts the number of maneuvers that can be performed and ultimately the lifetime of the mission. Over the last few decades, lowering fuel consumption for formation flying missions has been an area of interest for several researchers. Previously, optimization strategies have been employed to reduce fuel consumption while maintaining relative motion. Mazal and Gurfil developed formation keeping algorithms that aimed to reduce fuel consumption by using bi-impulsive maneuvers that enable desegregated satellite modules to attain relative states satisfying both maximum and minimum distance keeping constraints while tracking a desired reference orbit [79]. Rocco et al., studied low fuel consumption techniques where the maneuver time was restricted [22]. Lim et al., [24], similarly developed an MPC approach based on D'Amico's eccentricity and inclination vectors to describe the relative motion. Tillerson et al., [25], used linear programming (LP) to develop fuel-optimal control inputs to move a satellite from the disturbed state back to the desired state or to maintain the satellite within some tolerance of the desired state. DiMauro also looked at optimizing fuel usage of a formation through the formulation of a mixed integer control problem [26].

While an optimal fuel saving maneuver may be desirable, feedback is essential for overcoming disturbances from the erroneous motion models. Model predictive control is an approach that repeatedly uses optimal control techniques within a feedback loop [27]. Several papers use some variant of a linear time-varying MPC [23, 25, 80–85]. Tillerson et al., [25], designed an MPC scheme that used the HCW and Lawden equations to define the relative motion. Breger and How, [23], used an MPC approach that was based on Gauss' Variational Elements (GVEs) to describe the relative motion.

In conjunction with MPC, several relative dynamic sets have been used in the literature for formation control, as knowledge of the relative states of the formation is often more accurate than knowledge of the formation's absolute state. Generally, the control of the relative state is more important than the control of the absolute state. In [23], an MPC scheme was designed using GVEs to describe the relative motion between vehicles. Similarly, [24] developed an MPC strategy using GVEs with constraints defined by maintaining the eccentricity and inclination vectors of the D'Amico ROEs parallel to each other. GVEs are convenient for specifying and controlling formations when the distances between spacecraft are larger than 1000 km because they are linearized about the orbital elements. The orbital elements are described in a curvilinear frame in which large rectilinear distances can be captured by small element perturbations. However, the use of GVE dynamics can induce error at smaller ranges and incur an additional computation cost when compared to the HCW equations. Sevearl researchers have used the HCW equations to describe the relative orbital dynamic equations [25,29–32] and were able to maintain control of a spacecraft within their desired tolerances. The HCW dynamics are attractive to use with MPC because they are a well understood, linear, time-invariant model. Additionally, the use of Cartesian coordinates provides an intuitive understanding of the spacecraft states.

Another challenge that needs to be addressed in the area of spacecraft formation flight is data distribution between satellites. Inter-satellite communication can become a restrictive burden on satellites as it requires the antenna of each satellite to be aligned and the communication systems powered on. To reduce communication, [86–88] developed a strategy for using triggering events to switch from a coasting behavior to a controlled behavior, requiring communication only when vehicles pass some relative distance threshold. To lower communication even further, virtual structure techniques define the desired motion for each satellite based upon the ideal motion of a virtual satellite [23, 25]. The virtual satellite is a computational artifact used solely to define desired relative trajectories for the constellation [72]. As previously mentioned, a key advantage to a virtual structure architecture is that the trajectory information of the virtual leader is determined a priori by each of the spacecraft and does not need to be shared, reducing communication loads. Also, since the individual spacecraft follow a reference trajectory, an anomaly of one spacecraft would not impact the entire formation.

A virtual structures architecture also enables each individual vehicle to plan maneuvers within a given operational envelope, allowing the planning of the formation to be decomposed from the planning of the individual. Virtual structures enables safe and reliable formation planning. Individual spacecraft exploit the structure to create guaranteed collision-free maneuvers. Due to perturbing forces there will be deviations from the desired trajectory. These deviations must be bounded so that the vehicles do not collide with one another. Lim et al., [24], guaranteed collision avoidance by developing a framework that maintained the eccentric and inclination vectors of the D'Amico ROEs parallel to each other. Maintaining the ROE eccentricity and inclination vectors parallel to each other guarantees collision avoidance within an established formation, but it does not guarantee a collision free transfer trajectory nor does it guarantee the formation itself is maintained. Tillerson et al., [25], and Breger, [23] used a six-sided box as a keep-in volume, with a linear program and no mixed integers, but postulated that using a higher dimensional shape could reduce fuel costs. DiMauro et al., [26], also used a six-sided box but as a keep-out volume for the deputy spacecraft, but DiMauro et al., found the their MILP approach increased the computation time to a degree that it was deemed not suitable for flight.

This work develops a framework to address various aspects of these challenges through a decentralized, optimization-based control that can be used for autonomous formation flight of small satellites. To accomplish this goal, this work develops a formation flying control algorithm using a virtual structure, MPC architecture where the HCW equations and D'Amico ROEs are used to describe the relative motion.

2.4 Contributions

The major contributions of this work are the formulation and evaluation a decentralized, optimization-based control that can be used for autonomous formation flight of small satellites. This is accomplished through a virtual structure based MPC framework that guarantees flight safety and formation control while saving fuel and reducing communication. The formulation includes the use of the HCW equations to define the relative motion, and the development of a polytope sphere approximation that is used as the switching surface. A comparison of the computation time and fuel usage for L_1, L_2, and L_∞ norm formulations of the framework is also performed. These details are presented in Chapter 4. An optimized guidance trajectory is then added to the framework in Chapter 5. Adding this guidance trajectory to the framework proves to reduces the ΔV required by almost two orders of magnitude.

While the HCW equations are attractive to use because they are a well understood, linear, time-invariant model and provide an intuitive understanding of the spacecraft states, using Cartesian coordinates, or the HCW equations, to define a spacecraft's state can result in some complications. For instance, given a Cartesian state, it is generally non-trivial to determine the full relative orbit of the spacecraft or attempt to match some orbital element of a second spacecraft. Additionally, using the standard Euclidean definition of distance to define the distance between the actual and desired spacecraft states may not produce the most meaningful results. Consider, for example, a spacecraft whose position perfectly matches some desired location but has a velocity perpendicular to the desired velocity. The distance would be considered "close" in a Euclidean sense, but because of the direction of the velocity vector, these orbits are significantly different. This can cause the controller to erroneously turn off if the constraints are based solely on a Euclidean distance. While an HCW based architecture was developed and tested as part of this work, the difficulty of matching the full relative orbital elements and velocities lead to the utilization of D'Amico's ROEs. Thus, this work also developments and implementations an optimal guidance trajecotry and model predictive control framework where the switching surfaces and relative dynamics are based on the D'Amico ROEs (Chapter 6). Using an MPC and switching surface based on the ROEs accounts for the interplay between the control inputs and the natural dynamic motion of the system, that further reduce fuel usage when compared to a framework based on the HCW equations. Modeling the constraints and the control in ROE spaces allows the system to better match the desired orbit, not just a position state at a given time. As shown in Chapter 6, the ROE approach requires only

33% of the fuel required by the HCW based architecture.

CHAPTER 3
BACKGROUND

The purpose of this chapter is to familiarize the reader with several of the key elements that apply to this research, what foundations it builds on, and the approaches that exist within the field, namely, background information on the relative dynamics and model predictive control.

3.1 Relative Dynamics

For this work a virtual leader is used to design a reference orbit for each satellite in the formation. From the virtual leader, relative orbits are generated that define the nominal locations of each of the agents within the virtual structure. The relative orbits are propagated using two different approaches, the HCW equations and the D'Amico ROEs. [42, 75–77].

3.1.1 Hill-Clohessy-Wiltshire Equations

The HCW equations were "first" published in 1960 by Clohessy and Wiltshire [42, 76], but it was later discovered that these equations were just another form of the previously described equations produced by Hill in 1877 [75]. The HCW equations linearize the two-body gravitational dynamic equations about a near circular orbit. An exact solution of the HCW equations can be represented as:

$$\begin{aligned} \ddot{x} - 3n_c^2 x - 2n_c\dot{y} &= u_x \\ \ddot{y} + 2n_c\dot{x} &= u_y \\ \ddot{z} + n_c^2 z &= u_z \end{aligned} \tag{3.1}$$

where x, y, and z represent the relative position of the follower with respect to the leader in Cartesian coordinates using a Local Vertical, Local Horizontal (LVLH) frame; $\dot{x}$, $\dot{y}$, and

$\dot{z}$ represent the relative velocities; $\ddot{x}$, $\ddot{y}$, and $\ddot{z}$ represent the relative accelerations; and n_c is the mean motion, or average angular velocity, of the reference orbit. The LVLH frame is defined such that x is in the radial direction, z is along the angular momentum vector, and y satisfies the right-hand rule.

If the state is defined as $\mathbf{x} = [x\,y\,z\,\dot{x}\,\dot{y}\,\dot{z}]^T$ and the control as $\mathbf{u} = [u_x\,u_y\,u_z]^T$, Equation 3.1 can be written in the state space form

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{3.2}$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3n_c^2 & 0 & 0 & 0 & 2n_c & 0 \\ 0 & 0 & 0 & -2n_c & 0 & 0 \\ 0 & 0 & -n_c^2 & 0 & 0 & 0 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3.3}$$

An exact discrete solution can be found as

$$\mathbf{x}_{k+1} = \mathbf{A}_D\mathbf{x}_k + \mathbf{B}_D\mathbf{u}_k, \quad \mathbf{A}_D = e^{\mathbf{A}\Delta t}, \quad \mathbf{B}_D = \left(\int_0^{\Delta t} e^{\mathbf{A}(\Delta t-\tau)}\, d\tau\right)\mathbf{B}. \tag{3.4}$$

The resulting discretized HCW matrices are

$$\mathbf{A}_D = \begin{bmatrix} 4-3\cos n_c\Delta t & 0 & 0 & \frac{\sin n_c\Delta t}{n_c} & \frac{2-2\cos n_c\Delta t}{n} & 0 \\ 6(\sin n_c\Delta t - n_c\Delta t) & 1 & 0 & \frac{2\cos n_c\Delta t-2}{n} & \frac{4\sin n_c\Delta t-3n_c\Delta t}{n} & 0 \\ 0 & 0 & \cos n_c\Delta t & 0 & 0 & \frac{\sin n_c\Delta t}{n_c} \\ 3n_c\sin n_c\Delta t & 0 & 0 & \cos n_c\Delta t & 2\sin n_c\Delta t & 0 \\ 6n_c(\cos n_c\Delta t - 1) & 0 & 0 & -2\sin n_c\Delta t & 4\cos n_c\Delta t - 3 & 0 \\ 0 & 0 & -n_c\sin n_c\Delta t & 0 & 0 & \cos n_c\Delta t \end{bmatrix} \tag{3.5}$$

$$\mathbf{B}_D = \begin{bmatrix} \frac{2}{n_c^2}\sin^2\frac{n_c\Delta t}{2} & -\frac{2}{n_c^2}(\sin n_c\Delta t - n_c\Delta t) & 0 \\ \frac{2}{n_c^2}(\sin n_c\Delta t - n_c\Delta t) & -\frac{1}{2n_c^2}(8\cos n_c\Delta t + 3n_c^2\Delta t^2 - 8) & 0 \\ 0 & 0 & \frac{2}{n_c^2}\sin^2\frac{n_c\Delta t}{2} \\ \frac{1}{n_c}\sin n_c\Delta t & -\frac{1}{n_c}(2\cos n_c\Delta t - 2) & 0 \\ -\frac{4}{n_c}\sin^2\frac{n_c\Delta t}{2} & \frac{4}{n_c}\sin n_c\Delta t - 3\Delta t & 0 \\ 0 & 0 & \frac{1}{n_c}\sin n_c\Delta t \end{bmatrix} \tag{3.6}$$

Note that $\mathbf{A}_D$ matches the discrete transition matrix in [42] while $\mathbf{B}_D$ is derived using Equations 3.3 and 3.4. A full derivation of the HCW equations can be found in Appendix B.

3.1.2 Relative Orbital Elements

Traditionally, the HCW equations have been used to describe the relative orbital dynamics of spacecraft, [75,76]. More recently, however, relative orbital elements (ROEs) have been developed to describe relative dynamics in the context of Keplerian orbital elements. These are a set of states that define the orbit of a spacecraft relative to some reference orbit. Various ROE formulations have been proposed, notably a set created by Lovell and Spencer [89] and a set created by D'Amico [77]. For this work the D'Amico ROEs were chosen as they are derived directly from the standard orbital elements and were more amenable for the desired formation flying architecture. The ROEs assume two-body motion with no perturbations, but can be adjusted to account for J_2 disturbances [90]. Although the ROE formulation used in this paper does not include the J_2 formulation, the controller in this work does account for, and overcome J_2 perturbations.

Similar to Keplerian orbital elements, ROEs use six parameters to define the orbit of an agent spacecraft relative to the virtual leader. The elements describe the relative semi-major axis, mean latitude, a two element eccentricity vector, and a two element inclination vector of the agent, relative to the virtual leader. Allowing for a range of values for each of these elements creates a three-dimensional region that is used to define the relative space for the agent.

Given a set of standard orbital elements for the chief or reference orbit and a second set of standard orbital elements of a deputy or agent satellite, the ROEs, which describe the orbit of the deputy relative to the chief, can be determined. In the ROE states, the c and d subscripts indicate the elements of the chief and deputy, respectively. The orbital elements a, e, i, ω, Ω, and M are the semi-major axis, eccentricity, inclination, argument of perigee, right ascension of ascending node, and mean anomaly, respectively, [42]. The dimensionless ROEs, as defined by D'Amico [77], are given as

$$\mathbf{x} = \begin{bmatrix} \delta a \\ \delta\lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \end{bmatrix} = \begin{bmatrix} (a_d - a_c)/a_c \\ (f_d - f_c) + (\Omega_d - \Omega)\cos i_c \\ e_{x,d} - e_{x,c} \\ e_{y,d} - e_{y,c} \\ i_d - i_c \\ (\Omega_d - \Omega_c)\sin i_c \end{bmatrix}, \tag{3.7}$$

with

$$f_\square = \omega_\square + M_\square\ ,\ e_{x,\square} = e_\square \cos\omega_\square\ ,\ e_{y,\square} = e_\square \sin\omega_\square\ , \tag{3.8}$$

where the □ subscripts indicate the states correspond to either the deputy or chief states.

The δa term represents the relative difference in semi-major axis and the $\delta\lambda$ term gives the relative mean longitude between the two spacecraft. The δe_x and δe_y terms can be composed into a single $\delta\mathbf{e}$ relative eccentricity vector where the magnitude gives the orbit size along the radial direction. Similarly, the δi_x and δi_y terms can create the $\delta\mathbf{i}$ relative inclination vector where the magnitude defines the size of the orbit along the normal axis. The $\delta\mathbf{e}$ and $\delta\mathbf{i}$ vectors can additionally be used to create passively safe orbits where the agent does not pass through the relative orbital plane in front of the leader [24]. Note that this relative state representation is nonsingular for circular orbits ($e_c = 0$), whereas it is singular for strictly equatorial orbits ($i_c = 0$) [26].

Assuming two body motion with no perturbations, the orbital elements defined in Eq. (3.8) are all constant except the mean argument of latitude, f, which increases at a constant rate, with μ representing the standard gravitation parameter, defined by

$$\dot{f} = \frac{df}{dt} = \sqrt{\frac{\mu}{a^3}}. \tag{3.9}$$

Defining $\Delta f = f_d - f_c$, then $\Delta\dot{f}$ represents the drift in relative mean argument of latitude that will occur when the spacecraft has a different semi-major axis than the reference orbit, and we define $\Delta a = a_d - a_c$. This can be approximated to the first order by the differencing of Eq. (3.9) for the two orbits as

$$\Delta\dot{f} = \frac{d(\Delta f)}{dt} = -\frac{3}{2}\sqrt{\frac{\mu}{a^5}}\Delta a = -\frac{3}{2}n\frac{\Delta a}{a} = -\frac{3}{2}n\delta a, \tag{3.10}$$

where n is the mean orbit motion and with the reasonable assumption that Δf and Δa are small compared to the inertial virtual leader orbit radius [77]. Assuming two body motion, the remainder of the inertial orbital elements are constant over time, and as such the other ROEs are also constant with respect to time. The full dynamics can then be represented as

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{3n}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta a \\ \delta\lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \end{bmatrix} = \mathbf{A}\mathbf{x}. \tag{3.11}$$

If f_M represents the mean argument of latitude of any given maneuver and the prefix Δ indicates the change in the specified ROE, the instantaneous changes of the ROEs ($\Delta\delta a$, $\Delta\delta\lambda$, $\Delta\delta e_x$, $\Delta\delta e_y$, $\Delta\delta i_x$, $\Delta\delta i_y$) as a result of impulsive ΔV applied along the tangential, radial, and normal axes (δv_t, δv_r, and δv_n, respectively) is [77],

$$\begin{array}{lclll}
\Delta\delta a & = & & +\dfrac{2}{na}\delta v_t & \\
\Delta\delta\lambda & = & -\dfrac{2}{na}\delta v_r & & \\
\Delta\delta e_x & = & \dfrac{\sin f_M}{na}\delta v_r & +\dfrac{2\cos f_M}{na}\delta v_t & \\
\Delta\delta e_y & = & -\dfrac{\cos f_M}{na}\delta v_r & +\dfrac{2\sin f_M}{na}\delta v_t & \\
\Delta\delta i_x & = & & & +\dfrac{\cos f_M}{na}\delta v_n \\
\Delta\delta i_y & = & & & +\dfrac{\sin f_M}{na}\delta v_n
\end{array} . \tag{3.12}$$

as can be seen, these are linear, time-varying

As previously mentioned, Lovell and Spencer also developed a formulation of ROEs [89], for this work it was determined to use D'Amico's ROE formulation due to their ability to include secular effects of dominant differential perturbations in closed-form, the fact that they are not restricted to near-circular orbits, and their ability to be directly related to the inertial orbital elements for both the virtual leader and agent spacecraft.

3.2 Model Predictive Control

As discussed in Chapter 2, an MPC scheme is implemented in conjunction with the relative dynamics described in Section 3.1. This section presents a brief primer on MPC to present the basic idea and notation used throughout this work. A general overview of MPC with a linear system representation and a discussion of the closed-loop stability. The interested reader is referred to [91] for a thorough development of MPC for linear systems.

3.2.1 General MPC Overview

There are two problems that often occur when using an optimization-based control solution in practice. First, even with optimization algorithms taking advantage of linearities and convexities, a horizon time that is sufficiently large enough to produce desirable convergence characteristics may prove computationally prohibitive. Second, the model of the system is usually inaccurate and the system may be impacted by external disturbances that can cause it to diverge from the predicted path. For this reason, the state is also measured at x_{k+1} and the control problem is re-solved, where the measured state $x(1)$ is considered the new initial condition. This process is referred to as MPC.

While MPC generally loses any guarantee of optimality, it does provide the ability to express constraints, which is not common in many feedback control solutions. Furthermore, weights in the objective can provide intuitive "control knobs" for tuning to the desired behavior.

Common objective functions for linear systems include the L$_2$, L$_1$, and L$_\infty$ norms, as they can result in quadratic and linear programs. Given an initial state of x_0, and a time horizon of N intervals, the objective function can be written as:

$$J(x,u)_p = \frac{1}{2}||u_0||_{R,p} + \frac{1}{2}\sum_{k=1}^{N-1}\Big[||x_{e,k}||_{Q,p} + ||u_k||_{R,p}\Big] + \frac{1}{2}||x_{e,N}||_{P,p} \tag{3.13}$$

where R, Q, and P are the weightings on control usage, state error, and terminal error, respectively. And p represents the norm $1, 2$, or ∞.

3.2.2 MPC Stability

For a receding horizon control (RHC) problem, when an optimization problem is solved over a finite horizon repeatedly at each time step, the hope is that the controller results lead to a closed-loop behavior that mimics that of the infinite horizon controller. The RHC is preferred over the infinite horizon control, because constraints can be added to the control formulation.

An issue that arises with the RHC is that the generated control inputs may not lead to trajectories that converge to the origin, i.e. asymptotically stable. In general, stability is not guaranteed by the receding horizon optimization, but is essential for developing a quality control law. To guarantee stability, the terminal cost and constraint set must be carefully chosen [91].

For the RHC implementation, consider the discrete-time linear time-invariant system

$$
\begin{aligned}
&x(t+1) = Ax(t) + Bu(t) \\
&\text{s.t. } x(t) \in X, u(t) \in U, \forall t \geq 0,
\end{aligned}
\tag{3.14}
$$

where $x(t) \in \mathbb{R}^n, u(t) \in \mathbb{R}^m$, and the sets $X \subseteq \mathbb{R}^n$ and $U \subseteq \mathbb{R}^m$. If we assume that a measurement of the state $x(t)$ is available at the current time t, then the finite time optimal control problem can be written as

$$
\begin{aligned}
J_0^*(x(t)) = \min_{U_0} \quad & J_0(x(t), U_0) = p(x_N) + \sum_{k=0}^{N-1} q(x_k, u_k) & \\
\text{s.t.} \quad & x_{k+1} = Ax_k + Bu_k & k = 0, 1, \ldots, N-1 \\
& x_k \in X, u_k \in U & k = 0, 1, \ldots, N-1 \\
& x_N \in X_f & \\
& x_0 = x(t) &
\end{aligned}
\tag{3.15}
$$

where $U_0 = \{u_0, \ldots, u_{N-1}\}$. If we set $p(x_N) = ||Px_N||_p$ and $q(x_k, u_k) = ||Qx_k||_p + ||Ru_k||_p$, given an initial state of x_0, a time horizon of N intervals, and if we let $p = 1$, $p = 2$ or $p = \infty$ for the L_1, L_2, and L_∞ norms, respectively, then the cost function can be written as

$$
J_0(x(0), U_0) = ||Px_N||_p + \sum_{k=0}^{N-1} ||Qx_k||_p + ||Ru_k||_p, \tag{3.16}
$$

where the control law is

$$
u_k(t) = f_0(x(t)) = u_0^*(x(t)), \tag{3.17}
$$

for the closed loop system

$$
x(k+1) = Ax(k) + Bf_0(x(k)) = f_{cl}(x(k)), k \geq 0. \tag{3.18}
$$

Given Q and R, the terminal cost of the system can be designed to ensure asymptotic stability of the MPC problem. The objective is to find a Lyapunov function for the closed-loop system. If the terminal cost and constraint are appropriately chosen, then the value function $J_0^*(\cdot)$ is a Lyapunov function. The theorem below was developed by Borrelli and proofs of this theorem can be found in [91]. If the closed loop system meets these conditions, the system is an asymptotically stable system.

Theorem 3.2.1 *Consider system 3.17, the RHC law (3.15-3.17) and the closed-loop system (3.18). Assume that*

- *The stage cost q(x, u) and terminal cost p(x) are continuous and positive definite functions.*

- *The sets X, X_f and U contain the origin in their interior and are closed.*

- *X_f is control invariant, $X_f \subseteq X$.*

- $\min_{u\,\in\,U,\;\; Ax+Bu\,\in\,X_f} (-p(x) + q(x,u) + p(Ax+Bu)) \leq 0, \;\; \forall x \in X_f.$

Then, the origin of the closed-loop system (3.18) is asymptotically stable with domain of attraction X_0

The following sections present how this theorem can be met for the 2, 1, and ∞ norm cases.

Stability for 2-Norm Case: For the case of the L_2 norm, the terminal cost is chosen to represent the cost-to-go of the unconstrained infinite horizon problem. The Discrete-time Algebraic Riccati Equation (DARE) can be used to find the cost-to-go, $X^T P_\infty X$. The DARE is represented as:

$$0 = A'P_\infty A - P_\infty + Q - A'P_\infty B(B'P_\infty B + R)^{-1}B'P_\infty A \tag{3.19}$$

where the positive definite solution for P_∞ is used.

The P_∞ found solving the DARE is then used as the matrix in the terminal cost, P, in the objective function, equation 3.13. This guarantees asymptotic stability of the system for the L_2 norm formulation [91]. Note that A and B must be completely controllable, R must be positive definite, and Q must be positive semi-definite and completely observable.

Stability for 1 and ∞-Norm Cases: For the L$_1$ and L$_\infty$ cases, if the exact discrete system (Equation 3.14) is asymptotically stable, then X_f can be chosen as the positively invariant set of the autonomous system

$$x(k+1) = Ax(k) \quad \text{s.t.} \quad x \in X \tag{3.20}$$

Therefore, the input 0 is feasible in X_f and the Lyapunov inequality in Theorem 3.2.1 for the L$_\infty$ and L$_1$ cases becomes

$$|| \, Px \, ||_p + || \, PAx \, ||_p + || \, Qx \, ||_p \leq 0, \;\; \forall x \in X_f \tag{3.21}$$

where $p = 1$ or $p = \infty$ for the L$_1$ and L$_\infty$ norms, respectively. Equation 3.21 is satisfied if a P is chosen that satisfies the Lyapunov function for the L$_1$ and L$_\infty$ cases, equation 3.22.

$$V = || \, Px \, ||_p \tag{3.22}$$

The P_p found solving equation 3.22 is then used in the objective function, equation 3.16. This guarantees asymptotic stability of the system [91].

If such a X_f can be computed, it can be used as a terminal constraint in Theorem 3.2.1. With this choice, equation 3.23 is satisfied by the infinite time unconstrained optimal cost matrix P_∞ in equation 3.24, the optimal infinite horizon cost function.

$$\min_{u \in U, \;\; Ax+Bu \in X_f} (p(x) + q(x,u) + p(Ax+Bu) \leq 0, \;\; \forall x \in X_f \tag{3.23}$$

$$J^*_\infty(x(0)) = || \, P_\infty x(0) \, ||_\infty \tag{3.24}$$

CHAPTER 4

Model Predictive Control Switching Strategy for Safe Small Satellite Cluster Formation Flight

Aerospace Systems
https://doi.org/10.1007/s42401-023-00237-2

REVIEW

Model predictive control switching strategy for safe small satellite cluster formation flight

Tyson Smith[1] · John Akagi[1] · Greg Droge[2]

Received: 8 March 2023 / Revised: 31 May 2023 / Accepted: 14 June 2023

Abstract

This paper presents the development and analysis of a spacecraft formation flying architecture. The desired state of each spacecraft is maintained using a model predictive control-based control framework that is based on the Hill–Clohessy–Wiltshire equations and a polytope boundary constraint as a switching surface. This framework can be used to maintain the desired cluster formation while also guaranteeing internal cluster flight. The polytope boundaries are designed, such that no two agents have overlapping regions, allowing the vehicles to execute avoidance strategies without continually maintaining the trajectories of other agents. The model predictive control framework combined with the convex polytope boundary enables a scalable method that can support clusters of satellites to coordinate to safely achieve mission objectives while minimizing fuel usage. As part of the implementation of this control scheme, the authors created two spacecraft formation flying control approaches. The first approach uses fewer, large maneuvers to control a spacecraft to the center of a keep-in-volume. The second approach allows the spacecraft to perform many small maneuvers to stay just inside the boundary of the keep-in-volume. This paper compares the fuel cost savings of these two approaches. The results presented in this paper demonstrate that the first approach produces the lower total fuel usage, but if a lower amount of fuel per maneuver is required, then the second approach should be used. This work also compares the computation requirements and fuel usage for L_1, L_2, and L_∞ norms formulations of the framework, the L_1 and L_2 norms require the least amount of fuel usage, while the L_2 requires the least amount of computation time.

Keywords Formation flying · Model predictive control · Switching surfaces

1 Introduction

Spacecraft formation flying has been viewed as a key enabler for organizations like the European Space Agency (ESA) [1, 2], National Aeronautics and Space Administration (NASA) [1, 3, 4], and the Department of Defense (DoD) [5]. Formation flying technology is critical for a wide range of missions, including sparse antenna array missions, distributed sensing missions, advanced communications and internet access, spatial sampling for applications such as gravitational mapping, interferometric synthetic aperture radar, and target tracking, to name just a few [1, 6–16]. The idea of formation flying also allows for many lower cost spacecraft to fly multiple different payloads that traditionally would have been flown on a single larger and more expensive spacecraft [17]. Fundamental to space-based operations is the requirement that vehicles limit their fuel usage to increase longevity of their missions. Formations add the additional complexities of maintaining relative motion characteristics which require incorporating feedback for minor and major adjustments. This must be done while respecting the computational capabilities of the individual platforms and the communication restrictions within the constellation. This work develops an optimization-based feedback control strategy to allow each satellite to fly within a safe region of operation and maintain relative spacing with little-to-no inter-spacecraft communication required. Three different optimization objectives and

Tyson Smith
tyson.smith@sdl.usu.edu

John Akagi
john.akagi@sdl.usu.edu

Greg Droge
greg.droge@usu.edu

[1] GN&C Engineer, Space Dynamics Laboratory, Logan, UT 84341, USA

[2] Electrical and Computer Engineering Department, Utah State University, Logan, UT 84322, USA

Published online: 26 July 2023

two strategies for limiting thruster use are evaluated with respect to overall fuel usage and computation requirements.

Previous work in the field of formation flying has developed techniques that employ optimization strategies to reduce fuel usage while still maintaining relative motion, e.g., [15, 18–33]. While an optimal fuel saving maneuver may be desirable, feedback is essential for overcoming disturbances from the invariably erroneous motion models. Model predictive control (MPC) is a technique for using optimal control techniques within a feedback loop [34]. The optimization problem is solved given the current state of the system, one or more control inputs from the resulting optimal control trajectory are executed, and the process is repeated with the new state used as the initial condition.

In conjunction with MPC, several different relative dynamic sets have been used in the literature for formation control. In [22], an MPC scheme was designed using Gauss Variational Elements (GVE) to describe the relative motion between vehicles with inter-satellite collision avoidance ensured by defining an error box for each satellite. A related approach was taken in [24], but used the Hill–Clohessy–Wiltshire (HCW) and Lawden equations to define the relative motion, making the definition of the error box take on a form closer to a standard Cartesian frame. Similarly, Ref. [23] developed an MPC strategy using GVEs with constraints defined directly on the D'Amico Relative Orbital Elements (ROEs). However, like many optimization-based spacecraft formation techniques, e.g., [15, 19, 25, 27], Ref. [23] assumed centralized knowledge of all satellite states. In stark contrast to this, Refs. [22, 24] developed decentralized approaches that rely on a safety box constraint to allow each satellite to optimize its own trajectory. As thrusters often cannot be run continuously, an inner box is defined in [24] and the MPC controller only turned on when the satellite exited the inner box, using constraints in the optimization formulation to ensure that the trajectory never left the original error box. A linear program was developed to ensure that the optimization for each satellite was computationally feasible. Reference [24] decentralizes the problem by tying the center of the safety box to a reference orbit in a Cartesian frame.

Communication between the satellites within the cluster can often be a restricting requirement on satellites as the antenna from the satellites need to be aligned and the communication systems powered on. To reduce communication, Refs. [35–38] developed strategies for using triggering events to switch from a coasting behavior to a controlled behavior, requiring communication only when vehicles pass some relative distance threshold. To lower communication even further, this work implements a virtual structure architecture. Virtual structure techniques define the desired motion for each satellite based on the ideal motion of a virtual satellite. The virtual satellite is a computational artifact used solely to define desired relative trajectories for the constellation [39]. By agreeing a-priori on the virtual satellite orbital characteristics, all coordinated motion can be performed without communication.

The major contributions of this work are the formulation and evaluation of a virtual structure-based MPC switching strategy that can be implemented during flight of spacecraft clusters to guarantee flight safety and formation control. The formulation includes the use of a virtual leader to define the desired relative motion for each satellite using the HCW relative motion equations. Similar to [24], the MPC is used as the feedback control with a switching control strategy used to determine when the MPC control should be activated. Contributions beyond [24] include the definition of a spherical polytope as a constraint and switching definitions, a second switching surface to add hysteresis in the switching strategy (i.e., the control is not turned off using the same switching surface used to turn it on), and the formulation of different objectives for linear (L_1 and L_∞) and quadratic (L_2) programming solutions. Evaluations are performed on the required fuel usage and computation requirements of the MPC formulation. A comparison of two switching strategies is also compered, the first using a double switching surface and the second using a single switching surface.

The remainder of the paper is outlined as follows. In Sect. 2, a background of the relative dynamics used in this work is presented, as well as background on MPC, a discussion about MPC stability, and a review of switching strategies used in the literature. Section 3 presents a detailed discussion of the boundary constraints, the MPC formulation for the L_2, L_1, and L_∞ norms, as well as development of the switching strategy used for this work. Section 4 presents an overview of the simulation developed to prove out these concepts and the results and performance from the approach presented in Sect. 3. Section 5 gives a summary of the results as well as some final comments and conclusions.

2 Background

This section presents background information of the relative dynamics used with the control algorithm, preliminary information needed for the development of the MPC, as well as background information used in the switching strategy formulation.

2.1 Hill–Clohessy–Wiltshire relative dynamics

In this paper, a virtual leader is used to design a reference orbit for each satellite in the formation. There is no physical vehicle associated with the virtual leader; the virtual leader represents a fictitious satellite where motion is propagated through time according to a standard nonlinear motion model. From the virtual leader, relative orbits can be generated that define the nominal locations of each of the agents within the virtual structure. The relative orbits are propagated using the HCW equations [40–42].

The HCW dynamics are defined in the Local Vertical, Local Horizontal (LVLH) frame with the origin defined to be the location of the virtual leader. The x-axis of the LVLH frame is defined to be along the position vector of the virtual leader in the inertial frame, the z-axis is defined as the angular momentum vector of the virtual leader, and the y-axis is used to complete a right-handed coordinate system.

If the state is defined as $\mathbf{x} = \begin{bmatrix} x \; y \; z \; \dot{x} \; \dot{y} \; \dot{z} \end{bmatrix}^T$ and the control as $\mathbf{u} = \begin{bmatrix} u_x \; u_y \; u_z \end{bmatrix}^T$, where the individual control components represent the thrust accelerations of the system. The HCW equations can be written in the state space form

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \tag{1}$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3n_c^2 & 0 & 0 & 0 & 2n_c & 0 \\ 0 & 0 & 0 & -2n_c & 0 & 0 \\ 0 & 0 & -n_c^2 & 0 & 0 & 0 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{2}$$

where n_c is the mean motion of the virtual leader. An exact discrete solution can be found as

$$\mathbf{x}_{k+1} = \mathbf{A}_D \mathbf{x}_k + \mathbf{B}_D \mathbf{u}_k, \quad \mathbf{A}_D = e^{\mathbf{A}\Delta t},$$
$$\mathbf{B}_D = 0\left(\int_0^{\Delta t} e^{\mathbf{A}(\Delta t - \tau)}\, d\tau\right)\mathbf{B} \tag{3}$$

where Δt is the discretized time-step, e is the matrix exponential function, and $\mathbf{u}_k$ is the control input, assumed to be constant over the time-step. The resulting discretized HCW matrices are

$$\mathbf{A}_D = \begin{bmatrix} \Phi_{rr} & \Phi_{rv} \\ \Phi_{vr} & \Phi_{vv} \end{bmatrix} \tag{4}$$

$$\Phi_{rr} = \begin{bmatrix} 4 - 3\cos n_c \Delta t & 0 & 0 \\ 6(\sin n_c \Delta t - n_c \Delta t) & 1 & 0 \\ 0 & 0 & \cos n_c \Delta t \end{bmatrix} \tag{5}$$

$$\Phi_{rv} = \begin{bmatrix} \frac{1}{n_c} \sin n_c \Delta t & \frac{2}{n}(1 - \cos n_c \Delta t) & 0 \\ \frac{2}{n}(\cos n_c \Delta t - 1) & \frac{1}{n}(4 \sin n_c \Delta t - 3n_c \Delta t) & 0 \\ 0 & 0 & \frac{1}{n_c} \sin n_c \Delta t \end{bmatrix} \tag{6}$$

$$\Phi_{vr} = \begin{bmatrix} 3n_c \sin n_c \Delta t & 0 & 0 \\ 6n_c(\cos n_c \Delta t - 1) & 0 & 0 \\ 0 & 0 & -n_c \sin n_c \Delta t \end{bmatrix} \tag{7}$$

$$\Phi_{vv} = \begin{bmatrix} \cos n_c \Delta t & 2 \sin n_c \Delta t & 0 \\ -2 \sin n_c \Delta t & 4 \cos n_c \Delta t - 3 & 0 \\ 0 & 0 & \cos n_c \Delta t \end{bmatrix} \tag{8}$$

$$\mathbf{B}_D = \begin{bmatrix} \frac{2}{n_c^2} \sin^2 \frac{n_c \Delta t}{2} & -\frac{2}{n_c^2}(\sin n_c \Delta t - n_c \Delta t) & 0 \\ \frac{2}{n_c^2}(\sin n_c \Delta t - n_c \Delta t) & -\frac{1}{2n_c^2}(8 \cos n_c \Delta t + 3n_c^2 \Delta t^2 - 8) & 0 \\ 0 & 0 & \frac{2}{n_c^2} \sin^2 \frac{n_c \Delta t}{2} \\ \frac{1}{n_c} \sin n_c \Delta t & -\frac{1}{n_c}(2 \cos n_c \Delta t - 2) & 0 \\ -\frac{4}{n_c} \sin^2 \frac{n_c \Delta t}{2} & \frac{4}{n_c} \sin n_c \Delta t - 3\Delta t & 0 \\ 0 & 0 & \frac{1}{n_c} \sin n_c \Delta t. \end{bmatrix} \tag{9}$$

Note that $\mathbf{A}_D$ matches the discrete transition matrix in [42], while $\mathbf{B}_D$ is derived using Eqs. (2) and (3)

2.2 Model predictive control

This section presents a brief primer on MPC to present the basic idea and notation used throughout the sequel. A general overview of MPC is given, followed by a linear system representation and a discussion of the closed-loop stability. The interested reader is referred to [43] for a thorough development of MPC for linear systems.

2.2.1 General MPC overview

Optimal control is a tool often used within spaceflight design to find the most fuel-efficient strategy to perform an in-space maneuver. The basis for these dynamic optimization problems is a dynamic model that describes how the state $x(k)$ changes with time, assuming an initial condition $x(0)$ and control input $u(k)$

$$x(k+1) = g(x(k),\; u(k)), \quad x(0) = x_0, \tag{10}$$

where $g(x, u)$ generally represents a nonlinear function.

The goal of the optimal control procedure is to find the vector of inputs $U_N = \begin{bmatrix} u(0)^T \; \ldots \; u(N-1)^T \end{bmatrix}^T$, such that the objective function is optimized over a time horizon. This can be described by

$$\begin{aligned} \min_{U_N} \quad & \sum_{k=0}^{N-1} q(x_k, u_k) + p(x_N) \\ \text{s.t.} \quad & x_{k+1} = g(x(k),\; u(k)), \quad x(0) = x_0 \\ k = 1, 2, \ldots, N \quad & \\ & u_k \in U \qquad k = 0, 1, 2, \ldots, N-1 \\ & x_k \in X \qquad k = 1, 2, \ldots, N, \end{aligned} \tag{11}$$

where the terms $q(x, u)$ and $p(x)$ represent the *stage cost* and the *terminal cost*, respectively, the time horizon consists of N steps, and the state and control at each step are constrained to remain within some set of allowable values.

There are two problems that often occur when using an optimal control solution in practice. First, even with optimization algorithms taking advantage of linearities and convexities, a horizon time that is sufficiently large enough to produce desirable convergence characteristics may prove computationally prohibitive. Second, the model of the system is usually inaccurate and the system may be impacted by external disturbances that can cause it to diverge from the predicted path. For this reason, the state is measured at a future time-step, $x(k+1)$, and the optimal control problem is solved again with the current measured state, $x(k)$, set as the initial condition. This process is referred to as MPC.

While MPC generally loses any guarantee of optimality, it does provide the ability to express constraints, which is not common in many feedback control solutions. Furthermore, weights in the objective can provide intuitive "control knobs" for tuning to the desired behavior.

2.2.2 Common objective functions for linear system MPC

Common objective functions for linear systems include the L_2, L_1, and L_∞ norms as they can result in quadratic and linear programs. Given an initial state of x_0, and a time horizon of N intervals, the L_2, or quadratic, objective function can be written as

$$J(x,u) = \frac{1}{2}u_0^T R u_0 + \frac{1}{2}\sum_{k=1}^{N-1}\left[x_k^T Q x_k + u_k^T R u_k\right] \\ + \frac{1}{2}x_N^T P x_N, \tag{12}$$

where R, Q, and P are the weightings on control usage, state error, and terminal error, respectively, and are described in more detail in Appendix A.1. Similar formulations exist for the L_1 and L_∞ objective functions.

The spacecraft control literature is flooded with control strategies based on the L_2 norm [44] where it appears that there is an assumption that quadratic costs minimize propellant consumption. This paper analyzes the L_2, L_1, and L_∞ norms to evaluate if the L_2 norm is indeed the lowest fuel consuming approach and to evaluate the potential computational benefits in using the linear L_1, or L_∞ norms.

Further details on representing the L_2 norm as a quadratic programming problem, and the L_1 and L_∞ norms as linear programming problems can be found in the Appendix.

2.2.3 A note on MPC stability

Like any feedback control technique, stability of the feedback can become a concern. Given Q and R, the terminal cost of the system can be designed to ensure asymptotic stability of the MPC problem [43]. The terminal cost is chosen to represent the cost-to-go of the unconstrained infinite horizon problem for the L_2 norm. The Discrete-time Algebraic Riccati Equation (DARE) can be used to find the cost-to-go, $X^T P_\infty X$. The DARE is represented as

$$0 = A' P_\infty A - P_\infty + Q \\ - A' P_\infty B(B' P_\infty B + R)^{-1} B' P_\infty A, \tag{13}$$

where the positive definite solution for P_∞ is used.

The P_∞ found solving the DARE is then used as the matrix in the terminal cost, P, in the objective function, Eq. (12). This guarantees asymptotic stability of the system for the L_2 norm formulation [43]. Note that A and B must be completely controllable, R must be positive definite, and Q must be positive semi-definite and completely observable. For this work, the weighting matrices Q and R are chosen using *Bryson's Rule* as described in Appendix A.1.

Similar formulations to represent the cost-to-go exist for the L_1 and L_∞ cases [43].

2.3 Switching control

Switching controllers arise in many fields of applications including spacecraft formation control [45]. Previous work in the area of spacecraft formation flying switching strategies was based on maintaining a maximum and minimum relative distances [19] or six-sided error boxes [24]. References [35–38] developed an event-based system, where a corrective maneuver was triggered when the position error of a spacecraft exceeded some threshold. Reference [24] used Linear Programming (LP) to develop fuel-optimal control inputs to maintain the satellite within some tolerance of the desired state. To do this, Ref. [24] used three concentric boxes to generate the control. The *error box limit* was the largest box and represented the position volume the spacecraft was required to stay within. The error box limit was not actually used in the control design. A smaller box, the *planning error box*, was used as a conservative constraint within the LP. The smallest box was referred to as the *planning trigger box*. When the state exceeded the *planning trigger box*, a plan would be developed. The first half of the plan was then executed. If the position still exceeded the *planning error box* after executing the first half of the plan, a new plan was made, the first half executed, and the position reevaluated. This approach is similar to a general MPC formulation, except that half of the plan is executed and the system is allowed to drift if the conditions are right.

Investigating the use of a sphere polygon as the constraint in place of the six-side box was recommended in [24]. It is possible that the six-sided error box constraint could give initial conditions to the programming problem that result in higher fuel costs, such as when a satellite is near the corner of the error box with little room to maneuver. A polytope-shaped constraint alleviates this concern. The current work presents the development of two sphere-shaped polytope switching surfaces that are used to generate the switching conditions. Adding a second switching surface also has the added benefit of introducing hysteresis to the problem. The formulation of these polytope constraints is discussed in detail in Sect. 3.1. Further details on the switching strategies developed in this work are presented in Sect. 3.3.

3 Approach

This section presents a detailed discussion of the polytope boundary constraints, the objective functions, the MPC formulations for the three different norm definitions, L_1, L_2, and L_∞, as well as the control switching scheme used for this work. Change in velocity (ΔV) is used as the metric to compare the fuel savings of the different norms. These three different norms are compared in terms of requisite computation time and total ΔV. A summary of the formulation of the different norms is presented in this section with implementation details left to the Appendix. The results from the comparison of these three norms are found in Sect. 4.

3.1 Convex polytope boundary constraints

To maintain a given satellite formation, operational boundary constraints can be defined to force the state to stay within a designated volume. This allows the spacecraft to drift while also guaranteeing a given spacecraft stays in the required location within the formation. At each time-step k, the designated volume is defined by a convex polytope consisting of M faces where the position elements of the desired state $\mathbf{x}_{d,k}$ exist within the volume. The use of a convex polytope allows for a high degree of freedom in the possible constraints and for the formulation of the problem as a linear program. Each of the M planes are defined by a point $\mathbf{p}$ and a normal vector $\hat{\boldsymbol{\eta}}$ that is assumed to point toward the interior of the polytope. Let $\mathbf{r}$ be any arbitrary point. If $\mathbf{r}$ is on the boarder or interior of the polytope, then the dot product will satisfy

$$\hat{\eta} \cdot (\mathbf{r} - \mathbf{p}) \geq 0. \tag{14}$$

A matrix constraint can be used to force the state $\mathbf{x}_k$, consisting of three position and three velocity elements, to be within the polytope defined at time k. This is formulated as

$$\mathbf{A}_{poly,k}\mathbf{x}_k \leq \mathbf{b}_{poly,k}, \tag{15}$$

where

$$\mathbf{A}_{poly,k} = \begin{bmatrix} -\hat{\eta}_{1,k}^T & \mathbf{0}_{1\times 3} \\ -\hat{\eta}_{2,k}^T & \mathbf{0}_{1\times 3} \\ \vdots & \vdots \\ -\hat{\eta}_{M,k}^T & \mathbf{0}_{1\times 3} \end{bmatrix} \quad \mathbf{b}_{poly,k} = \begin{bmatrix} -\hat{\eta}_{1,k}^T \mathbf{p}_{1,k} \\ -\hat{\eta}_{2,k}^T \mathbf{p}_{2,k} \\ \vdots \\ -\hat{\eta}_{M,k}^T \mathbf{p}_{M,k} \end{bmatrix} \tag{16}$$

with $\hat{\eta}_{i,k}$ and $\mathbf{p}_{i,k}$ being the normal and point associated with the i^{th} face of the polytope at time k. Note that this implements a zero-padded version of Eq. (14) which allows the constraint to ignore the velocity.

For this work, the polytope is a close approximation of a sphere, although the six-sided error box constraint used by [24, 25] could also be accommodated. The spherical polytope approximation is done by selecting points from a spherical surface and using those for the vertices of the polytope. The convex hull formed by these points is then found and each face is used as a polytope face. At each simulation step, the polytope is formed by adding the points $\mathbf{p}_{i,k}$ to each position along the desired trajectory of the agent.

3.2 Model predictive control formulation

This work implements a model predictive controller to calculate the optimal control sequence for the case when the agent reaches the polytope boundary. Previously, Ref. [22] implemented an L_2 norm, while [24] choose to implement an L_1 norm to the spacecraft formation flying problem. This work implements L_1, L_2, and L_∞ norm objective functions with the goal of comparing the fuel savings.

If it is desired that the state, x, tracks a desired trajectory, x_d, then the error state $x_e = x - x_d$ is driven to zero, and the objective function described in Sect. 2.2 needs to be updated to account for this. Note that, for this work, the desired trajectory is the desired relative state of each agent and is initialized using ROEs and converted to the LVLH frame using a process discussed in [46]. The desired state for each agent is updated using the HCW equations.

Using the desired state, the L_2 objective function can be written as

$$\begin{aligned} J(x,u) = \frac{1}{2}u_0^T R u_0 \\ &+ \frac{1}{2}\sum_{k=1}^{N-1}\left[x_{e,k}^T Q x_{e,k} - 2x_{d,k}^T Q x_{e,k} + u_k^T R u_k\right] \\ &+ \frac{1}{2}x_{e,N}^T P x_{e,N}. \end{aligned} \tag{17}$$

The optimization problem with the polytope constraint can be written as

$$
\begin{aligned}
\min_{x,u} \quad & \frac{1}{2}u_0^T R u_0 \\
& + \frac{1}{2}\sum_{k=1}^{N-1}\left[x_k^T Q x_{e,k} - 2x_{d,k}^T Q x_{e,k} + u_k^T R u_k\right] \\
& + \frac{1}{2}\left(x_{e,N}^T P x_{e,N} - 2x_{d,N}^T P x_{e,N}\right) \\
\text{s.t.} \quad & x_{k+1} = \mathbf{A}_D \mathbf{x}_k + \mathbf{B}_D \mathbf{u}_k k = 0, 1, \ldots, N-1 \\
& \mathbf{A}_{poly,k}\mathbf{x}_k \leq \mathbf{b}_{poly,k} k = 1, 2, \ldots, N \\
& \mathbf{u}_k \leq \mathbf{1}u_{max} k = 0, 1, 2, \ldots, N-1 \\
& \mathbf{u}_k \geq -\mathbf{1}u_{max} k = 0, 1, 2, \ldots, N-1.
\end{aligned} \tag{18}
$$

The constraints ensure that the dynamics are followed, keep the spacecraft in the polytope, and limit the maximum and minimum control accelerations.

The L_1 objective function with $\mathbf{x}_e$ is

$$
J_1 = \sum_{k=1}^{N-1} |Q x_{e,k}|_1 + \sum_{k=0}^{N-1} |R u_k|_1 + |P x_{e,N}|_1. \tag{19}
$$

The full optimization formulation with the polytope constraint for the L_1 norm is then

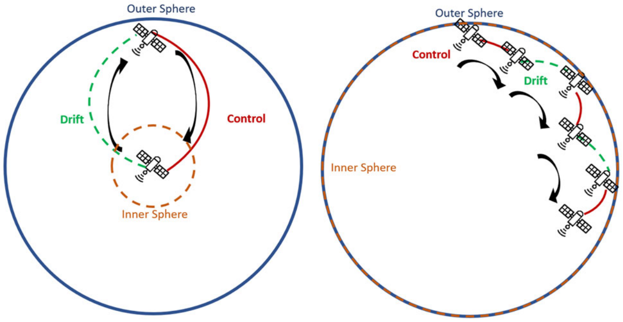
$$
\begin{aligned}
\min_{x,u} \quad & \sum_{k=1}^{N-1} |Q x_{e,k}|_1 + \sum_{k=0}^{N-1} |R u_k|_1 + |P x_{e,N}|_1 \\
\text{s.t.} \quad & x_{k+1} = \mathbf{A}_D \mathbf{x}_k + \mathbf{B}_D \mathbf{u}_k \qquad k = 0, 1, \ldots, N-1 \\
& \mathbf{A}_{poly,k}\mathbf{x}_k \leq \mathbf{b}_{poly,k} \qquad k = 1, 2, \ldots, N \\
& \mathbf{u}_k \leq \mathbf{1}u_{max} \qquad k = 0, 1, 2, \ldots, N-1 \\
& \mathbf{u}_k \geq -\mathbf{1}u_{max} \qquad k = 0, 1, 2, \ldots, N-1.
\end{aligned} \tag{20}
$$

The L_∞ norm objective function with x_e can be written as

$$
J_\infty = \sum_{k=1}^{N-1} |Q x_{e,k}|_\infty + \sum_{k=0}^{N-1} |R\mathbf{u}_k|_\infty + |P\mathbf{x}_{e,N}|_\infty. \tag{21}
$$

The full optimization formulation with the polytope constraint for the L_∞ norm is then

$$
\begin{aligned}
\min_{x,u} \quad & \sum_{k=1}^{N-1} |Q x_{e,k}|_\infty + \sum_{k=0}^{N-1} |R\mathbf{u}_k|_\infty + |P\mathbf{x}_{e,N}|_\infty \\
\text{s.t.} \quad & x_{k+1} = \mathbf{A}_D \mathbf{x}_k + \mathbf{B}_D \mathbf{u}_k \qquad k = 0, 1, \ldots, N-1 \\
& \mathbf{A}_{poly,k}\mathbf{x}_k \leq \mathbf{b}_{poly,k} \qquad k = 1, 2, \ldots, N \\
& \mathbf{u}_k \leq \mathbf{1}u_{max} \qquad k = 0, 1, 2, \ldots, N-1 \\
& \mathbf{u}_k \geq -\mathbf{1}u_{max} \qquad k = 0, 1, 2, \ldots, N-1.
\end{aligned} \tag{22}
$$



(a) Strategy 1 (b) Strategy 2

Fig. 1 **a** Strategy 1: Using fewer large maneuvers to control a spacecraft to the center of a keep-in-volume; **b** Strategy 2: Using a large number of smaller maneuvers to stay just inside the boundary of the keep-in-volume

3.2.1 A comment on stability

As discussed in Sect. 2.2.3, the terminal cost is chosen to represent the cost-to-go of the unconstrained infinite horizon problem. This guarantees asymptotic stability of the system [43]. For this work, the trajectory input is zero, the error state is substituted for the normal state, and the error is driven to zero. Thus, the same arguments for stability discussed in Sect. 2.2.3 can be employed. The P_∞ found from Eq. (13) was used for each norm, and through simulation, the authors found that it worked well for all norm cases.

3.3 Switching strategy

This paper creates and compares two formation flying operational control techniques with the goal of minimizing fuel usage. The first strategy, *Strategy 1*, is designed to periodically uses large maneuvers to drive the position and velocity error to near zero. The second strategy, *Strategy 2*, allows the spacecraft to perform many small maneuvers to stay just inside the boundary of the keep-in-volume. A depiction of these two strategies is presented in Fig. 1.

A switching strategy is developed to determine when to employ the MPC controller and when to allow the satellite to coast. Fundamental to the switching conditions are two spherical polytopes. A larger polytope, referred to as the *outer polytope*, is used as the agent's "keep-in-volume". If it is predicted that the spacecraft's trajectory will leave the *outer polytope*, then the control is turned on. A smaller polytope, referred to as the *inner polytope*, is used as part of the trigger that turns off the control when the spacecraft's predicted trajectory lies within it. For Strategy 2, the *inner polytope* is set to have the same radius as the *outer polytope*. This causes the control to only be on long enough to keep the agent just inside the *outer polytope*. The *outer* and *inner* polytope are depicted in Fig. 1. The inner polytope is also used in the design of the weighting matrix Q using the radius for the first three elements in *Bryson's Method*, as described in Appendix A.1.

There are two advantages to using Bryson's Method to determine the Q and R weighting matrices. First, it provides a balance, such that neither the state error nor control usage is overly focused on to the detriment of the other. Second, this method naturally guides the system to remain within the state and control constraints. While explicit constraints are still needed to guarantee that these bounds are not exceeded, this does provide a way to link the dynamics and boundaries to obtain more reasonable trajectories.

To determine if the spacecraft will leave a polytope, a drift horizon N_{drift} is used. The spacecraft state is propagated using the discrete form of the uncontrolled HCW equations as

$$\hat{\mathbf{x}}_{k+1} = A_D \hat{\mathbf{x}}_k \quad k = 0, \ldots, N_{drift} - 1, \tag{23}$$

where $\hat{\mathbf{x}}_k$ are the projected drift states of the spacecraft and $\hat{\mathbf{x}}_0$ is the current estimate for the state of the spacecraft. Since this propagation is used to determine the behavior of the spacecraft once the controls are turned off, no control inputs are used.

In each strategy, the information in Fig. 2 is used to determine whether to turn the MPC controller on or off. In addition to the switching surfaces, a velocity check is used to compare the velocity of the spacecraft at each point along the drift trajectory with the desired trajectory. The velocity cutoff condition is defined as

$$|\mathbf{v} - \mathbf{v}_d|_2 \leq \delta v, \tag{24}$$

where $\mathbf{v}$ is the current relative velocity, $\mathbf{v}_d$ is the desired relative velocity, and δv is the allowable relative velocity error. This velocity check causes the MPC to continue to operate after the desired positional constraint is met to provide better convergence to the desired trajectory. If a velocity check is not desired, then δv can be set to ∞ and is effectively ignored. This is the case for Strategy 2 as the vehicle is far from the desired state. When drifting, if any drift state is found to be outside the outer polytope, the controller turns on. The control remains on until all the drift states are found to be within the inner polytope and the velocity check is met. Note that in Fig. 2, the variables with primes indicate the inner polytope.

This work presents two sphere-shaped convex polytope switching surfaces that are used to develop switching conditions. The use of a sphere-shaped switching surface avoids the case where the spacecraft is near the corner of the error box. Also by adding a second surface to the switching strategy, hysteresis is added to the problem, adding robustness to the system [47].

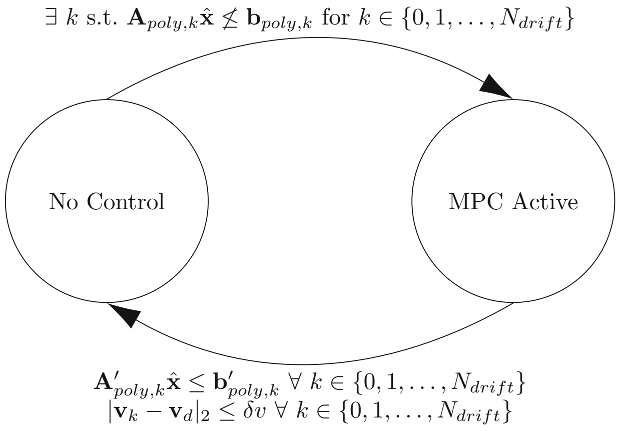

$\exists\ k$ s.t. $\mathbf{A}_{poly,k}\hat{\mathbf{x}} \not\leq \mathbf{b}_{poly,k}$ for $k \in \{0, 1, \ldots, N_{drift}\}$
No Control
MPC Active
$\mathbf{A}'_{poly,k}\hat{\mathbf{x}} \leq \mathbf{b}'_{poly,k}\ \forall\ k \in \{0, 1, \ldots, N_{drift}\}$
$|\mathbf{v}_k - \mathbf{v}_d|_2 \leq \delta v\ \forall\ k \in \{0, 1, \ldots, N_{drift}\}$

Fig. 2 Transitions between MPC control states. Primes indicate the inner polytope variables

4 Numerical simulation

This section discusses the implementation of the MPC and switching surfaces discussed in the previous sections. A discussion of the simulation is discussed followed by an evaluation of the control trajectories, comparison of computation time of the different norms, discussion of the control usage, and a comparison of the ΔV usages.

4.1 Simulation overview

For each formation, the virtual leader's initial state is defined using Keplerian orbital elements and then converted to a Cartesian Earth Centered Inertial (ECI) state. The actual and desired location of the agent in the formation are initialized using D'Amico's ROEs [48]. The ROEs are first converted to inertial orbital elements using the leader as a reference, and then to ECI coordinates [46]. Those ECI coordinates are then converted to LVLH coordinates, again using the leader as the reference. The advantage to using ROEs to define the initial states of the agent is that they offer a more intuitive understanding of the relative motion when compared to Cartesian coordinates.

The true states for the leader and agents are propagated using the nonlinear dynamics

$$F_d = -\frac{1}{2}\rho C_d A |v_{rel}| v_{rel} \frac{1}{m} \tag{25}$$

$$F_{J2} = \frac{3}{2}\frac{J_2 \mu R^2}{r^4}\left[\frac{x}{r}\left(5\frac{z^2}{r^2} - 1\right)\hat{i} + \frac{y}{r}\left(5\frac{z^2}{r^2} - 1\right)\hat{j} + \frac{z}{r}\left(5\frac{z^2}{r^2} - 3\right)\hat{k}\right] \tag{26}$$

$$\dot{r} = v \tag{27}$$

$$\dot{v} = -\mu\frac{r}{|r|_2^3} + F_{J2} + F_d + u, \tag{28}$$

where F_d is the force due to drag, F_{J2} is the force due to the J_2 perturbation [42], and u is the calculated control acceleration, applied as a zero-order hold.

The drag force is a simplified model where the air density, ρ, is determined by interpolating the altitude against selected values in the US Standard Atmosphere 1976 model [49]. The relative velocity of the spacecraft to the Earth's atmosphere, v_{rel} is calculated as [42]

$$v_{rel} = v - \omega \times r, \tag{29}$$

where the Earth's rotation, ω, is assumed to be entirely in the $\hat{k}$ direction. The coefficient of drag, C_d, and spacecraft area, A, are parameters for the spacecraft and held constant during the simulation. The virtual leader is assumed to only be affected by the J_2 perturbations and not by drag, while the agent spacecraft are all affected by both J_2 and drag. Note that while the example given is for a circular orbit including J_2, this method can also be applied to elliptical orbits and orbits with higher order perturbation terms.

The desired and drift trajectories are propagated using the HCW equations. Since the desired trajectory is compared against the drift trajectory and used in the MPC calculations, it is projected forward a number of steps equal to $\max(N_{drift}, N_{MPC})$. The drift trajectory is projected forward N_{drift} steps using the HCW equations and each agent spacecraft's current relative LVLH state. At each time-step of the desired trajectory, the inner and outer polytopes are applied. The desired trajectory is centered on the desired position at the time-step in question, and compared to the corresponding time-step on the drift trajectory. The control state is then changed as described in Sect. 3.3.

The control is calculated in the LVLH frame as an acceleration using one of the formulations described in Sect. 3.2 and then converted to the ECI frame where it is used to propagate the state forward. This acceleration is converted to a force using the current mass of the spacecraft to ensure that it is within the limits of the available thrust and then applied to the dynamics as indicated above. From the acceleration, the mass flowrate is calculated as

$$\dot{m} = \frac{||u||_1 m}{g_0 I_{sp}}, \tag{30}$$

where $u \in \mathbf{R}^3$ is the control acceleration, m is the current mass, g_0 is standard gravity, and I_{sp} is the specific impulse. The simulation then repeats, as seen in Algorithm 1, until a predesignated cut-off time is reached.

An example of the desired and actual trajectories of the agent spacecraft is shown in Fig. 3. Although the agent is always near the desired trajectory, the actual trajectory shows noticeable deviations as it drifts due to perturbations and course corrects. A single moment from the simulation can be seen in Fig. 4. Figure 4 shows the current inner and outer polytopes as well as the current desired and actual positions, shown as red and blue asterisks, respectively.

Algorithm 1 Simulation structure

```
Initialize spacecraft states
Compute relative LVLH states
$control\_on \leftarrow false$
for Each Simulation Step do
    Propagate drift trajectory, $\hat{\mathbf{x}}_k$,
    Propagate polytopes
    if $\exists\ \hat{\mathbf{x}}_k$ is outside the Outer Polytope then
        $control\_on \leftarrow true$
    if $\forall\ \hat{\mathbf{x}}_k$ inside Inner Polytope and Velocity constraint met then
        $control\_on \leftarrow false$
    if $control\_on$ then
        Calculate $u$ with MPC
    Propagate true states
```

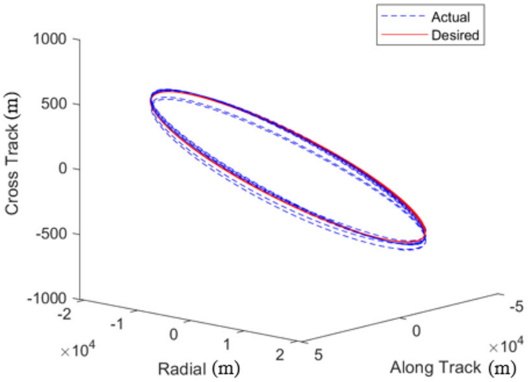


Fig. 3 Desired and actual spacecraft trajectories shown in the LVLH frame

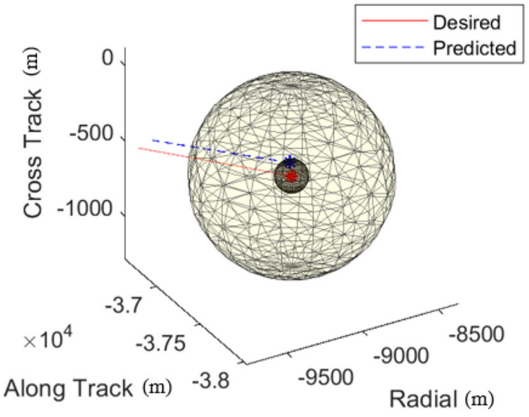


Fig. 4 The current inner and outer polytope volumes are shown with the desired and predicted future trajectories in red and blue, respectively

4.1.1 Simulation parameters

A scenario based on a standard 12U CubeSat design built by the Space Dynamics Laboratory for formation flying mis-

Table 1 Parameters for simulation

Parameter	Value
Spacecraft mass	24 kg
Spacecraft C_D	2.2
Spacecraft area to mass	0.001 m^2/kg
Spacecraft thrust	0.5 N
Simulation time-step	10 s
Simulation time-steps	8640 (86,400 s)
MPC horizon	60 steps (600 s)
Drift horizon	15 steps (150 s)

Table 2 Reference orbit of the virtual leader

Keplerian orbital element	Value
Semi-major axis	6878 km
Eccentricity	1×10^{-4}
Inclination	25°
RAAN	45°
Argument of perigee	0°
Initial true anomaly	100°

Table 3 Relative orbital elements for the desired agent trajectory

ROE element	Spacecraft 1	Spacecraft 2	Spacecraft 3
δa	0	0	0
$\delta\lambda$	0	3×10^{-3}	-3×10^{-3}
δe_x	3×10^{-3}	3×10^{-3}	3×10^{-3}
δe_y	3×10^{-4}	3×10^{-4}	3×10^{-4}
δi_x	1×10^{-4}	1×10^{-4}	1×10^{-4}
δi_y	0	0	0

sions was used to evaluate the MPC developed in Sect. 3. u_{max} is set as the maximum allowed acceleration based on the values in Table 1. z_{max} is set as the radius of the inner polytope for position terms and set as zero for velocity terms. Each strategy was tested over multiple outer polytope sizes which ranged in radius from 200 to 1000 m in increments of 50 m. An inner polytope radius of 100 m and a velocity cutoff of 0.06 m/s was used for Strategy 1.

The virtual leader reference orbit is a near circular Low Earth Orbit (LEO) orbit with the parameters shown in Table 2. The physical characteristics and the simulation parameters used in this work are shown in Table 1. The propulsion system modeled in this work is based on the hybrid system described in [50]. It is assumed that the spacecraft propulsion system is able to provide a thrust of up to 0.5 N along any three orthogonal axes and that the thrust is throttleable up to the maximum thrust of 0.5 N.

As shown in Table 1, the simulation uses a fixed time step of 10 s. The size of the time-step constrains the time the control must be on. And although, in many situations, a pure impulsive maneuver is the optimal solution, in practice, any maneuver that is $\leq$ 12% of the orbit period is near identical in performance to an impulsive maneuver. In addition, the duration of this time-step can be adjusted to offer the spacecraft operator a more refined maneuver if desired.

4.2 Computation time

The computation time for each of the three norms was evaluated as the time horizon of the MPC increases. For each norm definition, a series of trials with the time horizon ranging from 1 to 150 steps was performed. In each trial, the spacecraft was initialized according to the parameters in Tables 1 and 3 with the exception that the simulations were only run for 120 steps, or 1200 s, so that a broader range of horizons could be tested in a reasonable time. Additionally, only an outer polytope radius of 800 m was employed. Data were only collected when the MPC was active and so does not include any computation times of 0 s when the controller was not used.

In the results for both Strategy 1, Fig. 5, and Strategy 2, Fig. 6, the L$_2$ computation time is much lower than either the L$_1$ or L$_\infty$ norm. This is, however, dependent on the approach taken with implementing the optimization. Initially, the L$_2$ time was higher, but scaling the optimization problem showed a drastic decrease in computation time. As would be expected, there is an increase in computation time for all norm definitions as the length of the MPC horizon increases. The minimum and maximum computation times for each of the norms are shown in Table 4.

One thing to notice is that for L_2, the Strategy 2 computation times are generally higher than Strategy 1 for a given MPC horizon. This is due, in part, to the fact that Strategy 2 cycles between the controller on and off states more frequently than Strategy 1. For both strategies, when the controller activated it does not have a meaningful initial solution. However, for subsequent iterations, the controller can use the previous solution to warm start the solver and reduce the computation time. Since Strategy 2 cycles the controller more often, it is frequently re-initializing the solver, while the controller for Strategy 1 is generally active for several consecutive simulation steps.

4.3 Strategy trajectory evaluation

We now show that the switching strategies achieve the desired behaviors depicted in Fig. 1. Figures 7 and 8 show example trajectories produced using the switching strategies with an

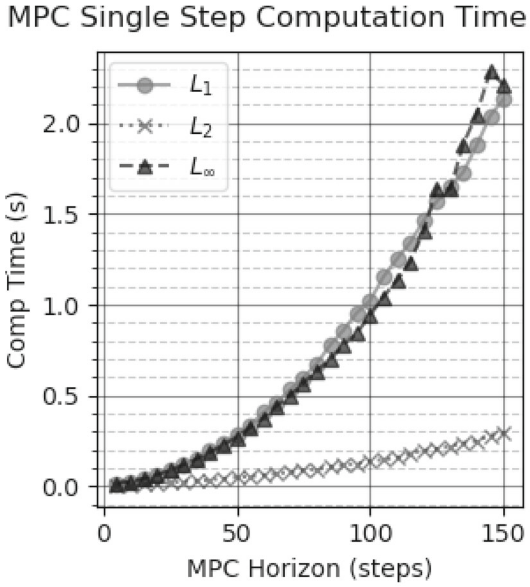


Fig. 5 Computation time required for planning a single simulation step for the various norm definitions when using Strategy 1

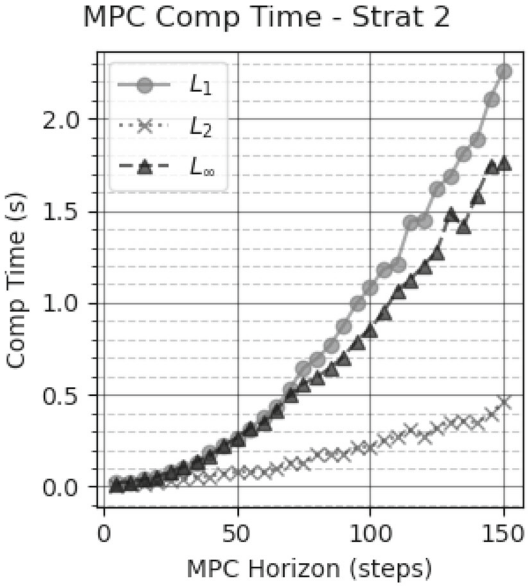


Fig. 6 Computation time required for planning a single simulation step for the various norm definitions when using Strategy 2

outer radius of 800 m using the L_1 norm. The L_2 and L_∞ norms gave similar results but are not presented.

The figures are shown in an LVLH frame centered on the desired trajectory, so the displayed line indicates the perturbed trajectory of the spacecraft, relative to the desired. Each simulation is initialized with a random initial state, indicated by the blue dot with the outer polytope shown in blue and the inner polytope shown in light green. For Strategy 2, the inner and outer polytopes are identical, so only the inner polytope is seen.

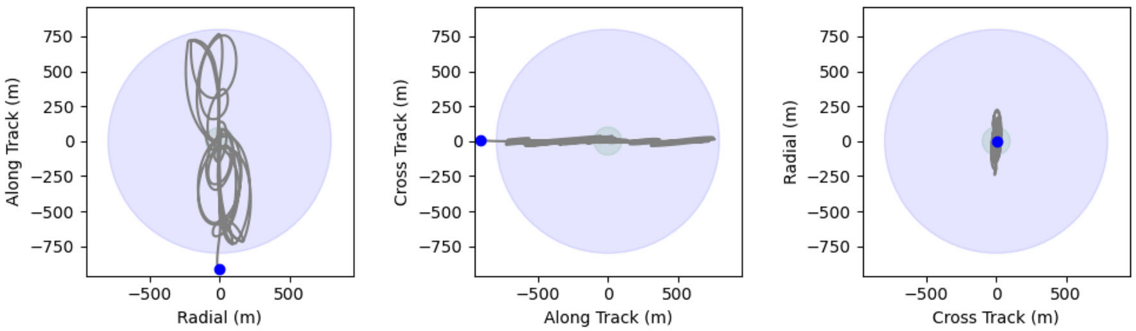
In Fig. 7, the spacecraft trajectory is shown using Strategy 1. It is initialized just outside the outer polytope, but, once it guides itself within the polytope, it is able to remain inside. It executes a number of loops where it returns to the center of the polytope once it reaches the boundary and then turns off the control until it nears the outer polytope as its drift becomes too severe. As seen in the cross track, along track, and radial-cross track plots (Fig. 7b, c, respectively), the motion exists almost exclusively in the along track axis with only minor deviations in the radial direction.

Contrastingly, with Strategy 2, the spacecraft initially moves in a large loop in the along track-radial plane (Fig. 8a) but eventually ends up maintaining a position just along the edge of the outer polytope. As seen in Fig. 8b, c, the motion, like with Strategy 1, is almost entirely in the radial and along track dimensions.

The magnitude of the spacecraft position error is shown in Fig. 9a for Strategy 1 and Fig. 9b for Strategy 2. In Strategy 1, there is a dramatic drop in the error as the spacecraft maneuvers back to the center of the inner polytope and the controller turns off. The spacecraft continues to undergo a cycle of drifting and correcting, although the period of the cycles is not entirely consistent. The smaller oscillations are a result of the dynamics slowing moving the spacecraft away from the desired trajectory, while the sharp drops in error are the result of the controls. Some of the steep drops in error, due to the control, are followed by nearly equally steep rises in error. This is a result of the spacecraft not matching the desired trajectory and quickly drifting away from it, since the closer the spacecraft is to the desired trajectory, the smaller the perturbations are between the two.

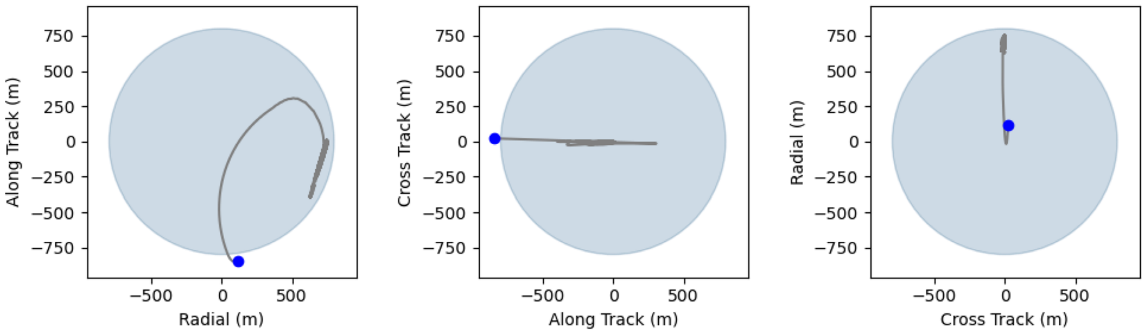
Table 4 Computation time comparisons

Norm	Strategy	Min computation time (s)	Max computation time (s)
L_1	1	0.016 ± 0.003	2.128 ± 1.029
	2	0.019 ± 0.005	2.258 ± 0.916
L_2	1	0.004 ± 0.001	0.293 ± 0.153
	2	0.005 ± 0.002	0.466 ± 0.359
L_∞	1	0.016 ± 0.003	2.284 ± 5.979
	2	0.018 ± 0.004	1.759 ± 0.573



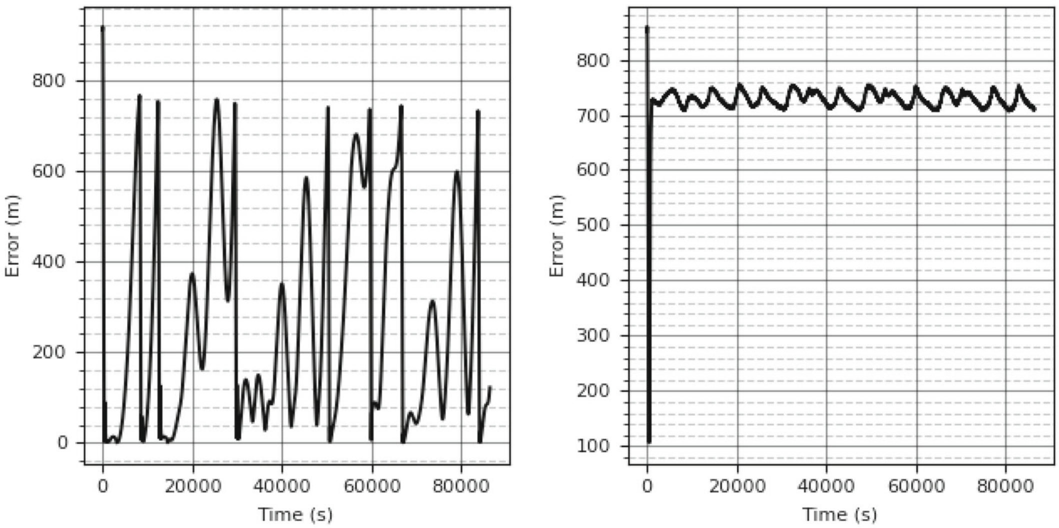
(a) (b) (c)

Fig. 7 Strategy 1 trajectory for the L_1 norm



(a) (b) (c)

Fig. 8 Strategy 2 trajectory for the L_1 norm



(a) (b)

Fig. 9 Position error for Strategy 1 (**a**) and 2 (**b**) using the L_1 norm

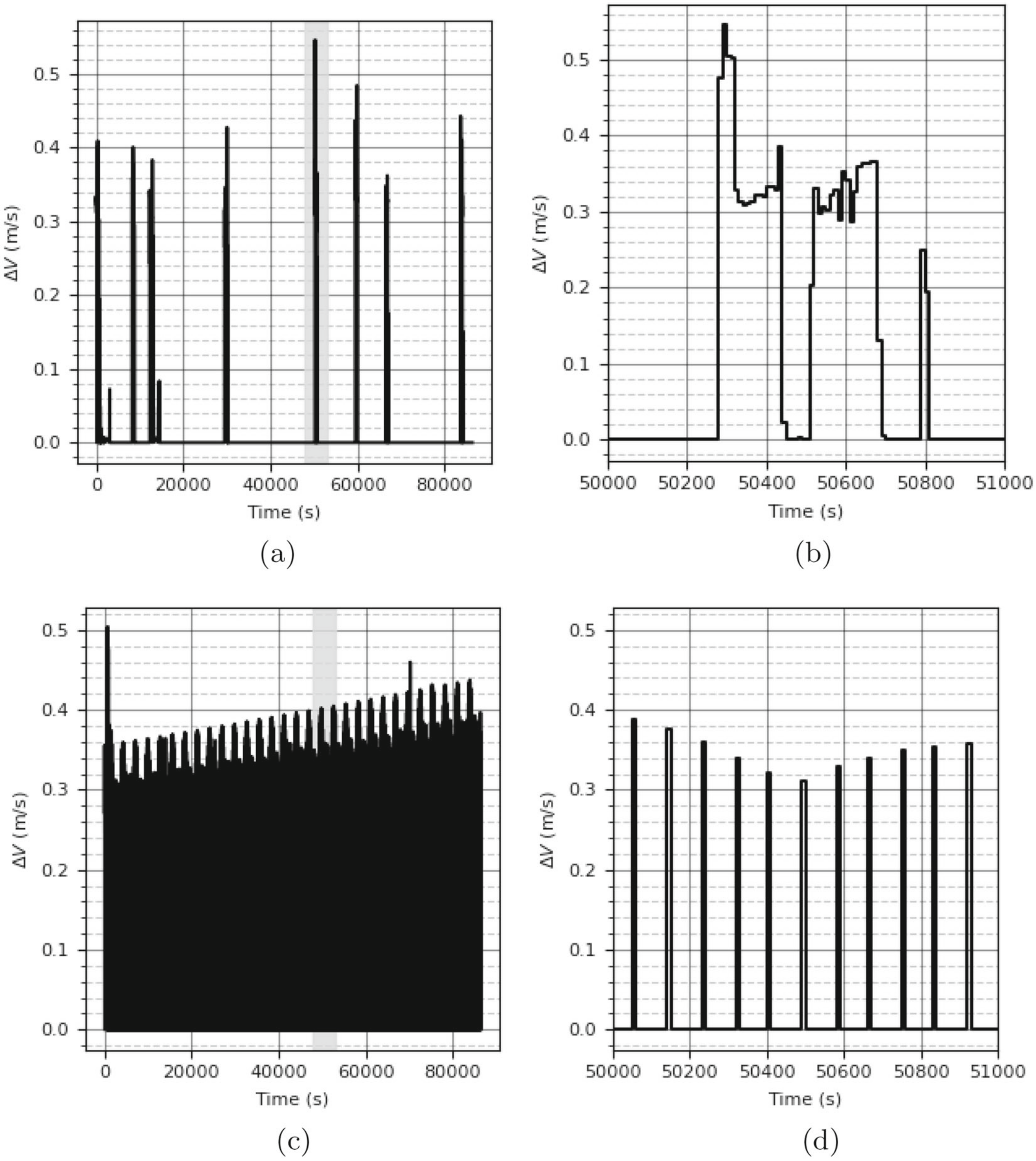
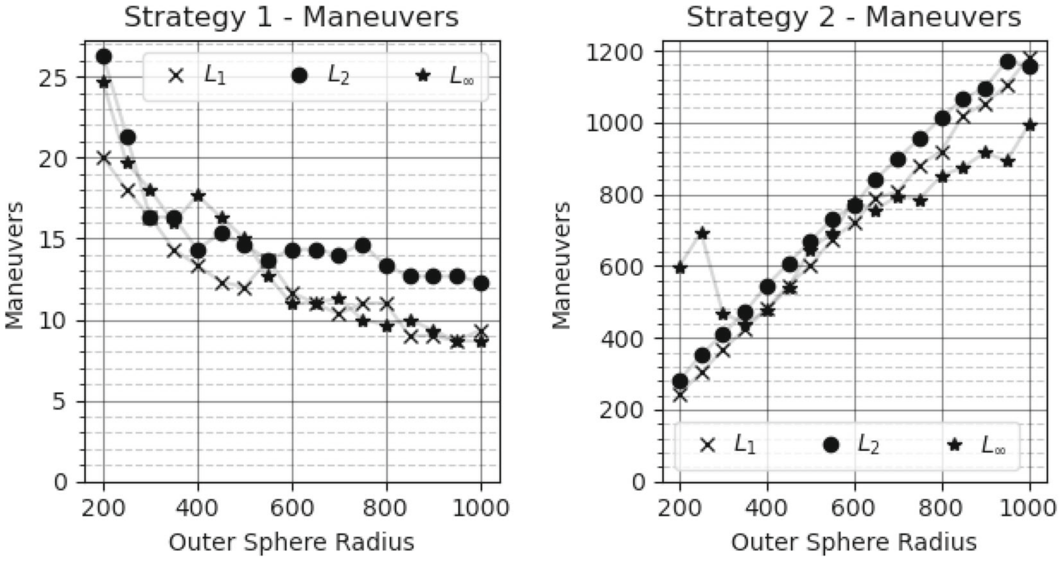


Fig. 10 ΔV usage for Strategy 1 (**a**) and 2 (**c**) using the L_1 norm with highlighted sections expanded in (**b**) and (**d**), respectively

With Strategy 2 (Fig. 9b), there is also a significant initial drop in the position error where the control places the spacecraft on an unpowered course that passes near the desired trajectory. However, Strategy 2 allows the spacecraft to continue to drift until it again reaches the boundary of the outer polytope. However, after this, the error is roughly constant for the remainder of the simulation, as the spacecraft stays near the boundary. There continues to be variations in the error but with much smaller magnitude and more regular periods than seen in Strategy 1.

4.4 Control usage

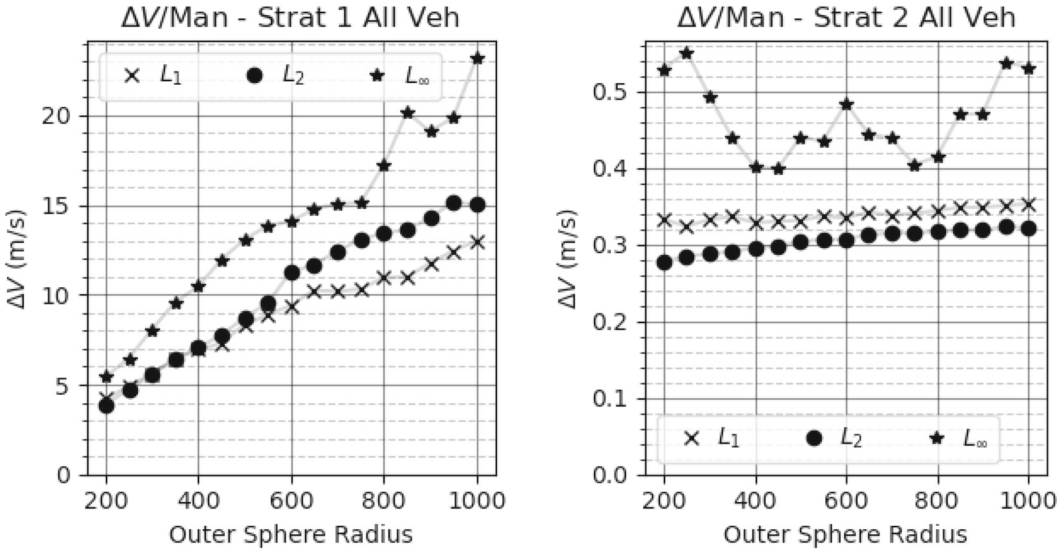
An example of the control usage for both strategies is shown in Fig. 10 where the ΔV usage over time is plotted for the example trajectories shown in Figs. 7 and 8. The full time history is shown in Fig. 10a, c for Strategy 1 and 2, respectively, while Fig. 10b, d expands the indicated time segments for a more detailed examination of the control usage. With Strategy 1 (Fig. 10a, b), the periods where the control is active are relatively infrequent but can span several minutes. If these plots are compared to the error plot (Fig. 9a), the correlation between the ΔV usage and the drop in error can be seen. While the small oscillations in error are due to the natural dynamics of the spacecraft, the sharp drops that occur after



(a)

(b)

Fig. 11 Number of maneuvers used in Strategy 1 (**a**) and 2 (**b**)



(a)

(b)

Fig. 12 Average ΔV per maneuver in Strategy 1 (**a**) and 2 (**b**)

the error exceeds roughly 750 m are the direct result of the control usage.

The ΔV usage for Strategy 2 (Fig. 10c, d) seems, at first glance, to be continuous. However, when examining the detailed section (Fig. 10d), it is clear that while the control usage is quite regular, it is not constant. The instances of control usage are approximately 1 simulation step in duration and separated by about 7–10 simulation steps where no control is applied. As seen in the error plot (Fig. 9b), apart from a sudden drop in error at the start of the simulation, the control used is just enough to keep the error in the 700–800 m range.

4.5 Comparison analysis of varying polytope size

To examine the impact of the polytope sizes, simulations were run with outer polytopes ranging from 200 to 1000 m in increments of 50 m. Control usage was tracked across maneuvers where each maneuver is defined as a period beginning when the control turns on and ending when it turns off, or

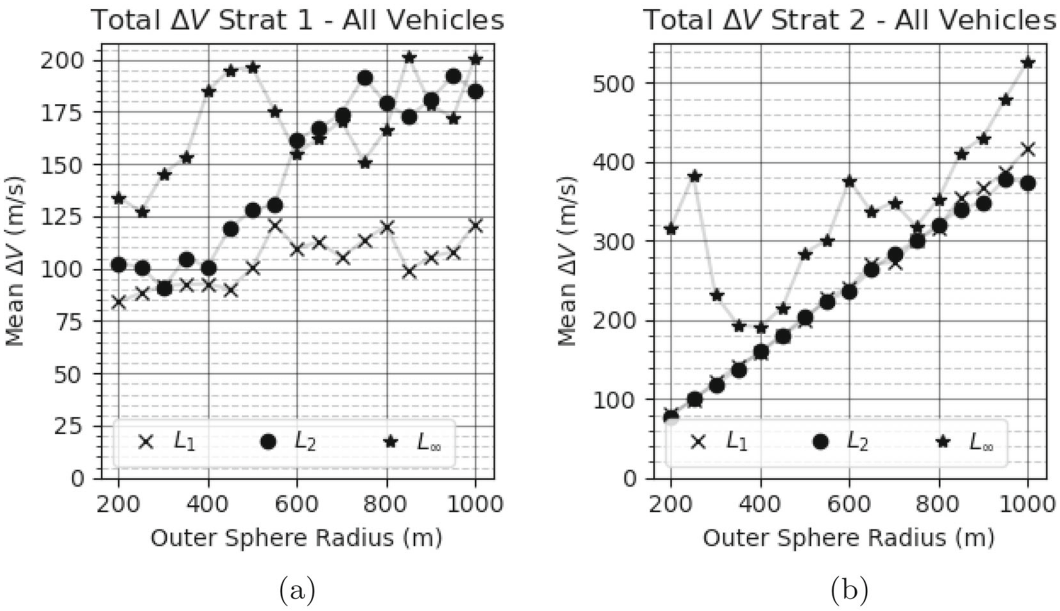


Fig. 13 Total ΔV used in Strategy 1 (**a**) and 2 (**b**)

one loop through the state diagram in Fig. 2. Additionally, the number of maneuvers for each simulation is tracked, as well as the total ΔV used over the full simulation. Each simulation was run for a duration of 1 day with the parameters indicated previously (Tables 1, 2, and 3). Each simulation run has three spacecraft in formation, all with identical parameters other than the desired orbit. The results present the average of the three spacecraft in each formation for each outer sphere size.

In Fig. 11a, b, the number of maneuvers for each simulation is shown. In each case, simulations using Strategy 2 used significantly more maneuvers than those using Strategy 1, for an identical outer polytope size. For Strategy 1, the decrease in number of maneuvers with polytope size is due to the fact that it takes longer for the spacecraft to drift back out to the polytope. The number of maneuvers needed for Strategy 2 generally increases as the polytope size increases for the L_1 or L_2 norms which is to be expected, since hovering at the boundary requires a large number of small maneuvers.

The increasing number of maneuvers needed can be explained by considering the relative dynamics of the agent and virtual leader as a function of the distance between the two. When the agent and leader are close, their dynamics are very similar and the relative drift between them is small. However, as they separate, the dynamics diverge and the drift effects are magnified. Thus, when hovering at the boundary of a larger polytope, the agent drifts out of bounds faster and needs more frequent control usage. Further investigation is required to understand why a monotonically increasing value is not seen in the L_∞ norm case.

Figure 12a, b presents the average ΔV per maneuver as a function of increasing polytope size. Strategy 1's control usage increases as the polytope sizes increase primarily because the larger keep-in volume requires a longer burn duration to return to the center. Strategy 2 uses significantly less control per maneuver, and while there is a slight increase in control usage as the polytope size gets larger, it is significantly less pronounced than what is seen in Strategy 1. While the increase in Strategy 1 is primarily due to the lengthened burn duration, the increase in Strategy 2's ΔV usage is due to the fact that as the spacecraft moves further from the desired trajectory the magnitude of the relative perturbing forces increases and requires an equivalent increase in the control. For both strategies, the L_1 and L_2 norms show roughly equivalent performance, while the L_∞ norm is notably higher.

The average total ΔV used over the course of the simulation can be seen in Fig. 13a, b. This metric shows the combined effect of the number of maneuvers and the ΔV used on each maneuver. For Strategy 1, there is a small rise in the total ΔV as the polytope size increase. However, there is also a large degree of variability which makes understanding individual trends difficult. Strategy 2 has a more defined trend for the L_1 and L_2 norms where the total ΔV increases with the polytope size. For the L_∞ norm, there is not a clearly defined trend and the control usage appears less predictable. The results show that using the L_1 or L_2 norms result in lower control usage when staying close to the desired trajectory and changes consistently when increasing the allowable distance from that trajectory.

When comparing the strategies, Strategy 1 shows notable improvement over Strategy 2 in terms of overall ΔV usage, especially as the outer polytope size increases. It should be noted that many operational constraints seen in practice, but ignored for this analysis, would also favor using Strategy 1. For example, realistic engine throttling, required engine cool-down periods, and mission requirements would limit the applicability of Strategy 2 since continual minor thrust impulses may not be practical.

Comparing the different norm definitions, the L_1 norm seems to provide the best performance when implementing either of these strategies since it's overall ΔV usage is lower, and more consistent, than the other norms.

5 Conclusion

This paper presents an MPC-based control scheme that is used in conjunction with two polytope switching surfaces, and the HCW equations, to maintain a desired trajectory while also allowing freedom to maneuver within the allowable bounds. The operational polytope switching surfaces enable the predictive framework of the MPC to be used to maintain flight safety and achieve mission formation. The surfaces are designed, such that multiple agents can be flown in a single formation with no overlapping regions. This allows the vehicles to execute avoidance strategies without continually maintaining the trajectories of other agents. The MPC framework combined with the polytope switching surfaces enables a scalable method for clusters of satellites to coordinate and safely achieve mission objectives, while minimizing fuel usage.

Numerical simulations show that the control formulation is able to maintain its position within the polytope volume. This work shows a comparison of two operational control strategies that can be implemented, without the sharing of state information between spacecraft. A comparison of the number of maneuvers required for the two control strategies and the total ΔV usage is presented. This paper shows that, in general, the lower total ΔV option is a strategy where the controller commands the spacecraft to the center of the keep-in-volume, *Strategy 1*. This paper also shows that if a lower amount of fuel per maneuver is required, then a strategy where the controller keeps the spacecraft just inside the boundary, *Strategy 2*, is preferable. This paper also presents and compares the formulation of using L_1, L_2, and L_∞ norm objective functions for the MPC where, in general, the L_1 and L_2 norms require the least amount of fuel usage.

Acknowledgements This work was funded by Space Dynamics Laboratory.

Funding This work was funded by the Space Dynamics laboratory's Internal Research and Development program.

Declarations

Conflict of interest The authors have not disclosed any competing interests.

Consent for publication Granted.

A Appendix

A.1 Bryson's method

Bryson's method is a simple method for getting a reasonable choice of the **Q** and **R** matrices [51], where the diagonal elements are selected as

$$Q_{ij} = \begin{cases} \frac{1}{z_{i,\max}^2} & i = j \\ 0 & i \neq j \end{cases} \quad i, j \in 1, \ldots, \ell \tag{31}$$

$$R_{ij} = \begin{cases} \frac{1}{u_{i,\max}^2} & i = j \\ 0 & i \neq j \end{cases} \quad i, j \in 1, \ldots, k, \tag{32}$$

where $z_{i,\max}$ is the maximum desired deviation for the i^{th} state element and $u_{i,\max}$ is the maximum desired control for the ith control element. This effectively normalizes or nondimensionalizes the optimization problem, so that the states and controls are equally balanced. The choice of **Q** and **R** guide, but do not constrain, the evolution of the state error and control usage, and without other constraints, there is no guarantee that the $z_{i,max}$ and $u_{i,max}$ values will not be exceeded.

For the L_2 norm, the cost function is related to the square of the control and error states, as shown in Eqs. (31) and (32). For the L_1 and L_∞ norms, the cost functions are related to the absolute value and so Bryson's method as described above will need to be modified. To normalize the cost function to roughly match the L_2 objective, the **Q** and **R** weights are set as

$$\mathbf{Q}_{\infty,ij} = \begin{cases} \frac{1}{|z_{i,\max}|} & i = j \\ 0 & i \neq j \end{cases} \quad i, j \in 1, \ldots, \ell \tag{33}$$

$$\mathbf{R}_{\infty,ij} = \begin{cases} \frac{1}{|u_{i,\max}|} & i = j \\ 0 & i \neq j \end{cases} \quad i, j \in 1, \ldots, \ell. \tag{34}$$

For this paper, $\mathbf{Q}_{\infty,ij}$ and $\mathbf{R}_{\infty,ij}$ are used to represent the state error and control usage weights for the L_∞ and L_1 formulations.

A.2 L_1 formulation

The L_1 norm is based on work from Borrelli, Bemporad, and Morari [43]. By introducing the variables ε_k^x and ε_k^u, such that

$$\boldsymbol{\varepsilon}_k^x = \left[\varepsilon_{k,1}^x \; \varepsilon_{k,2}^x \; \cdots \; \varepsilon_{k,n}^x\right]^T \tag{35}$$

$$\boldsymbol{\varepsilon}_k^u = \left[\varepsilon_{k,1}^u \; \varepsilon_{k,2}^u \; \cdots \; \varepsilon_{k,m}^u\right]^T \tag{36}$$

into the the objective function for the L$_1$, the L$_1$ objective function can be written as

$$J_1 = \sum_{k=1}^{N} \sum_{j=1}^{n} \boldsymbol{\varepsilon}^x_{k,j} + \sum_{k=0}^{N-1} \sum_{j=1}^{m} \boldsymbol{\varepsilon}^u_{k,j} \tag{37}$$

with constraints formulated as

$$\begin{aligned}
-\boldsymbol{\varepsilon}^x_k &\leq Q\mathbf{x}_{e,k},\ k = 1, 2, \ldots, N-1\\
-\boldsymbol{\varepsilon}^x_k &\leq -Q\mathbf{x}_{e,k},\ k = 1, 2, \ldots, N-1\\
-\boldsymbol{\varepsilon}^x_N &\leq P\mathbf{x}_{e,N}\\
-\boldsymbol{\varepsilon}^x_N &\leq -P\mathbf{x}_{e,N}\\
-\boldsymbol{\varepsilon}^u_k &\leq R\mathbf{u}_k\ k = 0, 1, \ldots, N-1\\
-\boldsymbol{\varepsilon}^u_k &\leq -R\mathbf{u}_k\ k = 0, 1, \ldots, N-1,
\end{aligned} \tag{38}$$

where each inequality is evaluated element-wise. The constraints allow each element of the ε vectors to be individually adjusted to find the minimum possible value while not decreasing any further than the weighted errors or controls allow. The use of two constraints with each ε, one positive and one negative, forces each ε value to be positive, consistent with the L$_1$ norm definition.

The full optimization formulation with the polytope constraint for the L$_1$ norm is then

$$\begin{array}{lll}
\min\limits_{\varepsilon} & \displaystyle\sum_{k=1}^{N} \sum_{j=1}^{n} \boldsymbol{\varepsilon}^x_{k,j} + \sum_{k=0}^{N-1} \sum_{j=1}^{m} \boldsymbol{\varepsilon}^u_{k,j} & \\
\text{s.t.} & x_{k+1} = \mathbf{A}_D\mathbf{x}_k + \mathbf{B}_D\mathbf{u}_k & k = 1, 2, \ldots, N\\
 & \mathbf{A}_{poly,k}\mathbf{x}_k \leq \mathbf{b}_{poly,k} & k = 1, 2, \ldots, N\\
 & \boldsymbol{\varepsilon}^x_k + Q\mathbf{x}_k \geq Q\mathbf{x}_{d,k} & k = 1, 2, \ldots, N-1\\
 & \boldsymbol{\varepsilon}^x_k - Q\mathbf{x}_k \geq -Q\mathbf{x}_{d,k} & k = 1, 2, \ldots, N-1\\
 & \boldsymbol{\varepsilon}^x_N + P\mathbf{x}_N \geq S\mathbf{x}_{d,N} & \\
 & \boldsymbol{\varepsilon}^x_N - P\mathbf{x}_N \geq -S\mathbf{x}_{d,N} & \\
 & \boldsymbol{\varepsilon}^u_k + R\mathbf{u}_k \geq 0 & k = 0, 1, \ldots, N-1\\
 & \boldsymbol{\varepsilon}^u_k - R\mathbf{u}_k \geq 0 & k = 0, 1, \ldots, N-1\\
 & \mathbf{u}_k \leq \mathbf{1}u_{max} & k = 0, 1, 2, \ldots, N-1\\
 & \mathbf{u}_k \geq -\mathbf{1}u_{max} & k = 0, 1, 2, \ldots, N-1,
\end{array} \tag{39}$$

where the x_e terms have been expanded using the relationship $\mathbf{x}_e = \mathbf{x} - \mathbf{x}_d$. To be implemented as a linear problem, the following batch procession formulation is used. The L$_1$ norm optimal control has an objective function of

$$J_1(\boldsymbol{\xi}_1) = \mathbf{q}_1^T \boldsymbol{\xi}_1 \tag{40}$$

with

$$\mathbf{q}_1 = \left[\, \mathbf{0}_{1\times nN} \,\vert\, \mathbf{0}_{1\times mN} \,\vert\, \mathbf{1}_{1\times nN} \,\vert\, \mathbf{1}_{1\times mN} \,\right]. \tag{41}$$

The control variables are

$$\xi_1 = \left[\mathbf{x}_1^T \quad \ldots \quad \mathbf{x}_N^T \quad \mathbf{u}_0^T \quad \ldots \quad \mathbf{u}_{N-1}^T \quad (\boldsymbol{\varepsilon}^x_1)^T \quad \ldots \right.$$
$$\left. (\boldsymbol{\varepsilon}^x_N)^T \quad (\boldsymbol{\varepsilon}^u_0)^T \quad \ldots \quad (\boldsymbol{\varepsilon}^u_{N-1})^T \right]^T. \tag{42}$$

The overall constraint formulation

$$\begin{aligned}
\mathbf{b}^x_{1,lb} &\leq \mathbf{A}^x_1 \boldsymbol{\xi}_1 \leq \mathbf{b}^x_{1,ub}\\
\mathbf{b}^u_{1,lb} &\leq \mathbf{A}^u_1 \boldsymbol{\xi}_1 \leq \mathbf{b}^u_{1,ub}
\end{aligned} \tag{43}$$

is functionally identical to the L$_\infty$ norm but with the matrices defined as

$$\mathbf{A}^x_1 = \left[\begin{array}{cccc|c|cccc|c}
\mathbf{Q}_1 & \mathbf{0}_{n\times n} & \ldots & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times mN} & \mathbf{I}_{n\times n} & \mathbf{0}_{n\times n} & \ldots & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times mN}\\
-\mathbf{Q}_1 & \mathbf{0}_{n\times n} & \ldots & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times mN} & \mathbf{I}_{n\times n} & \mathbf{0}_{n\times n} & \ldots & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times mN}\\
\mathbf{0}_{n\times n} & \mathbf{Q}_1 & \ldots & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times mN} & \mathbf{0}_{n\times n} & \mathbf{I}_{n\times n} & \ldots & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times mN}\\
\mathbf{0}_{n\times n} & -\mathbf{Q}_1 & \ldots & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times mN} & \mathbf{0}_{n\times n} & \mathbf{I}_{n\times n} & \ldots & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times mN}\\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\
\mathbf{0}_{n\times n} & \mathbf{0}_{n\times n} & \ldots & \mathbf{P}_1 & \mathbf{0}_{n\times mN} & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times n} & \ldots & \mathbf{I}_{n\times n} & \mathbf{0}_{n\times mN}\\
\mathbf{0}_{n\times n} & \mathbf{0}_{n\times n} & \ldots & -\mathbf{P}_1 & \mathbf{0}_{n\times mN} & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times n} & \ldots & \mathbf{I}_{n\times n} & \mathbf{0}_{n\times mN}
\end{array}\right] \tag{44}$$

$$\mathbf{A}^u_1 =$$
$$\left[\begin{array}{c|cccc|c|cccc}
\mathbf{0}_{m\times nN} & \mathbf{R}_1 & \mathbf{0}_{m\times m} & \ldots & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times nN} & \mathbf{I}_{m\times m} & \mathbf{0}_{m\times m} & \ldots & \mathbf{0}_{m\times m}\\
\mathbf{0}_{m\times nN} & -\mathbf{R}_1 & \mathbf{0}_{m\times m} & \ldots & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times nN} & \mathbf{I}_{m\times m} & \mathbf{0}_{m\times m} & \ldots & \mathbf{0}_{m\times m}\\
\mathbf{0}_{m\times nN} & \mathbf{0}_{m\times m} & \mathbf{R}_1 & \ldots & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times nN} & \mathbf{0}_{m\times m} & \mathbf{I}_{m\times m} & \ldots & \mathbf{0}_{m\times m}\\
\mathbf{0}_{m\times nN} & \mathbf{0}_{m\times m} & -\mathbf{R}_1 & \ldots & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times nN} & \mathbf{0}_{m\times m} & \mathbf{I}_{m\times m} & \ldots & \mathbf{0}_{m\times m}\\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots\\
\mathbf{0}_{m\times nN} & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} & \ldots & \mathbf{R}_1 & \mathbf{0}_{m\times nN} & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} & \ldots & \mathbf{I}_{m\times m}\\
\mathbf{0}_{m\times nN} & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} & \ldots & -\mathbf{R} & \mathbf{0}_{m\times nN} & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} & \ldots & \mathbf{I}_{m\times m}
\end{array}\right] \tag{45}$$

$$\mathbf{b}^x_{1,ub} = \left[\, \infty\ \infty \ldots \infty \,\right]^T \tag{46}$$

$$\mathbf{b}^x_{1,lb} = \left[\begin{array}{c} (\mathbf{Q}_1\mathbf{x}_{d,1})^T\ (-\mathbf{Q}_1\mathbf{x}_{d,1})^T\ \ldots\ (\mathbf{P}_1\mathbf{x}_{d,N})^T\\ (-\mathbf{P}_1\mathbf{x}_{d,N})^T \end{array} \right]^T \tag{47}$$

$$\mathbf{b}^u_{1,ub} = \left[\, \infty\ \infty \ldots \infty \,\right]^T \tag{48}$$

$$\mathbf{b}^u_{1,lb} = \left[\, 0\ 0 \ldots 0 \,\right]^T. \tag{49}$$

The weighting matrices $\mathbf{Q}_1$, $\mathbf{R}_1$, and $\mathbf{P}_1$ are initialized identically to the weighting matrices in the L$_\infty$ formulation.

The dynamic and boundary constraints are the same as the L$_\infty$ matrices but with the final columns of zeros sized to match the new input matrix.

$$
\mathbf{A}_{1,dyn} = \left[A_{1,1}\ A_{1,2}\ A_{1,3}\ A_{1,4}\right]
$$

$$
A_{1,1} = \begin{bmatrix} -\mathbf{I}_{n\times n} & \mathbf{0}_{n\times n} & \cdots & \cdots & \mathbf{0}_{n\times n} \\ \mathbf{A} & -\mathbf{I}_{n\times n} & \mathbf{0}_{n\times n} & \cdots & \vdots \\ \mathbf{0}_{n\times n} & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \mathbf{0}_{n\times n} \\ \mathbf{0}_{n\times n} & \cdots & \mathbf{0}_{n\times n} & \mathbf{A} & -\mathbf{I}_{n\times n} \end{bmatrix}
$$

$$
A_{1,2} = \begin{bmatrix} \mathbf{B} & \mathbf{0}_{n\times m} & \cdots & \cdots & \mathbf{0}_{n\times m} \\ \mathbf{0}_{n\times m} & \mathbf{B} & & & \vdots \\ \vdots & & \ddots & & \vdots \\ \vdots & & & \ddots & \mathbf{0}_{n\times m} \\ \mathbf{0}_{n\times m} & \cdots & \cdots\ \mathbf{0}_{n\times m} & & \mathbf{B} \end{bmatrix}
$$

$$
A_{1,3} = \begin{bmatrix} \mathbf{0}_{n\times nN} \\ \mathbf{0}_{n\times nN} \\ \cdots \\ \cdots \\ \mathbf{0}_{n\times nN} \end{bmatrix}
$$

$$
A_{1,4} = \begin{bmatrix} \mathbf{0}_{n\times mN} \\ \mathbf{0}_{n\times mN} \\ \cdots \\ \cdots \\ \mathbf{0}_{n\times mN} \end{bmatrix} \tag{50}
$$

$$
\mathbf{b}_{1,dyn} = \left[\, (-\mathbf{A}\mathbf{x}_0)^T\ \mathbf{0}^T_{n\times 1} \ \ldots\ \mathbf{0}^T_{n\times 1}\, \right]^T \tag{51}
$$

$$
\mathbf{A}_{1,bound} = \left[\begin{array}{cccc|ccc|c|c} \mathbf{A}_{poly,1} & \mathbf{0}_{?\times n} & \cdots & \mathbf{0}_{?\times n} & \mathbf{0}_{?\times n} & \cdots & \mathbf{0}_{?\times n} & \mathbf{0}_{?\times nN} & \mathbf{0}_{?\times mN} \\ \mathbf{0}_{?\times n} & \mathbf{A}_{poly,2} & \cdots & \mathbf{0}_{?\times n} & \mathbf{0}_{?\times n} & \cdots & \mathbf{0}_{?\times n} & \mathbf{0}_{?\times nN} & \mathbf{0}_{?\times mN} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0}_{?\times n} & \mathbf{0}_{?\times n} & \cdots & \mathbf{A}_{poly,N} & \mathbf{0}_{?\times n} & \cdots & \mathbf{0}_{?\times n} & \mathbf{0}_{?\times nN} & \mathbf{0}_{?\times mN} \end{array}\right] \tag{52}
$$

$$
\mathbf{b}_{1,bound} = \left[\, b_1\ b_2\ \ldots\ b_N\, \right]^T . \tag{53}
$$

Since each $\mathbf{A}_{poly,k}$ polytope boundary matrix can have a different number of rows, each matrix of zeros in $\mathbf{A}_{1,bound}$ has a variable number of rows, as indicated by the “?" subscript.

A.3 L_2 norm formulation

The derivation and mathematical expression for propellant usage is traditionally based on the L_2 norm as shown below

$$
\dot{m} = \frac{||\ \mathbf{T}\ ||_2}{V_e}, \tag{54}
$$

where $\dot{m}$ is the propellant mass flow rate, $\mathbf{T}$ is the thrust vector, or the control input $\mathbf{u}$ shown in Eq. (1), and V_e is the exhaust velocity of the engine [42]. The overall propellant usage of the engine for a given maneuver can be calculated by

$$
\Delta m = \int_{t_0}^{t_f} \dot{m}\ dt = \int_{t_0}^{t_f} \frac{||\ \mathbf{T}\ ||_2}{V_e}\ dt. \tag{55}
$$

Given an initial state of x_0, and a time horizon of N intervals, the quadratic objective function can be written as

$$
J(x,u) = \frac{1}{2} u_0^T R u_0 + \frac{1}{2} \sum_{k=1}^{N-1} \left[x_k^T Q x_k + u_k^T R u_k \right] + \frac{1}{2} x_N^T P x_N \tag{56}
$$

subject to

$$
x_{k+1} = A x_k + B u_k, \tag{57}
$$

where R, Q, and P are the weightings on control usage, state error, and terminal error, respectively. This objective function will regulate x to zero. A small change to the standard formulation used her in is driving the state to a desired value instead of the origin. The error state is defined as

$$
x_e = x - x_d. \tag{58}
$$

The desired trajectory x_d is the relative state, converted to the LVLH frame. The desired state for each agent is updated using the HCW equations. Using the error state, the objective function is updated to

$$
J(x,u) = \frac{1}{2} u_0^T R u_0 + \frac{1}{2} \sum_{k=1}^{N-1} \left[x_{e,k}^T Q x_{e,k} + u_k^T R u_k \right] + \frac{1}{2} x_N^T P x_{e,N}; \tag{59}
$$

assuming that the cost matrices are symmetrical, Eq. (59) can be expressed as

$$
\begin{aligned} J(x,u) &= \frac{1}{2} u_0^T R u_0 \\ &+ \frac{1}{2} \sum_{k=1}^{N-1} \left[x_k^T Q x_k - 2 x_{d,k}^T Q x_k + x_{d,k}^T Q x_{d,k} + u_k^T R u_k \right] \\ &+ \frac{1}{2} \left(x_N^T P x_N - 2 x_{d,N}^T P x_N + x_{d,N}^T P x_{d,N} \right). \end{aligned} \tag{60}
$$

Since any constant terms will only change the overall magnitude of the objective function and not the location of the minimum, the constants can be ignored. This results in

$$J_2(x,u) = \frac{1}{2}u_0^T R u_0 + \frac{1}{2}\sum_{k=1}^{N-1}\left[x_k^T Q x_k - 2x_{d,k}^T Q x_k + u_k^T R u_k\right] + \frac{1}{2}\left(x_N^T P x_N - 2x_{d,N}^T P x_N\right). \tag{61}$$

This results in the optimization problem with the polytope constraint

$$\begin{aligned} \min_{x,u} \quad & \frac{1}{2}u_0^T R u_0 + \frac{1}{2}\sum_{k=1}^{N-1}\left[x_k^T Q x_k - 2x_{d,k}^T Q x_k + u_k^T R u_k\right] \\ & + \frac{1}{2}\left(x_N^T P x_N - 2x_{d,N}^T P x_N\right) \\ \text{s.t.} \quad & x_{k+1} = \mathbf{A}_D \mathbf{x}_k + \mathbf{B}_D \mathbf{u}_k \qquad k = 1, 2, \ldots, N \\ & \mathbf{A}_{poly,k}\mathbf{x}_k \le \mathbf{b}_{poly,k} \qquad k = 1, 2, \ldots, N \\ & \mathbf{u}_k \le \mathbf{1}u_{max} \qquad k = 0, 1, 2, \ldots, N-1 \\ & \mathbf{u}_k \ge -\mathbf{1}u_{max} \qquad k = 0, 1, 2, \ldots, N-1. \end{aligned} \tag{62}$$

The constraints ensure the dynamics are followed, keep the spacecraft in the polytope, and limit the maximum and minimum control accelerations. If the state is modified to be a combination $\boldsymbol{\xi}_2$ of the states and controls as

$$\boldsymbol{\xi}_2 = \left[\mathbf{x}_1\ \mathbf{x}_2\ \ldots\ \mathbf{x}_N\ \mathbf{u}_0\ \mathbf{u}_1\ \ldots\ \mathbf{u}_{N-1}\right]^T, \tag{63}$$

where N is the number of steps in the MPC time horizon, and then, the objective function can be rewritten as

$$J_2(\boldsymbol{\xi}_2) = \frac{1}{2}\boldsymbol{\xi}_2^T H \boldsymbol{\xi}_2 + \mathbf{q}_2^T \boldsymbol{\xi}_2. \tag{64}$$

The quadratic cost matrix $\mathbf{H}$ is defined to be

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{x,1} & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & \mathbf{H}_{x,2} & & & & & & \vdots \\ \vdots & & \ddots & & & & & \vdots \\ \vdots & & & \mathbf{H}_{x,N} & & & & \vdots \\ \vdots & & & & \mathbf{H}_{u,0} & & & \vdots \\ \vdots & & & & & \mathbf{H}_{u,1} & & \vdots \\ \vdots & & & & & & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \mathbf{H}_{u,N-1} \end{bmatrix}, \tag{65}$$

and the linear cost vector $\mathbf{q}_2$ is defined as

$$\mathbf{q}_2 = \left[\mathbf{c}_1\ \ldots\ \mathbf{c}_N\ 0\ \ldots\ 0\right]^T \tag{66}$$

with

$$\begin{aligned} \mathbf{H}_{x,k} &= \mathbf{Q} \qquad k = 1, \ldots, N-1 \\ \mathbf{H}_{x,N} &= \mathbf{P} \\ \mathbf{H}_{u,k} &= \mathbf{R} \qquad k = 0, \ldots, N-1 \\ \mathbf{c}_k &= -\mathbf{H}_{x,k}\mathbf{x}_{d,k} \qquad k = 1, \ldots, N. \end{aligned} \tag{67}$$

Given ξ_2, the dynamic constraints can be formulated as

$$\mathbf{A}_{eq}\xi_2 = b_{eq} \tag{68}$$

with

$$\mathbf{A}_{eq} = \begin{bmatrix} -\mathbf{I} & 0 & \ldots\ldots & 0 & \mathbf{B} & 0 & \ldots\ldots & 0 \\ \mathbf{A} & -\mathbf{I} & 0 \ \ldots & \vdots & 0 & \mathbf{B} & & \vdots \\ 0 & \ddots & \ddots & \vdots & \vdots & & \ddots & \vdots \\ \vdots & & \ddots \ \ddots & 0 & \vdots & & \ddots & 0 \\ 0 & \ldots & 0 \ \ \mathbf{A} & -\mathbf{I} & 0 & \ldots\ldots & 0 & \mathbf{B} \end{bmatrix} \tag{69}$$

$$\mathbf{b}_{eq} = \left[-\mathbf{A}\mathbf{x}_0\ 0\ \ldots\ 0\right]^T, \tag{70}$$

where $\mathbf{A}$ and $\mathbf{B}$ are the state transition and control matrices, respectively, as described in Sect. 2.1, and $\mathbf{x}_0$ is the initial state of the system.

A.4 L_∞ norm formulation

The L_∞ norm optimal control has an objective function of

$$J_\infty = \varepsilon_{1,\infty}^x + \ldots + \varepsilon_{N,\infty}^x + \ldots + \varepsilon_{0,\infty}^u + \ldots + \varepsilon_{N-1,\infty}^u \tag{71}$$

with the constraints

$$\begin{aligned} -\mathbf{1}\varepsilon_{k,\infty}^x &\le Q\mathbf{x}_{e,k}, \ \ k = 1, 2, \ldots, N-1 \\ -\mathbf{1}\varepsilon_{k,\infty}^x &\le -Q\mathbf{x}_{e,k}, \ \ k = 1, 2, \ldots, N-1 \\ -\mathbf{1}\varepsilon_{N,\infty}^x &\le P\mathbf{x}_{e,N} \\ -\mathbf{1}\varepsilon_{N,\infty}^x &\le -P\mathbf{x}_{e,N} \\ -\mathbf{1}\varepsilon_{k,\infty}^u &\le R\mathbf{u}_k \ k = 0, 1, \ldots, N-1 \\ -\mathbf{1}\varepsilon_{k,\infty}^u &\le -R\mathbf{u}_k \ k = 0, 1, \ldots, N-1, \end{aligned} \tag{72}$$

where $\mathbf{1} = \left[1\ 1\ \ldots\ 1\right]^T$, all inequalities are evaluated element-wise, and each ε_∞ is a scalar value. This works similarly to the L_1 formulation except that since the ε values are now scalars, only the weighted error or control with the highest magnitude is counted, in accordance with the L_∞ definition. Substituting in the definition for x_e and rearranging give

$$
\begin{aligned}
&\mathbf{1}\varepsilon^x_{k,\infty} + Q\mathbf{x}_k \geq Q\mathbf{x}_{d,k} \; k = 1, 2, \ldots, N-1\\
&\mathbf{1}\varepsilon^x_{k,\infty} - Q\mathbf{x}_k \geq -Q\mathbf{x}_{d,k} \; k = 1, 2, \ldots, N-1\\
&\mathbf{1}\varepsilon^x_{N,\infty} + S\mathbf{x}_N \geq S\mathbf{x}_{d,N}\\
&\mathbf{1}\varepsilon^x_{N,\infty} - S\mathbf{x}_N \geq -S\mathbf{x}_{d,N}\\
&\mathbf{1}\varepsilon^u_{k,\infty} + R\mathbf{u}_k \geq 0 \; k = 0, 1, \ldots, N-1\\
&\mathbf{1}\varepsilon^u_{k,\infty} - R\mathbf{u}_k \geq 0 \; k = 0, 1, \ldots, N-1.
\end{aligned}
\tag{73}
$$

The full optimization formulation with the polytope constraint for the L$_\infty$ norm is then

$$
\begin{aligned}
\min_{\varepsilon_\infty} \quad & \varepsilon^x_{1,\infty} + \ldots + \varepsilon^x_{N,\infty} + \ldots + \varepsilon^u_{0,\infty} + \ldots + \varepsilon^u_{N-1,\infty}\\
\text{s.t.} \quad & x_{k+1} = \mathbf{A}_D\mathbf{x}_k + \mathbf{B}_D\mathbf{u}_k \qquad k = 1, 2, \ldots, N\\
& \mathbf{A}_{poly,k}\mathbf{x}_k \leq \mathbf{b}_{poly,k} \qquad k = 1, 2, \ldots, N\\
& \mathbf{1}\varepsilon^x_{k,\infty} + Q\mathbf{x}_k \geq Q\mathbf{x}_{d,k} \qquad k = 1, 2, \ldots, N-1\\
& \mathbf{1}\varepsilon^x_{k,\infty} - Q\mathbf{x}_k \geq -Q\mathbf{x}_{d,k} \qquad k = 1, 2, \ldots, N-1\\
& \mathbf{1}\varepsilon^x_{N,\infty} + S\mathbf{x}_N \geq S\mathbf{x}_{d,N}\\
& \mathbf{1}\varepsilon^x_{N,\infty} - S\mathbf{x}_N \geq -S\mathbf{x}_{d,N}\\
& \mathbf{1}\varepsilon^u_{k,\infty} + R\mathbf{u}_k \geq 0 \qquad k = 0, 1, \ldots, N-1\\
& \mathbf{1}\varepsilon^u_{k,\infty} - R\mathbf{u}_k \geq 0 \qquad k = 0, 1, \ldots, N-1\\
& \mathbf{u}_k \leq \mathbf{1}u_{max} \qquad k = 0, 1, 2, \ldots, N-1\\
& \mathbf{u}_k \geq -\mathbf{1}u_{max} \qquad k = 0, 1, 2, \ldots, N-1.
\end{aligned}
\tag{74}
$$

The batch processing formulation becomes

$$\mathbf{J}_\infty(\boldsymbol{\xi}_\infty) = \mathbf{q}^T_\infty \boldsymbol{\xi}_\infty, \tag{75}$$

where

$$\mathbf{q}_\infty = \left[\, \mathbf{0}_{1\times nN} \,\vdots\, \mathbf{0}_{1\times mN} \,\vdots\, \mathbf{1}_{1\times N} \,\vdots\, \mathbf{1}_{1\times N} \,\right]^T, \tag{76}$$

and the control variables are

$$\boldsymbol{\xi}_\infty = \left[\begin{array}{ccc|ccc|cc|}
\mathbf{x}_1^T & \ldots & \mathbf{x}_N^T & \mathbf{u}_0^T & \ldots & \mathbf{u}_{N-1}^T & (\varepsilon^x_1)^T & \ldots \\
 & (\varepsilon^x_N)^T & (\varepsilon^u_0)^T & \ldots & (\varepsilon^u_{N-1})^T & & &
\end{array}\right]^T. \tag{77}$$

The constraints for ε are formulated as

$$
\begin{aligned}
&\mathbf{b}^x_{\infty,lb} \leq \mathbf{A}^x_\infty \boldsymbol{\xi}_\infty \leq \mathbf{b}^x_{\infty,ub}\\
&\mathbf{b}^u_{\infty,lb} \leq \mathbf{A}^u_\infty \boldsymbol{\xi}_\infty \leq \mathbf{b}^u_{\infty,ub},
\end{aligned}
\tag{78}
$$

where

$$\mathbf{A}^x_\infty = \left[\begin{array}{cccc|c|cccc|c}
\mathbf{Q}_\infty & \mathbf{0}_{n\times n} & \ldots & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times mN} & \mathbf{1} & \mathbf{0}_{n\times 1} & \ldots & \mathbf{0}_{n\times 1} & \mathbf{0}_{n\times N}\\
-\mathbf{Q}_\infty & \mathbf{0}_{n\times n} & \ldots & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times mN} & \mathbf{1} & \mathbf{0}_{n\times 1} & \ldots & \mathbf{0}_{n\times 1} & \mathbf{0}_{n\times N}\\
\mathbf{0}_{n\times n} & \mathbf{Q}_\infty & \ldots & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times mN} & \mathbf{0}_{n\times 1} & \mathbf{1} & \ldots & \mathbf{0}_{n\times 1} & \mathbf{0}_{n\times N}\\
\mathbf{0}_{n\times n} & -\mathbf{Q}_\infty & \ldots & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times mN} & \mathbf{0}_{n\times 1} & \mathbf{1} & \ldots & \mathbf{0}_{n\times 1} & \mathbf{0}_{n\times N}\\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\
\mathbf{0}_{n\times n} & \mathbf{0}_{n\times n} & \ldots & \mathbf{S}_\infty & \mathbf{0}_{n\times mN} & \mathbf{0}_{n\times 1} & \mathbf{0}_{n\times 1} & \ldots & \mathbf{1} & \mathbf{0}_{n\times N}\\
\mathbf{0}_{n\times n} & \mathbf{0}_{n\times n} & \ldots & -\mathbf{S}_\infty & \mathbf{0}_{n\times mN} & \mathbf{0}_{n\times 1} & \mathbf{0}_{n\times 1} & \ldots & \mathbf{1} & \mathbf{0}_{n\times N}
\end{array}\right] \tag{79}$$

$$\mathbf{A}^u_\infty = \left[\begin{array}{c|cccc|c|cccc}
\mathbf{0}_{m\times nN} & \mathbf{R}_\infty & \mathbf{0}_{m\times m} & \ldots & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times nN} & \mathbf{1} & \mathbf{0}_{m\times 1} & \ldots & \mathbf{0}_{m\times 1}\\
\mathbf{0}_{m\times nN} & -\mathbf{R}_\infty & \mathbf{0}_{m\times m} & \ldots & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times nN} & \mathbf{1} & \mathbf{0}_{m\times 1} & \ldots & \mathbf{0}_{m\times 1}\\
\mathbf{0}_{m\times nN} & \mathbf{0}_{m\times m} & \mathbf{R}_\infty & \ldots & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times nN} & \mathbf{0}_{m\times 1} & \mathbf{1} & \ldots & \mathbf{0}_{m\times 1}\\
\mathbf{0}_{m\times nN} & \mathbf{0}_{m\times m} & -\mathbf{R}_\infty & \ldots & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times nN} & \mathbf{0}_{m\times 1} & \mathbf{1} & \ldots & \mathbf{0}_{m\times 1}\\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots\\
\mathbf{0}_{m\times nN} & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} & \ldots & \mathbf{R}_\infty & \mathbf{0}_{m\times nN} & \mathbf{0}_{m\times 1} & \mathbf{0}_{m\times 1} & \ldots & \mathbf{1}\\
\mathbf{0}_{m\times nN} & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} & \ldots & -\mathbf{R}_\infty & \mathbf{0}_{m\times nN} & \mathbf{0}_{m\times 1} & \mathbf{0}_{m\times 1} & \ldots & \mathbf{1}
\end{array}\right]. \tag{80}$$

The state dimension is represented as n with m representing the dimension size of the control input with $n = 6$ and $m = 3$ for this analysis. Additionally, the upper and lower boundaries for the constraints are

$$\mathbf{b}^x_{\infty,ub} = \left[\infty \; \infty \; \ldots \; \infty\right]^T \tag{81}$$

$$\mathbf{b}^x_{\infty,lb} = \left[\begin{array}{ccc} (\mathbf{Q}_\infty\mathbf{x}_{d,1})^T & (-\mathbf{Q}_\infty\mathbf{x}_{d,1})^T & \ldots \\ & (\mathbf{S}_\infty\mathbf{x}_{d,N})^T & (-\mathbf{S}_\infty\mathbf{x}_{d,N})^T \end{array}\right]^T \tag{82}$$

$$\mathbf{b}^u_{\infty,ub} = \left[\infty \; \infty \; \ldots \; \infty\right]^T \tag{83}$$

$$\mathbf{b}^u_{\infty,lb} = \left[0 \; 0 \; \ldots \; 0\right]^T. \tag{84}$$

The dynamic and boundary constraints are the same as previously stated but expanded with zeros to match the new input vector $\boldsymbol{\xi}_\infty$

$$\mathbf{A}_{\infty,dyn} = \left[\begin{array}{ccccc:ccccc:c:c} -\mathbf{I}_{n\times n} & \mathbf{0}_{n\times n} & \dots & \dots & \mathbf{0}_{n\times n} & \mathbf{B} & \mathbf{0}_{n\times m} & \dots & \dots & \mathbf{0}_{n\times m} & \mathbf{0}_{n\times N} & \mathbf{0}_{n\times N} \\ \mathbf{A} & -\mathbf{I}_{n\times n} & \mathbf{0}_{n\times n} & \dots & \vdots & \mathbf{0}_{n\times m} & \mathbf{B} & & & \vdots & \mathbf{0}_{n\times N} & \mathbf{0}_{n\times N} \\ \mathbf{0}_{n\times n} & \ddots & \ddots & & \vdots & \vdots & & \ddots & & \vdots & \vdots & \vdots \\ \vdots & & \ddots & \ddots & \mathbf{0}_{n\times n} & \vdots & & & \ddots & \mathbf{0}_{n\times m} & \vdots & \vdots \\ \mathbf{0}_{n\times n} & \dots & \mathbf{0}_{n\times n} & \mathbf{A} & -\mathbf{I}_{n\times n} & \mathbf{0}_{n\times m} & \dots & \dots & \mathbf{0}_{n\times m} & \mathbf{B} & \mathbf{0}_{n\times N} & \mathbf{0}_{n\times N} \end{array}\right] \tag{85}$$

$$\mathbf{b}_{\infty,dyn} = \left[-\mathbf{A}\mathbf{x}_0 \; 0 \ldots 0 \right]^T \tag{86}$$

$$\mathbf{A}_{\infty,bound} = \left[\begin{array}{cccc:ccc:c:c} \mathbf{A}_{poly,1} & \mathbf{0}_{?\times n} & \dots & \mathbf{0}_{?\times n} & \mathbf{0}_{?\times n} & \dots & \mathbf{0}_{?\times n} & \mathbf{0}_{?\times N} & \mathbf{0}_{?\times N} \\ \mathbf{0}_{?\times n} & \mathbf{A}_{poly,2} & \dots & \mathbf{0}_{?\times n} & \mathbf{0}_{?\times n} & \dots & \mathbf{0}_{?\times n} & \mathbf{0}_{?\times N} & \mathbf{0}_{?\times N} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0}_{?\times n} & \mathbf{0}_{?\times n} & \dots & \mathbf{A}_{poly,N} & \mathbf{0}_{?\times n} & \dots & \mathbf{0}_{?\times n} & \mathbf{0}_{?\times N} & \mathbf{0}_{?\times N} \end{array}\right] \tag{87}$$

$$b_{\infty,bound} = \left[b_1 \; b_2 \ldots b_N \right]^T . \tag{88}$$

As previously mentioned, each $\mathbf{A}_{poly,k}$ polytope boundary matrix can have a different number of rows, each matrix of zeros in $\mathbf{A}_{\infty,bound}$ has a variable number of rows, as indicated by the “?" subscript.

A.5 Polytope constraint

For each point along the desired trajectory a volume of space is defined. The full boundary constraint is defined as

$$\mathbf{A}_{bound}\xi \leq \mathbf{b}_{bound}, \tag{89}$$

where

$$\mathbf{A}_{bound} = \begin{bmatrix} \mathbf{A}_{poly,1} & 0 & \dots & 0 & 0 \dots 0 \\ 0 & \mathbf{A}_{poly,2} & \dots & 0 & 0 \dots 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \ddots \vdots \\ 0 & 0 & \dots & \mathbf{A}_{poly,N} & 0 \dots 0 \end{bmatrix} \tag{90}$$

$$\mathbf{b}_{bound} = \left[\mathbf{b}^T_{poly,1} \; \mathbf{b}^T_{poly,2} \ldots \mathbf{b}^T_{poly,N} \right]^T \tag{91}$$

with $\mathbf{A}_{poly,k}$ and $\mathbf{b}_{poly,k}$ referring to the polytope corresponding to the k$^{\text{th}}$ point along the desired trajectory. The $\mathbf{A}_{poly,k}$ matrix is padded with zeros to properly match the dimensions of ξ. Also, note that while each $\mathbf{A}_{poly,k}$ has the same number of columns, the number of rows is free to vary as long as each $\mathbf{A}_{poly,k}$ matches its corresponding $\mathbf{b}_{poly,k}$. This allows the polytope at each time-step k to vary with the only requirement being that the polytope is convex.

References

1. D’Amico S, Ardaens JS, De Florio S (2013) Autonomous formation flying based on GPS - PRISMA flight results. Acta Astronaut 82(1):69–79. https://doi.org/10.1016/j.actaastro.2012.04.033
2. Fridlund CVM (2000) Darwin-The Infrared Space Interferometry Mission. In: ESA Bulletin 103, pp. 20–63
3. Mesbahi M, Hadaegh FY (2001) Formation flying control of multiple spacecraft via graphs matrix inequalities and switching. J Guid Control Dyn 24(2):369–377. https://doi.org/10.2514/2.4721
4. Ticker RL, Azzolini JD (2000) 2000 survey of distributed spacecraft technologies and architectures for NASA’s Earth Science Enterprise in the 2010–2025 Time Frame. Technical Report 209964, NASA
5. Martin M, Klupar P, Kilberg S, Winter J (2001) TechSat 21 and Revolutionizing Space Missions using Microsatellites. USU/AIAA Small Satellite Conference
6. Bristow J, Folta D, Hartman K (2000) A formation flying technology vision. Space 2000 Conference and Exhibition. https://doi.org/10.2514/6.2000-5194
7. Scharf DP, Hadaegh FY, Ploen SR (2004) A survey of spacecraft formation flying guidance and control (part II): control. Proc Am Control Conf 4(Part 11):2976–2985. https://doi.org/10.23919/acc.2004.1384365
8. Krieger G, Moreira A, Fiedler H, Hajnsek I, Werner M, Younis M, Zink M (2007) TanDEM-X: a satellite formation for high-resolution SAR interferometry. IEEE Trans Geosci Remote Sens 45(11):3317–3340. https://doi.org/10.1109/TGRS.2007.900693
9. Fiedler H, Krieger G, Werner M, Reiniger K (2006) The TanDEM-X mission design and data acquisition plan. In: European Conference on Synthetic Aperture Radar 2006, pp. 16–18
10. Roth NH, Risi B, Grant CC, Zee RE (2016) Flight results from the CanX-4 and CanX-5 formation flying mission. In: 29th Annual AIAA/USU Conference on Small Satellites, pp. 1–15. https://doi.org/10.1016/j.jcrysgro.2014.02.053
11. Tapley BD, Bettadpur S, Watkins M, Reigber C (2004) The gravity recovery and climate experiment: mission overview and early results. Geophys Res Lett 31(9):1–4. https://doi.org/10.1029/2004GL019920
12. Bertiger W, Bar-Sever Y, Bettadpur S, Dunn C (2002) GRACE: millimeters and microns in orbit. In: ION GPS 2002 Conference, pp. 2022–2029

13. Farahmand M, Long A, Hollister J, Rose J, Godine D (2018) Magnetospheric multiscale mission navigation performance during apogee-raising and beyond. Adv Astronaut Sci 162:2738–2739
14. Winternitz LB, Bamford WA, Price SR, Carpenter JR, Long AC, Farahmand M (2017) Global positioning system navigation above 76,000 km for NASA'S magnetospheric multiscale mission. J Inst Navig 64(2):289–300. https://doi.org/10.1002/navi.198
15. Wang J, Zhang D, Chen G, Shao X (2018) A new fuel-balanced formation keeping reference trajectories planning method. Aircr Eng Aerosp Technol 90(6):927–936. https://doi.org/10.1108/aeat-03-2017-0084
16. Hakima H, Bazzocchi MCF (2022) In-orbit target tracking by flyby and formation-flying spacecraft. Aerosp Syst 5:197–212
17. Di Mauro G, Lawn M, Bevilacqua R (2018) Survey on guidance navigation and control requirements for spacecraft formation-flying missions. J Guid Control Dyn 41(3):581–602. https://doi.org/10.2514/1.G002868
18. Li J, Xi X (2012) Fuel-optimal low-thrust reconfiguration of formation- flying satellites via homotopic approach. J Guid Control Dyn 35(6):1709–1717. https://doi.org/10.2514/1.57354
19. Mazal L, Gurfil P (2013) Cluster flight algorithms for disaggregated satellites. J Guid Control Dyn 36(1):124–135. https://doi.org/10.2514/1.57180
20. Rocco EM, De Oliveira E Souza ML, De Almeida Prado AFB (2013) Station keeping of constellations using multi-objective strategies. Math Probl Eng 2(1). https://doi.org/10.1155/2013/476451
21. Chernick M, D'Amico S (2018) New closed-form solutions for optimal impulsive control of spacecraft relative motion. J Guid Control Dyn 41(2):301–319. https://doi.org/10.2514/1.G002848
22. Breger LS (2004) Model predictive control for formation flying spacecraft. Master's thesis, Massachusetts Institute of Technology
23. Lim Y, Jung Y, Bang H (2018) Robust model predictive control for satellite formation keeping with eccentricity/inclination vector separation. Adv Space Res 61(10):2661–2672. https://doi.org/10.1016/j.asr.2018.02.036
24. Tillerson M, How JP (2002) Advanced guidance algorithms for spacecraft formation-keeping. Proc Am Control Conf 4:2830–2835. https://doi.org/10.1109/ACC.2002.1025218
25. Di Mauro G, Spiller D, Rafano Carnà SF, Bevilacqua R (2019) Minimum-fuel control strategy for spacecraft formation reconfiguration via finite-time maneuvers. J Guid Control Dyn 42(4):752–768. https://doi.org/10.2514/1.G003822
26. Wu B, Wang D, Poh EK, Xu G (2009) Nonlinear optimization of low-thrust trajectory for satellite formation: Legendre pseudospectral approach. J Guid Control Dyn 32(4):1371–1381. https://doi.org/10.2514/1.37675
27. Riano-Rios C, Bevilacqua R, Dixon WE (2020) Differential drag-based multiple spacecraft maneuvering and on-line parameter estimation using integral concurrent learning. Acta Astronaut 174(May):189–203. https://doi.org/10.1016/j.actaastro.2020.04.059
28. Thevenet J-B, Epenoy R (2008) Minimum-fuel deployment for spacecraft formations via optimal control. J Guid Control Dyn 31(1):101–113. https://doi.org/10.2514/1.30364
29. Takahashi Y, Sakamoto H, Sakai S-I (2022) Kinematics control of electromagnetic formation flight using angular-momentum conservation constraint. J Guid Control Dyn 45(2):280–295. https://doi.org/10.2514/1.G005873
30. Wang W, Wu D, Lei H, Baoyin H (2021) Fuel-optimal spacecraft cluster flight around an ellipsoidal asteroid. J Guid Control Dyn 44(10):1875–1882. https://doi.org/10.2514/1.G005993
31. Ariba Y, Arzelier D, Urbina-Iglesias LS (2018) Minimum-fuel fixed-time impulsive elliptic glide-slope guidance algorithms using semidefinite programming. J Guid Control Dyn 41(9):1873–1887. https://doi.org/10.2514/1.G003395
32. Massari M, Bernelli-Zazzera F (2009) Optimization of low-thrust reconfiguration maneuvers for spacecraft flying in formation. J Guid Control Dyn 32(5):1629–1638. https://doi.org/10.2514/1.37335
33. Chen T, Zhang D, Shao X (2020) Control parameters design of spacecraft formation flying via modified biogeography-based optimization. Aerosp Syst 3:1–8
34. Rawlings JB, Mayne DQ, Diehl MM (2019) Model predictive control: theory, computation, and design, 2nd edn. Nob Hill Publishing, Santa Barbara. https://doi.org/10.1155/2012/240898
35. Cheng B, Li Z (2019) Coordinated tracking control with asynchronous edge-based event-triggered communications. IEEE Trans Autom Control 64(10):4321–4328
36. Cheng B, Li Z (2019) Fully distributed event-triggered protocols for linear multi-agent networks. IEEE Trans Autom Control 64(4):1655–1662
37. Cheng B, Wu Z, Li Z (2019) Distributed edge-based event-triggered formation control. IEEE Trans Cybern 51:1241–1252
38. Hu Q, Shi Y (2020) Event-based coordinated control of spacecraft formation flying under limited communication. Nonlinear Dyn 99(3):2139–2159. https://doi.org/10.1007/s11071-019-05396-6
39. Ren W, Beard RW (2004) Decentralized scheme for spacecraft formation flying via the virtual structure approach. J Guid Control Dyn 27(1):73–82
40. Hill GW (1886) On the part of the motion of lunar perigee which is a function of the mean motions of the sun and moon. Acta Math 8:1–36. https://doi.org/10.1007/BF02417081
41. Wiltshire RS, Clohessy WH (1960) Terminal guidance system for satellite rendezvous. J Aerosp Sci 27(9):663–674. https://doi.org/10.2514/8.8704
42. Curtis H (2009) Orbital mechanics for engineering students, 2nd edn. Elsevier, Oxford
43. Borrelli F, Bemporad A, Morari M (2017) Predictive control for linear and hybrid systems. Cambridge University Press, Cambridge. https://doi.org/10.1017/9781139061759
44. Ross IM (2004) How to find minimum-fuel controllers. In: AIAA guidance, navigation and control conference and exhibit
45. Shorten R, Wirth F, Mason O, Wulff K, King C (2007) Stability criteria for switched and hybrid systems. SIAM Rev 49(4):545–592. https://doi.org/10.1137/05063516X
46. Sullivan J, Grimberg S, D'Amico S (2017) Comprehensive survey and assessment of spacecraft relative motion dynamics models. J Guid Control Dyn 40(8):1837–1859. https://doi.org/10.2514/1.G002309
47. Aguilera RP, Acuna P, Konstantinou G, Vazquez S, Leon JI (2018) Chapter 2 - Basic control principles in power electronics: analog and digital control design. In: Blaabjerg F (ed) Control of power electronic converters and systems. Academic Press, London, pp 31–68. https://doi.org/10.1016/B978-0-12-805245-7.00002-0
48. D'Amico S (2010) Autonomous formation flying in low earth orbit. PhD thesis, Delft University of Technology
49. Anon (1976) U.S. Standard Atmosphere, 1976. Technical report
50. Smith TK, Lewis Z, Olsen K, Bulcher MA, Whitmore SA (2022) A miniaturized, green, end-burning, and sandwich hybrid propulsion system. J Propuls Power 38(5):701–713. https://doi.org/10.2514/1.B38623
51. Hespanha JP (2018) Linear systems theory, 2nd edn. Princeton University Press, Princeton

CHAPTER 5

Satellite Cluster Flight Using Guidance Trajectory and Model Predictive Control

Satellite Cluster Flight Using Guidance Trajectory and Model Predictive Control

Tyson Smith[1*], John Akagi,[1*] and Greg Droge[2*]

[1]GN&C Engineer, Space Dynamics Laboratory, Logan, 84341, Utah, USA.

[2]Associate Professor, Electrical and Computer Engineering Department, Utah State University, Logan, 84322, Utah, USA.

*Corresponding author(s). E-mail(s): tyson.smith@sdl.usu.edu; john.akagi@sdl.usu.edu; greg.droge@usu.edu;

Abstract

This paper presents a model predictive control based framework that is used in conjunction with a polytope boundary constraint and the Hill-Clohessy-Wiltshire equations to maintain a desired formation of a cluster of spacecraft while also allowing freedom to maneuver within the allowable bounds. A fuel optimal guidance trajectory is generated and the model predictive control framework controls to this desired trajectory. The operational polytope boundaries enable the predictive framework of the model predictive control to be used to make small maneuvers to correct perturbations from its desired trajectory. The boundaries are designed such that no two agents have overlapping regions, allowing the vehicles to execute avoidance strategies without continually maintaining the trajectories of other agents. The model predictive control framework combined with the convex polytope boundary enables a scalable method that can support clusters of satellites in safely achieving mission objectives while minimizing fuel usage. As part of the implementation of this control scheme, this paper compares the fuel usage for a three spacecraft system. This work also compares the computation and fuel requirements for $\mathbf{L_1}$, $\mathbf{L_2}$, and $\mathbf{L_\infty}$ norm formulations of the framework.

Keywords: Formation Flying, Model Predictive Control, Switching Surfaces, Spacecraft Guidance

1 Introduction

The coordinated motion of satellites within a constellation or cluster, i.e., spacecraft formation flying, has been viewed as a key enabler for organizations like the European Space Agency (ESA) [1, 2], National Aeronautics and Space Administration (NASA) [1, 3, 4], and the Department of Defense (DoD) [5]. Formation flying technology is critical for a wide range of missions, including sparse antenna array missions, distributed sensing missions, advanced communications and internet access, spatial sampling for applications such as gravitational mapping, interferometric synthetic aperture radar, and target tracking [1, 6–15]. The idea of formation flying also allows for missions that are traditionally flown on a single large and expensive satellite, to be completed by several smaller and less expensive SmallSats or CubeSats [16]. Fundamental to space-based operations is the requirement to maintain relative motion characteristics while also limiting fuel usage. These factors must be considered while respecting the computational capabilities of the individual platforms and the communication restrictions of the cluster. This work develops an optimization-based feedback control strategy to allow each satellite to fly within a safe region of operation and maintain relative spacing with little to no inter-spacecraft communication required.

There are several challenges that arise in formation flying. These challenges include, but are not limited to: fuel consumption, overcoming disturbances or perturbing forces, data distribution between spacecraft, collision avoidance, mission operational constraints, and computation feasibility.

Creating a formation flying framework becomes an act of balancing priorities between meeting these different challenges or objectives. Previously, optimization strategies have been employed to reduce fuel consumption while maintaining relative motion, e.g., [17–23]. But while an optimal fuel saving maneuver may be desirable, feedback is essential for overcoming disturbances from the invariably erroneous motion models. Model predictive control (MPC) is an approach that repeatedly uses optimal control techniques within a feedback loop [24]. Tillerson et al., [23], designed an MPC scheme that used the Hill-Clohessy-Whitshire (HCW) and Lawden equations to define the relative motion. [23] used a six-sided error box as a keep-in volume constraint. Tillerson et al, found that, when using an error box as a constraint, it was possible to get initial conditions for the programming problem that results in higher fuel usage. This occurs when a satellite is near the corner of the error box with little room to maneuver and could result in an infeasible solution where the spacecraft requires more control than available to avoid leaving the error box. To alleviate this feasibility problem discussed by [23], our previous work, developed an approach that implemented a polytope approximation of a sphere [25]. However, the ΔV required is not suitable for small satellites. [22] developed an MPC strategy using Gauss Variational Equations (GVEs) with constraints defined directly on the eccentricity and inclination vectors of the D'Amico Relative Orbital Elements (ROEs). Constraining the ROE eccentricity and inclination vectors parallel to each other guarantees collision avoidance within an established formation, but it does not guarantee the formation is maintained nor does it guarantee a collision free transfer trajectory.

Another challenge that needs to be addressed in the area of spacecraft formation flight is data distribution between satellites. Inter-satellite communication can become a restrictive burden on satellites as it requires the antenna of each satellite to be aligned and the communication systems powered on. To reduce communication, [26–28] developed a strategy for using triggering events to switch from a coasting behavior to a controlled behavior, requiring communication only when vehicles pass some relative distance threshold. To lower communication even further, virtual structure techniques define the desired motion for each satellite based upon the ideal motion of a virtual satellite [21, 23, 25, 29]. The virtual satellite is a computational artifact used solely to define desired relative trajectories for the constellation [30]. By agreeing a-priori on the virtual satellite orbital characteristics, coordinated motion can be performed without communication.

The major contributions of this work are the formulation and evaluation of a virtual structure based MPC framework that guarantees flight safety and formation control while saving fuel and reducing communication. In particular, the amount of ΔV required to maintain the formation is reduced by combining a new optimal guidance trajectory with an MPC feedback controller. The formulation includes the use of a virtual leader to define the desired relative motion for each satellite using the HCW relative motion equations. A novel switching control strategy is used to determine when the MPC control should be activated as well as when a guidance trajectory should be calculated. Contributions beyond [23] include the use of the spherical polytopes implemented by [25] as well as the ability to ensure the satellites honor safety constraints when following the guidance trajectory. This work also performs an evaluation of different objectives for linear (L_1 and L_∞) and quadratic (L_2) programming solutions in terms of fuel usage and computation requirements. It is shown that the L_1 or L_∞ maybe be more applicable given a spacecraft with a high thrust, non-throttlable, on/off type propulsion system. The L_2 norm is more applicable for a smaller thrust, longer duration burning system, such as an electric propulsion system.

The remainder of the paper is outlined as follows. Section 2 provides background information about the relative dynamics and MPC. Section 3 presents a discussion of the switching strategy, the guidance trajectory formulation, and the MPC formulation for the L_2, L_1, and L_∞ norms. Section 4 presents simulation results. Concluding remarks are given in Section 5.

2 Background

This section presents background information of the relative dynamics used with the control algorithm and preliminary information needed for the development of the MPC.

2.1 Hill-Clohessy-Wiltshire Relative Dynamics

In this paper, a virtual leader is used to design a reference orbit for each satellite in the formation. There is no physical vehicle associated with the virtual leader; the virtual leader represents a fictitious satellite where motion is propagated through time according to a standard nonlinear motion model. Relative orbits can be generated from the virtual leader that define the nominal locations of each of the agents over time within the virtual structure. The relative orbits are propagated using the HCW equations [31–33].

The HCW dynamics are defined in the Local Vertical, Local Horizontal (LVLH) frame with the origin defined to be the location of the virtual leader. The x-axis of the LVLH frame is defined to be along the position vector of the virtual leader in the inertial frame, the z-axis is defined as the angular momentum vector of the virtual leader, and the y-axis is used to complete a right-handed coordinate system.

The relative state dynamics of each agent compared to the leader is defined as $\mathbf{x} = \begin{bmatrix} x \ y \ z \ \dot{x} \ \dot{y} \ \dot{z} \end{bmatrix}^T$ and the control as $\mathbf{u} = \begin{bmatrix} u_x \ u_y \ u_z \end{bmatrix}^T$, where the individual control components represent the thrust accelerations of the system. Using n_c to represent the mean motion of the reference orbit [33], and Δt to be a discretization step, the dynamics can be expressed in discrete time as

$$\mathbf{x}_{k+1} = \mathbf{A}_D \mathbf{x}_k + \mathbf{B}_D \mathbf{u}_k. \tag{1}$$

Using the exact discretization from [25], the resulting state and input matrices take the form

$$\mathbf{A}_D = \begin{bmatrix} \Phi_{rr} & \Phi_{rv} \\ \Phi_{vr} & \Phi_{vv} \end{bmatrix} \tag{2}$$

$$\Phi_{rr} = \begin{bmatrix} 4 - 3\cos n_c \Delta t & 0 & 0 \\ 6(\sin n_c \Delta t - n_c \Delta t) & 1 & 0 \\ 0 & 0 & \cos n_c \Delta t \end{bmatrix} \tag{3}$$

$$\Phi_{rv} = \begin{bmatrix} \frac{1}{n_c} \sin n_c \Delta t & \frac{2}{n}(1 - \cos n_c \Delta t) & 0 \\ \frac{2}{n}(\cos n_c \Delta t - 1) & \frac{1}{n}(4 \sin n_c \Delta t - 3 n_c \Delta t) & 0 \\ 0 & 0 & \frac{1}{n_c} \sin n_c \Delta t \end{bmatrix} \tag{4}$$

$$\Phi_{vr} = \begin{bmatrix} 3 n_c \sin n_c \Delta t & 0 & 0 \\ 6 n_c (\cos n_c \Delta t - 1) & 0 & 0 \\ 0 & 0 & -n_c \sin n_c \Delta t \end{bmatrix} \tag{5}$$

$$\Phi_{vv} = \begin{bmatrix} \cos n_c \Delta t & 2 \sin n_c \Delta t & 0 \\ -2 \sin n_c \Delta t & 4 \cos n_c \Delta t - 3 & 0 \\ 0 & 0 & \cos n_c \Delta t \end{bmatrix} \tag{6}$$

$$\mathbf{B}_D = \begin{bmatrix} \frac{2}{n_c^2}\sin^2\frac{n_c\Delta t}{2} & -\frac{2}{n_c^2}(\sin n_c\Delta t - n_c\Delta t) & 0 \\ \frac{2}{n_c^2}(\sin n_c\Delta t - n_c\Delta t) & -\frac{1}{2n_c^2}(8\cos n_c\Delta t + 3n_c^2\Delta t^2 - 8) & 0 \\ 0 & 0 & \frac{2}{n_c^2}\sin^2\frac{n_c\Delta t}{2} \\ \frac{1}{n_c}\sin n_c\Delta t & -\frac{1}{n_c}(2\cos n_c\Delta t - 2) & 0 \\ -\frac{4}{n_c}\sin^2\frac{n_c\Delta t}{2} & \frac{4}{n_c}\sin n_c\Delta t - 3\Delta t & 0 \\ 0 & 0 & \frac{1}{n_c}\sin n_c\Delta t \end{bmatrix}. \quad (7)$$

2.2 Model Predictive Control

This section presents a brief primer on MPC to present the basic idea and notation used throughout the sequel. A general overview of MPC is given, followed by a linear system representation and a discussion of the closed-loop stability. The interested reader is referred to [34] for a thorough development of MPC for linear systems.

While MPC generally loses any guarantee of optimality, it does provide the ability to express constraints, which is not common in many feedback control solutions. Furthermore, weights in the objective function can provide intuitive "control knobs" for tuning to the desired behavior.

2.2.1 Common Objective Functions for Linear System MPC

Common objective functions for linear systems include the L$_2$, L$_1$, and L$_\infty$ norms as they can result in quadratic and linear programs. Given an initial state of x_0, a time horizon of N intervals, and if we let $p = 1$, $p = 2$ or $p = \infty$ for the L$_1$, L$_2$, and L$_\infty$ norms, respectively, a general objective function can be written as

$$J(x,u)_p = || \; u_0 \; ||_{R,p} + \sum_{k=1}^{N-1} \Big[|| \; x_k \; ||_{Q,p} + || \; u_k \; ||_{R,p} \Big] + || \; x_N \; ||_{P,p} \qquad (8)$$

where R, Q, and P are the weightings on control usage, state error, and terminal error, respectively. Where previous work did not include portions of the cost function [17–21, 23], this approach allows the user to choose the relative weights. If the state error is more heavily weighted, the spacecraft trajectory is forced to stay close to the desired trajectory. This allows a greater degree of control beyond just staying with a given keep-in-volume.

This paper analyzes the L$_2$, L$_1$, and L$_\infty$ norms to evaluate the lowest fuel consuming approach and to evaluate the potential computational benefits in using one norm over the others. Details on representing the L$_2$ norm as a quadratic programming problem, and the L$_1$ and L$_\infty$ norms as linear programming problems can be found in [25].

A Note on MPC Stability

Like any feedback control technique, stability of the feedback can become a concern. Given Q and R, the terminal cost of the system can be designed to ensure asymptotic stability of the MPC problem [34]. The terminal cost is chosen to represent the cost-to-go of the unconstrained infinite horizon problem for the L_2 norm. The Discrete-time Algebraic Riccati Equation (DARE) can be used to find the cost-to-go, $X^T P_\infty X$. The DARE is represented as

$$0 = A^T P_\infty A - P_\infty + Q - A^T P_\infty B(B^T P_\infty B + R)^{-1} B^T P_\infty A, \tag{9}$$

where the positive definite solution for P_∞ is used.

The P_∞ found solving the DARE is then used as the matrix in the terminal cost, P, in the objective function, Eq. 8. This guarantees asymptotic stability of the system for the L_2 norm formulation [34]. Note that A and B must be completely controllable, R must be positive definite, and Q must be positive semi-definite and completely observable.

For the L_1 and L_∞ cases, if the exact discrete system is asymptotically stable, then X_f can be chosen as the positively invariant set of the system

$$x(k+1) = Ax(k) \quad \text{s.t.} \quad x \in X. \tag{10}$$

Therefore, the input 0 is feasible in X_f and the Lyapunov inequality for the L_∞ and L_1 cases is

$$-||\ Px\ ||_p + ||\ PAx\ ||_p + ||\ Qx\ ||_p \leq 0, \ \ \forall x \in X_f, \tag{11}$$

where $p = 1$ or $p = \infty$ for the L_1 and L_∞ norms, respectively. Eq. 11 is satisfied if a P is chosen that satisfies the Lyapunov function for the L_1 and L_∞ cases.

$$V = ||\ Px\ ||_p. \tag{12}$$

The P_p found solving Eq. 12 is then used in the objective function. This guarantees asymptotic stability of the system [34].

If such a X_f can be computed, it can be used as the terminal constraint. Eq. 13 is now satisfied by the infinite time unconstrained optimal cost matrix P_∞ in Eq. 14. Eq. 14 represents the optimal infinite horizon cost function.

$$\min_{u \in U, \ \ Ax+Bu \in X_f} -p(x) + q(x,u) + p(Ax+Bu) \leq 0, \ \ \forall x \in X_f \tag{13}$$

$$J^*_\infty(x(0)) = ||\ P_\infty x(0)\ ||_\infty. \tag{14}$$

3 Formation Flying Framework Development

The formation flying framework was developed with the objective of keeping an agent spacecraft within a spherical "keep-in volume", a time-varying volume within which the spacecraft must remain. The three main components used to maintain this constraint are a switching condition, a guidance law, and an MPC control law.

For the switching condition, the spacecraft's state is propagated forward in time and compared to a second time-varying volume known as the inner boundary. If any of these propagated states lie outside of the inner boundary then the spacecraft enters the guidance and control loop. The guidance law then calculates a fixed time, fuel optimal trajectory and control profile that is computed to guide the spacecraft from its current state to the nominal, desired orbit. This guidance trajectory is then given to the MPC control. The MPC follows the guidance trajectory, using feedback to reject any perturbations that may occur. Once the guidance trajectory is fully executed, the spacecraft returns to its drifting state and again begins comparing its propagated states to the inner boundary.

The full framework is outlined in Algorithm 1 and the remainder of this section presents the details of the switching strategy, guidance law, and MPC control law.

Algorithm 1 Guidance and control structure

```
$\hat{\mathbf{x}}_0 \leftarrow \mathbf{x}$                                   (Initialize From Current State)
$\hat{\mathbf{x}}_{k+1} = A_D \hat{\mathbf{x}}_k \quad k = 0, \dots, N_{drift} - 1$          (Propagate Drift Trajectory)
if not control_on and $\exists \, || \hat{\mathbf{x}}_k - \mathbf{x}_{d,k} ||_2 \geq r_{inner}, k = 0, \dots, N_{drift}$ then
    $control\_on \leftarrow true$
    $\min_{\mathbf{x}_g, \mathbf{u}_{nom}} \frac{1}{2} \sum_{k=0}^{N_g - 1} || \mathbf{u}_{nom,k} ||_{R,p} + \frac{1}{2} || \mathbf{x}_{e,N_g} ||_{P,p}$     (Guidance Trajectory)
    $mpc\_step \leftarrow 1$
if $control\_on$ then
    if $mpc\_step > N_g$ then
        $control\_on \leftarrow false$
    else
        $\mathbf{P}_k = \mathbf{P}_{nom} + \mathbf{x}_{d,k} \quad k = 0, \dots, N_{drift}$          (Translate Sphere Points)
        $\mathbf{L}_k \leftarrow convhull(\mathbf{P}_k) \quad k = 0, \dots, N_{drift}$       (Generate Polytope Faces)
        $\mathbf{A}_{poly,k}, \mathbf{b}_{poly,k} \leftarrow face2mat(\mathbf{L}_k), k = 0, \dots, N_{drift-1}$
                                                (Formulate Constraints as Matrices)
        $\min_{\mathbf{x}_k, \mathbf{u}_k} \frac{1}{2} || \mathbf{u}_0 ||_{R,p} + \frac{1}{2} \sum_{k=1}^{N-1} \left( || \mathbf{x}_{e,k} ||_{Q,p} + || \mathbf{u}_k ||_{R,p} \right)$
            $+ \frac{1}{2} || \mathbf{x}_{e,N} ||_{P,p}$                              (MPC Control Law)
        $\mathbf{u}_{ECI} \leftarrow lvlh2eci(\mathbf{u}_0)$    (Convert Control from LVLH to ECI Frame)
        $mpc\_step = mpc\_step + 1$
else
    $\mathbf{u}_{ECI} = \mathbf{0}_{1 \times 3}$
```

3.1 Switching Strategy

Since it is unreasonable, due to mission objectives or operating constraints, for the spacecraft to be constantly maneuvering, a switching strategy is developed to define the periods when maneuvering is most effective. Central to this strategy are two boundary constraints that define a volume of space centered on the desired orbit of the spacecraft at each moment in time. The outer boundary is the keep-in boundary and defines the volume of space the spacecraft is not allowed to leave. The inner boundary is used to trigger the beginning of a correction maneuver. A corrective maneuver begins when the spacecraft is predicted to cross the inner boundary.

To determine if the guidance and control loop should begin, the desired ($\mathbf{x}_{d,k}$) and drift ($\hat{\mathbf{x}}_k$) trajectories are first updated over N_{drift} steps of the drift horizon using the linearized HCW dynamics, A_D. With the current state, $\mathbf{x}$, used as the initial state for the drift trajectory, $\hat{\mathbf{x}}_0$, and the current desired state being used as the initial desired state, the full trajectories are found as

$$\hat{\mathbf{x}}_{k+1} = A_D \hat{\mathbf{x}}_k \quad k = 0, \ldots, N_{drift} - 1 \tag{15}$$

$$\mathbf{x}_{d,k+1} = A_D \mathbf{x}_{d,k} \quad k = 0, \ldots, N_{drift} - 1. \tag{16}$$

Each pair of desired and drift states are then compared to determine if the spacecraft is predicted to cross the inner boundary at any point over the drift horizon. The boundary is defined as a sphere with radius r_{inner}. Crossing the inner boundary is detected by comparing the Euclidean distance between the desired and drift trajectories as

$$|| \hat{\mathbf{x}}_k - \mathbf{x}_{d,k} ||_2 \geq r_{inner} \quad k = 0, \ldots, N_{drift}. \tag{17}$$

If the drift trajectory is found to exceed the allowable bounds, then the guidance and control law begins and continues until the guidance planned trajectory is fully executed. Once fully executed, the MPC controller is turned off and the evaluation of the drift trajectory is reinitiated.

3.2 The Guidance Trajectory Generation

The purpose of the guidance law is to compute a fixed time, fuel optimal trajectory that guides the spacecraft from its current state back to the desired orbit. Given the guidance time horizon, N_g, control cost, R_g, and terminal cost, P_g, bounds on control inputs, $\mathbf{u}_{max}$ and $\mathbf{u}_{min}$, and bounds on states $\mathbf{x}_{g,max}$ and $\mathbf{x}_{g,min}$, the guidance trajectory $\mathbf{x}_{g,k}$ and nominal guidance control $\mathbf{u}_{g,k}$ can be calculated with the p norm objective function. Additionally, an error state, $\mathbf{x}_{e,k}$ is introduced to allow the tracking of a non-zero desired trajectory, $\mathbf{x}_{d,k}$. From these, the trajectory is determined based on the following optimization problem.

$$\begin{aligned}
\min_{\mathbf{x}_{g,k}, u_{g,k}} \quad & \frac{1}{2}\sum_{k=0}^{N_g-1} \parallel \mathbf{u}_{g,k} \parallel_{R_g,p} + \frac{1}{2} \parallel \mathbf{x}_{e,N_g} \parallel_{P_g,p} \\
\text{s.t.} \quad & \mathbf{x}_{g,k+1} = \mathbf{A}_D\mathbf{x}_{g,k} + \mathbf{B}_D\mathbf{u}_{g,k} \quad k = 0, 1, \ldots, N_g - 1 \\
& \mathbf{x}_{g,0} = \mathbf{x}(0) \\
& \mathbf{x}_{e,k} \leq \mathbf{x}_{g,max} \\
& \mathbf{x}_{e,k} \geq \mathbf{x}_{g,min} \\
& \mathbf{x}_{e,k} = \mathbf{x}_{g,k} - \mathbf{x}_{d,k} \quad k = 1, 2, \ldots, N_g \\
& \mathbf{u}_{g,k} \leq \mathbf{u}_{max} \quad k = 0, 1, 2, \ldots, N_g - 1 \\
& \mathbf{u}_{g,k} \geq \mathbf{u}_{min} \quad k = 0, 1, 2, \ldots, N_g - 1
\end{aligned} \quad . \tag{18}$$

The initial state $\mathbf{x}_0$ is defined as the spacecraft's state when the guidance law is initialized. The desired orbit states, $\mathbf{x}_{d,k}$, are propagated and found at each point along the time horizon and used as the desired states. The control limits, $\mathbf{u}_{max}$, are set based on the maximum thrust available along each axis and the current mass of the spacecraft.

One consequence of orbital dynamics is that there are points along an orbit where a given maneuver is the most fuel efficient, for example, perigee, apogee, and the intersection of two orbital planes. This has two major implications in the generation of the guidance trajectory. First, a fuel optimal trajectory can experience prolonged periods with significant state errors while it waits for the optimal point to maneuver. This means that if the guidance law were incentivized to minimize the transient state error via the Q weights, it would maneuver too early, expending unnecessary control usage. Second, the time horizon over which the guidance law optimizes must be sufficiently long to encompass the fuel optimal points.

Since the guidance law optimizes over a longer time horizon, the full outer polytope constraint can create a significant computation burden. To alleviate this burden, a simpler cube volume is used to approximate the constraint. This cube is inscribed within the outer spherical keep-in volume and allows the constraint to be expressed as a pair of simpler state constraints,

$$\mathbf{x}_{e,k} \leq \mathbf{x}_{max} \tag{19}$$

$$\mathbf{x}_{e,k} \geq \mathbf{x}_{min}. \tag{20}$$

Simply solving the fuel optimal maneuver problem without constraints does not guarantee that the spacecraft stays withing the desired region, e.g., [35, 36], but with this approach any solution found by the guidance law is guaranteed to remain within the outer keep-in volume since the simplified constraint is a conservative approximation. Once a solution is found, the trajectory $\mathbf{x}_g$ and control profile $\mathbf{u}_g$ are stored to be used by the control law.

3.3 MPC Control Law

The MPC control law is tasked with following the trajectory found by the guidance law and rejecting any perturbations along the guidance trajectory. The MPC is built on the optimization formulation shown below in Eq. 26, which is similar to the guidance law. However, while the guidance law is only run a single time when the spacecraft is predicted to leave the keep-in volume, the MPC is repeatedly run over consecutive simulation steps in standard MPC fashion.

Since an optimal trajectory has already been found by the guidance law, the MPC is able to calculate deviations from the guidance trajectory over a shorter time horizon, N_c. From the stored trajectory and control profile, the MPC selects the period that corresponds to the current interval it is optimizing over. Given the current step of the guidance trajectory that the MPC control law is on, $n_c \in \{1, \dots, N_g\}$,

$$\mathbf{x}_{gd,\ell} = \mathbf{x}_{g,n} \quad \ell = 1, \dots, N_c;\ n = n_c + \ell, \tag{21}$$

$$\mathbf{u}_{nom,\ell} = \mathbf{u}_{g,n} \quad \ell = 0, \dots, N_c - 1;\ n = n_c + \ell. \tag{22}$$

The trajectory is used as the desired states, $\mathbf{x}_{gd,\ell}$, while the associated control profile is used as the nominal control inputs, $\mathbf{u}_{nom,\ell}$. For $n > N_g$,

$$\mathbf{x}_{gd,\ell} = \mathbf{A}_D^{n-N_g} \mathbf{x}_{d,N_g} \quad \ell = 1, \dots, N_c;\ n = n_c + \ell, \tag{23}$$

$$\mathbf{u}_{nom,\ell} = \mathbf{0}_{3\times 1} \quad \ell = 0, \dots, N_c - 1;\ n = n_c + \ell, \tag{24}$$

where the desired states and nominal controls correspond to the nominal, desired uncontrolled orbit starting at the end of the guidance trajectory.

While the outer boundary is nominally a sphere, a convex polytope is used as an approximation since that formulation can easily be integrated into the MPC formulation as a linear constraint. This work leverages the polytope formulated by Smith et al, [25], as the boundary constraints. Given the outer boundary's radius, r_{outer}, a number of points, N_{bound}, are selected on the surface of a sphere which is centered on the origin with a radius of r_{outer}. At each point along the desired trajectory over the time horizon of the MPC, the samples are translated so that they are centered on the nominal desired trajectory

$$\mathbf{P}_{\ell,j} = \mathbf{P}_{nom,j} + \mathbf{x}_{d,\ell},\ \ell = 0, \dots, N_c, j = 1, \dots, N_{bound}. \tag{25}$$

For each step ℓ, the polytope approximation is constructed by finding the convex hull of the points in $\mathbf{P}_\ell$ to get a set of faces $\mathbf{L}_\ell$. Using the normal and a point on each face, the polytope constraint can be formulated as $\mathbf{A}_{poly,\ell}\mathbf{x}_\ell \leq \mathbf{b}_{poly,\ell}$. A summary of the formulation of these constraints can be found in Appendix A.1.

The full polytope constraints, $\mathbf{A}_{poly,\ell}$, $\mathbf{b}_{poly,\ell}$, are calculated at each step within the time horizon using the propagated desired orbit and the outer polytope keep-in volume. Note that while the desired trajectory, $\mathbf{x}_{d,\ell}$, comes from the guidance solution, the polytope is centered on the nominal desired orbit. Although the guidance trajectory is designed to stay within a conservative approximation of the keep-in volume, perturbing forces, or model mismatch can push the spacecraft off that trajectory. The full polytope constraint is included here to assure that even if the spacecraft drifts, it will not violate the boundary constraints.

The MPC then optimizes over its time horizon and at each step along the guidance trajectory, a new solution is found as

$$\begin{aligned}
\min_{x,u} \quad & \frac{1}{2} \mid\mid \mathbf{u}_0 \mid\mid_{R,p} + \frac{1}{2} \sum_{\ell=1}^{N_c-1} \Big[\mid\mid \mathbf{x}_{e,\ell} \mid\mid_{Q,p} + \mid\mid \mathbf{u}_\ell \mid\mid_{R,p} \Big] \\
& \qquad\qquad\qquad\qquad + \frac{1}{2} \mid\mid \mathbf{x}_{e,N_c} \mid\mid_{P,p} \\
\text{s.t.} \quad & \mathbf{x}_{\ell+1} = \mathbf{A}_D \mathbf{x}_\ell + \mathbf{B}_D \mathbf{u}_\ell + \mathbf{B}_D \mathbf{u}_{nom,\ell} \quad \ell = 0, 1, \ldots, N_c - 1 \\
& \mathbf{x}_0 = \mathbf{x}(0) \\
& \mathbf{A}_{poly,\ell} \mathbf{x}_\ell \leq \mathbf{b}_{poly,\ell} \quad \ell = 1, 2, \ldots, N_c \\
& \mathbf{x}_{e,\ell} = \mathbf{x}_\ell - \mathbf{x}_{gd,\ell} \quad \ell = 1, 2, \ldots, N_c \\
& \mathbf{u}_\ell \leq \mathbf{u}_{max} - \mathbf{u}_{nom,\ell} \quad \ell = 0, 1, 2, \ldots, N_c - 1 \\
& \mathbf{u}_\ell \geq -\mathbf{u}_{max} - \mathbf{u}_{nom,\ell} \quad \ell = 0, 1, 2, \ldots, N_c - 1
\end{aligned} \quad . \tag{26}$$

Once the control law finds the solution of controls, $\mathbf{u}_\ell$, that solve the optimization problem, the actual applied control input is determined as

$$\mathbf{u}_{apply,\ell} = \mathbf{u}_\ell + \mathbf{u}_{nom,\ell}. \tag{27}$$

The first control in the sequence, $\mathbf{u}_{apply,0}$, is applied to the system, the new state is then measured, the control law is re-initialized, and the next optimal control is found. The control law runs until the full guidance trajectory has been executed.

4 Numerical Simulation

This section discusses the implementation of the switching condition, guidance law, and control law discussed in the previous sections. The simulation used to evaluate the framework is presented, followed by an assessment of the control trajectories, comparison of computation time of the different norms, discussion of the control usage, and a comparison of the ΔV usages.

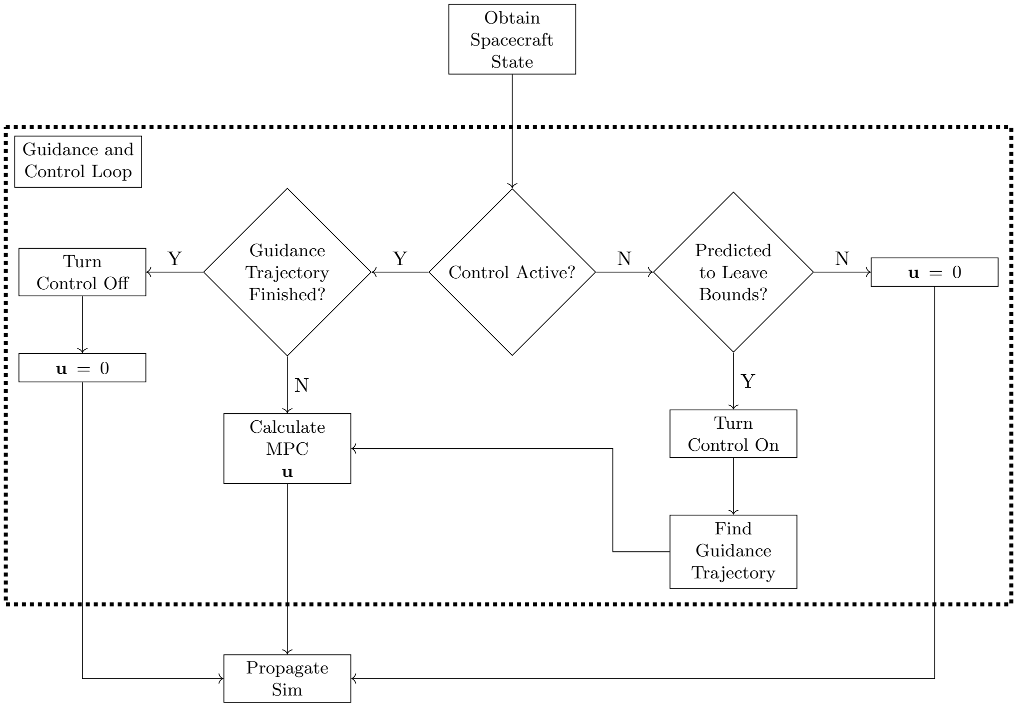


Fig. 1: Flowchart showing the simulation logic structuring and flow for a single spacecraft. The dashed line indicates the guidance and control loop that each spacecraft in the formation runs in parallel.

4.1 Simulation Overview

A flowchart of the framework used to test the guidance and control laws is depicted, for a single spacecraft, in Fig. 1.

The virtual leader or reference orbit's initial state is defined using Keplerian orbital elements and then converted to a Cartesian Earth Centered Inertial (ECI) state. The relative states and desired relative location of the agents spacecraft in the formation are initialized using D'Amico's ROEs [37]. The ROEs are first converted to inertial orbital elements, using the leader as a reference, and then to both the inertial ECI coordinates [38] and relative LVLH coordinates. The advantage to using ROEs to define the initial states of the agent is that they offer a more intuitive understanding of the relative motion when compared to Cartesian coordinates.

Before entering the guidance and control loop, each spacecraft receives its true LVLH states. While measurement noise and navigation uncertainties would exist in an actual system, these are not included for these simulations and the spacecraft all have perfect knowledge of their states.

Using the state, the guidance and control laws calculate the desired acceleration (Section 3). Since the control is calculated in the LVLH frame, it is first converted to the ECI frame to match with the dynamics model.

The desired control acceleration is converted to a force using the current mass of the spacecraft, bounded to ensure it is within the limits of the available thrust, and then applied to the dynamics. From the acceleration, the mass flow rate is calculated as

$$\dot{m} = -\frac{||\mathbf{u}||_1 m}{g_0 I_{sp}} \tag{28}$$

where $\mathbf{u} \in \mathbf{R}^3$ is the control acceleration, m is the current mass, g_0 is standard gravity, and I_{sp} is the specific impulse. Before planning any trajectories or controls, the spacecraft's current mass is used with the maximum allowable thrust to determine the upper bound on acceleration. Thus, while the guidance and control laws assume an unchanging mass over each iteration of the planning cycle, any changes in mass due to previous control usage are accounted for in the model dynamics. Since the decreasing fuel mass means the spacecraft will experience a higher acceleration for a given thrust, this provides a conservative bound on the maximum allowable control.

The true states for the leader and agents are propagated using the nonlinear dynamics. In the following equations, F_d is the force due to drag, F_{J_2} is the force due to the J_2 perturbation [33], and $\mathbf{u}$ is the calculated control acceleration, applied as a zero-order hold. The ECI state of the spacecraft is given as $\mathbf{r}$ with the individual portions of the state indicated by the subscripts p, for position, x, y, z, for individual position axes, and v for velocity.

$$F_d = -\frac{1}{2}\rho C_d A|\mathbf{v}_{rel}|\mathbf{v}_{rel} \tag{29}$$

$$F_{J_2} = \frac{3}{2}\frac{J_2 \mu R_E^2}{\mathbf{r}_p{}^4}\left[\frac{r_x}{\mathbf{r}_p}\left(5\frac{r_z^2}{\mathbf{r}_p{}^2} - 1\right)\hat{i} + \frac{r_y}{\mathbf{r}_p}\left(5\frac{r_z^2}{\mathbf{r}_p{}^2} - 1\right)\hat{j} + \frac{r_z}{\mathbf{r}_p}\left(5\frac{r_z^2}{\mathbf{r}_p{}^2} - 3\right)\hat{k}\right] \tag{30}$$

$$\dot{\mathbf{r}}_p = \mathbf{r}_v \tag{31}$$

$$\dot{\mathbf{r}}_v = -\mu\frac{\mathbf{r}_p}{\mathbf{r}_p{}^3} + \frac{F_{J_2}}{m} + \frac{F_d}{m} + \mathbf{u} \tag{32}$$

The drag force is a simplified model where the air density, ρ, is determined by interpolating the altitude against selected values in the US Standard Atmosphere 1976 model [39].

The relative velocity of the spacecraft to the Earth's atmosphere, $\mathbf{v}_{rel}$ is calculated as[33]

$$\mathbf{v}_{rel} = \mathbf{r}_v - \boldsymbol{\omega}_E \times \mathbf{r}_p. \tag{33}$$

with the Earth's rotation, $\boldsymbol{\omega}_E$, assumed to be entirely in the $\hat{k}$ direction. The spacecraft coefficient of drag, C_d, and spacecraft area, A, are held constant during the simulation. While the simulations all assume a near-circular orbit, a similar approach could be applied to elliptical orbits by replacing the relative motion HCW equations with the Yamanaka-Ankerson relative dynamic model [40].

4.1.1 Simulation Parameters

A scenario based on a standard 12U CubeSat design built by the Space Dynamics Laboratory for formation flying missions was used to evaluate the architecture developed in Section 3. The simulation was performed on an Intel Xeon W-1250P 4.10 GHz processor with 32 GB of RAM. The virtual leader reference orbit is a near circular Low Earth Orbit (LEO) with the parameters shown in Table 1. The physical characteristics and the simulation parameters used in this work are shown in Table 2. The propulsion system modeled in this work is based on the hybrid system described in [41]. It is assumed that the spacecraft propulsion system is able to provide a thrust of up to 0.5 N along any three orthogonal axes and that the thrust is throttleable up to the maximum thrust of 0.5 N.

Table 1: Reference Orbit of the virtual leader

Keplerian Orbital Element	Value
Semi-major Axis	6878 km
Eccentricity	1×10^{-4}
Inclination	25°
RAAN	45°
Argument of Perigee	0°
Initial True Anomaly	100°

Table 2: Parameters for simulation

Parameter	Value
Spacecraft Wet Mass	24 kg
Spacecraft C_D	2.2
Spacecraft Area to Mass	0.002 m^2/kg
Spacecraft Thrust	0.5 N
Spacecraft ISP	160 s
Simulation Timestep	10 s
Total Simulation Timesteps	60 480 (7 days)
Guidance Horizon	820 Steps (2.28 hrs)
MPC Horizon	10 Steps (100 s)
Drift Horizon	600 Steps (1.67 hrs)
Inner Sphere Radius	289 m
Outer Sphere Radius	500 m
Guidance Cube Edge	577 m

As shown in Table 2, the simulation uses a fixed time step of 10 seconds. The size of the time step constrains the time that the control must be on. The duration of this time step could be adjusted to offer the spacecraft operator a more refined maneuver if desired.

The drift horizon is set to 600 steps or approximately 1.67 hrs, which is longer than a single orbital period. The guidance horizon is longer still at 820 steps (2.28 hrs). The length of the guidance horizon is set to be both longer than one orbital period and longer than the drift horizon. Setting these lengths to be greater than one orbit allows for the spacecraft to detect when it may leave the bounds and still have time to wait for the optimal moment to begin maneuvering, which in many cases is at apogee, perigee, or a node crossing. However, the exact length of the drift and guidance horizons presented here are otherwise arbitrarily chosen. The controller horizon is much shorter, 10 steps (100 sec), since its purpose is to reject short term disturbances. A longer time horizon could be chosen for the controller which may improve performance at the cost of additional computation time.

The outer polytope was set as a sphere with radius 500 m with the inner, trigger sphere given a radius 289 m. For the guidance loop, the simplified keep-in-boundary was modeled as a cube with each edge length being 577 m long. The size of the cube for the guidance loop is determined by the maximum size that will fit within the outer polytope, and the size of the inner sphere is the maximum sphere size that will fit within the cube, as depicted in Fig. 2.

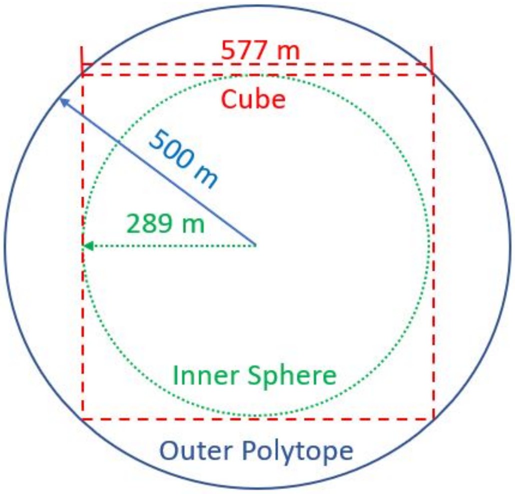


Fig. 2: Boundary sizes, outer polytope (blue circle), inner sphere (green circle), cube (red square).

With this geometry, the spacecraft will always be within the cube when it triggers the start of the guidance and control portion by predicting it will cross the inner sphere. If the inner sphere were larger than the cube used for the guidance boundary, then situations could arise where the spacecraft attempts to develop a guidance trajectory while being outside the allowable volume. Similarly, if the cube is larger than the outer polytope, a guidance trajectory could be planned which violates the outer keep-in constraint. While smaller boundaries could be used, this geometry maximizes the volume the guidance and control laws can use, relative to the overall keep-in volume, while maintaining consistent keep-in constraints across the full control architecture.

For this paper, a three spacecraft formation is used to illustrate the approach detailed above. Since no communication or tracking of other vehicles is required between the spacecraft, a formation can be composed of as many individual spacecraft as desired without increasing the complexity of the approach. The main limitation of the approach is that all the spacecraft in the formation must reference a single, common orbit,that the HCW equations can approximate well. Additionally, each spacecraft orbit must be defined such that none of the boundary volumes overlap.

The three spacecraft in the formation have desired orbits defined by Relative Orbital Elements (Table 3). The desired orbit for each spacecraft is identical with regards to the shape and orientation of the orbit but they are spread out along the reference orbit. Thus, the first spacecraft has its relative orbit centered on the reference orbit with the second spacecraft ahead of it and the third spacecraft behind it with the along track separation between each of the spacecraft being approximately 6 km. Each spacecraft is in a non-drifting natural motion circumnavigation (NMC) orbit with the maximum radial and normal components creating a 1x2 km ellipse, as shown in Fig. 3. The algorithm maintains this non-drifting NMC by keeping the agent spacecraft on the desired trajectory. For this work, the relative orbits and keep-in volumes have been defined such that the keep-in volumes of any two spacecraft never intersect. This ensures that two spacecraft can never exist in the same space and the formation is collision free.

Table 3: Relative Orbital Elements for the desired agent trajectory

ROE Element	Spacecraft 1 (unitless)	Spacecraft 2 (unitless)	Spacecraft 3 (unitless)
δa	0	0	0
$\delta \lambda$	0	8.723×10^{-4}	-8.723×10^{-3}
δe_x	1.454×10^{-4}	1.454×10^{-4}	1.454×10^{-4}
δe_y	0	0	0
δi_x	1.454×10^{-4}	1.454×10^{-4}	1.454×10^{-4}
δi_y	0	0	0

4.1.2 Controller Tuning

The controller calculates the control necessary to reject perturbations and stay on the guidance trajectory. Thus, increasing the transient error weights, relative to the control weights, will cause the controller to overcome more perturbations and stay closer to the guidance trajectory at the cost of additional fuel usage. In contrast, if the controller state error weights are reduced to zero, the controller will only execute the control inputs that were initially calculated for the guidance trajectory. Effectively, this causes the full guidance and control law to run in an open-loop configuration whereas any non-zero state error weightings close the loop.

The behavior of the spacecraft was observed through simulations and the weights adjusted to minimize ΔV usage, maximize time between needed

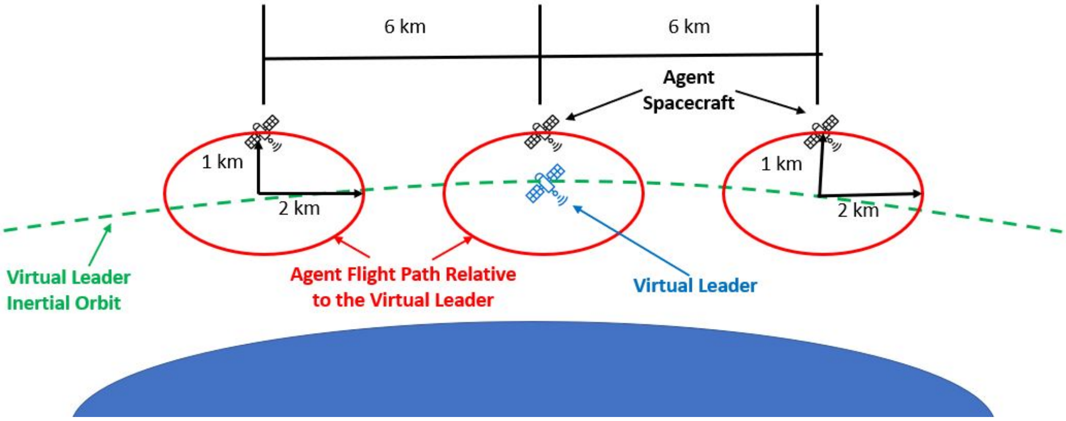

6 km
6 km
Agent Spacecraft
1 km
2 km
1 km
2 km
Virtual Leader Inertial Orbit
Agent Flight Path Relative to the Virtual Leader
Virtual Leader

Fig. 3: Configuration of agent spacecraft used for simulation.

maneuvers, and minimize deviation from the guidance trajectory. The controller weights used as part of this work are seen in Table 4. It should be noted that these values are not claimed to be optimal in any sense and merely represent an acceptable blend of ΔV usage and spacecraft state error.

Although the weights are similar between the three norms, the different norm definitions mean that the impact is not identical. For example, all of the components of the L_1 norm contribute linearly to the cost, the L_2 norm squares the vector under consideration, and the L_∞ norm only accounts for the largest scaled value. This means that even if identical weights are used, the resulting behavior can be expected to be different between the norm definitions and, similarly, one cannot define a common baseline between the norm definitions. The weights given were chosen to try and have similar control characteristics across each norm type in terms of how aggressive the perturbation rejection is. Ultimately though, while the descriptive behavior of the controllers is unlikely to change as a result of the weights, some variation is to be expected based on how the baseline between the norms is determined.

4.2 Computation Time

An additional simulation was set up to evaluate the computation time for the controller for each of the three norms as the time horizon of the MPC increased. For each norm definition, a series of trials with the time horizon ranging from 1 to 150 steps was performed. In each trial, a single spacecraft was initialized and run for a total simulation duration of 1,200 seconds. This time frame was selected to allow for a broader range of horizons to be tested in a reasonable amount of time. Data was only collected when the MPC was active and does not include any computation times of 0 s when the controller was not used. The solutions for all three of the norms were found using the Gurobi software [42].

The mean computation time for each norm over the various planning horizons can be seen in Fig. 4. For short time horizons, the computation time is approximately identical for all three norms. As the MPC optimizes over longer time horizons, the necessary time increases, as expected, and differences

Table 4: Controller weights for the various norm types

Norm	Transient Error	Control Usage
L_1	$\begin{bmatrix} 10.0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10.0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10.0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.1 \end{bmatrix}$	$\begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & 0.01 \end{bmatrix}$
L_2	$\begin{bmatrix} 10.0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10.0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10.0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.0 \end{bmatrix}$	$\begin{bmatrix} 0.0001 & 0 & 0 \\ 0 & 0.0001 & 0 \\ 0 & 0 & 0.0001 \end{bmatrix}$
L_∞	$\begin{bmatrix} 10.0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10.0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10.0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.01 \end{bmatrix}$	$\begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & 0.01 \end{bmatrix}$

between the norms become clearer. The time required for the L_1 norm increases faster than the other two that remain relatively comparable until a horizon of about 125 steps. A summary of the results of just the MPC computation time is shown in Table 5 where the minimum and maximum average times and standard deviations are shown. From the simulation results (presented in full in Section 4.6) the computation time of the full guidance and control law can be found in Table 6. This is the time needed to calculate the control over a time horizon of 10 steps as well as the time needed to compute the guidance trajectory over the longer 820 step horizon. As can be seen, the computation time requirements for the L_1 norm and L_∞ norms are comparable while the L_2 norm is significantly faster which is likely due to differences between the optimization solving techniques for linear programs (L_1 and L_∞) and quadratic programs (L_2) in Gurobi [42]. However, in all cases, the maximum computation time required was less than the 10 s simulation timestep. While no special effort was taken to mimic flight hardware conditions or optimize for computation time these results do suggest that running the full guidance and control law in real time is a feasible approach for all norms.

Table 5: MPC Computation time comparisons

Norm	1 Step Horizon	75 Step Horizon	150 Step Horizon
L_1	0.013 ± 0.203 s	0.494 ± 0.191 s	1.589 ± 0.232 s
L_2	0.006 ± 0.04 s	0.39 ± 0.099 s	1.028 ± 0.168 s
L_∞	0.011 ± 0.154 s	0.367 ± 0.147 s	1.248 ± 0.196 s

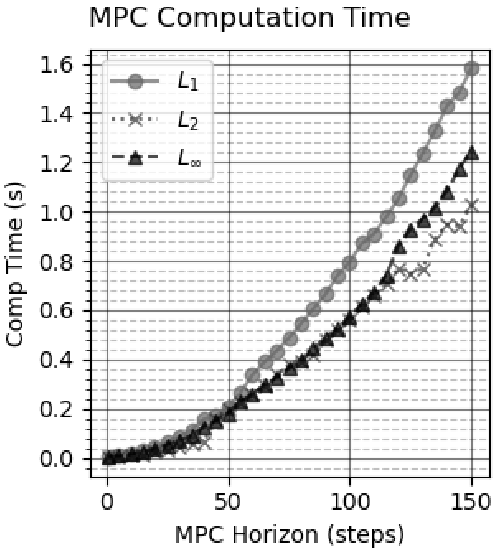


Fig. 4: Computation time required for planning a single simulation step for the various norm definitions and receding horizons.

Table 6: Computation times for the MPC with guidance

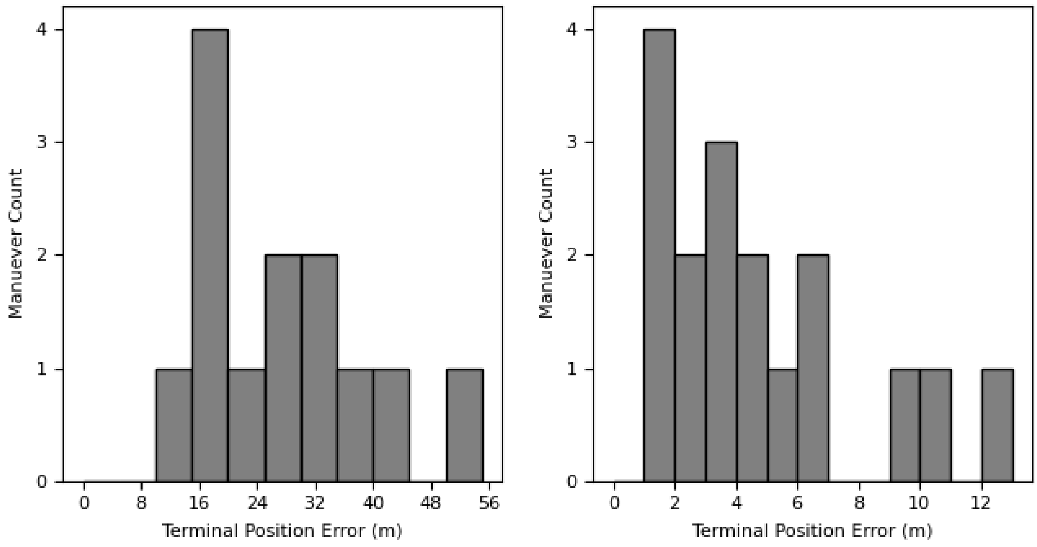
Norm	Mean Computation Time	Max Computation Time
L_1	3.123 ± 0.162 s	4.848 s
L_2	0.705 ± 0.047 s	1.122 s
L_∞	2.236 ± 0.206 s	7.185 s

4.3 Control Approach Comparisons

Other control approaches can be used to maintain a spacecraft formation. This section briefly explores the other approaches and compares them to the control architecture described above.

4.3.1 Open vs Closed Loop Control

Since the MPC is designed to reject small perturbations it is, in a sense, optional and the guidance trajectory can be used as an open-loop control. An example can be seen in Fig. 5 where Fig. 5a shows the histogram of position error at the end of each maneuver for the open-loop, guidance only control and Fig. 5b shows the histogram for the closed-loop control using MPC. Note that the x-axis is scaled differently between the two plots due to the disparity in terminal maneuver error in the two methods. As expected, the closed-loop control is able to drive the final position error much lower at the end of the guidance law. On average, the open-loop control has 27.28 m of error at the end of the guidance trajectory while the closed-loop control has 4.79 m of error at the end of the guidance trajectory. However, this additional accuracy comes at the cost of ΔV usage; the open-loop control only used 2.59 m/s of ΔV over the week-long simulation while the closed loop control used 4.34 m/s over an identical period. This balance can be adjusted by the weightings in the MPC where additional weight on the control cost will lean towards open-loop performance and additional weights on the error will lean towards higher rejection of state errors.



(a)

(b)

Fig. 5: Histogram of position error at the end of each maneuver for the open loop (5a) and the closed loop MPC control (5b).

4.3.2 MPC vs LQR Control

Another possibility is to use a linear-quadratic regulator (LQR) feedback control law in place of the MPC. For this analysis the LQR feedback control was given the guidance trajectory, which was planned using the L_1 norm, and attempted to minimize the error between the spacecraft's current state and the state planned by the guidance trajectory. The ΔV results can be seen in Fig. 6 and, superficially, look similar to the MPC results, where large spikes in control usage dominate with smaller control inputs in-between. However, the ΔV usage is significantly higher. For the LQR case, a total of $25.95\,\mathrm{m/s}$ of ΔV was used for the full week-long simulation. This is three times larger than any of the MPC simulations, which had a max total ΔV usage of $8.55\,\mathrm{m/s}$ when using the L_∞ norm (Section 4.6).

4.3.3 Continuous Control

The final comparison performed as part of this work, is to have the control always active with no drift periods. For this scenario the guidance and control loop would reinitialize immediately after the previous loop finished so

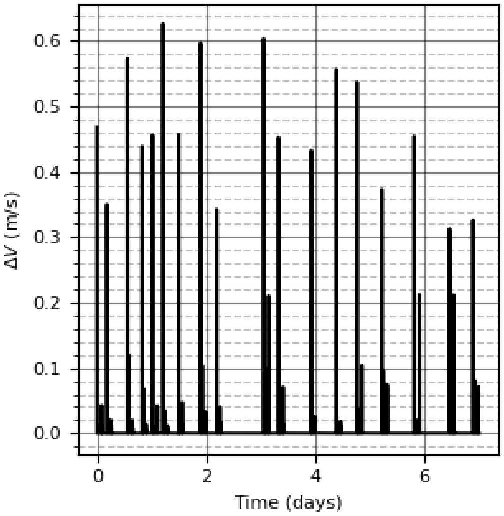


Fig. 6: ΔV usage for the guidance and control law using an LQR controller in place of the MPC.

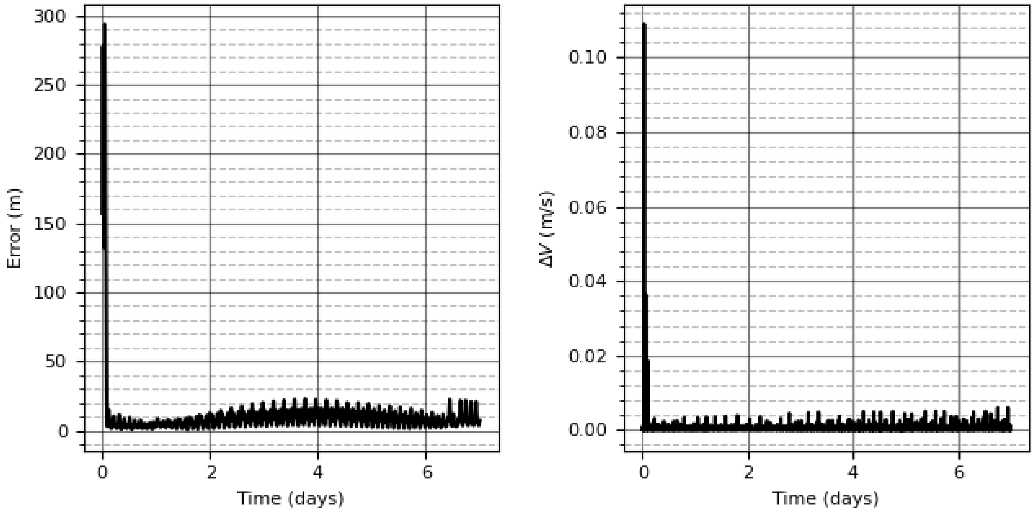
that there were no drifting periods. The results are shown in Fig. 7 where 7a shows the position error over time and 7b shows the ΔV usage over time. For the position errors, after the initial error is corrected, the position error never increases above 25 m. Additionally, since only minor corrections are needed to correct these small position errors, the instantaneous ΔV usage is approximately an order of magnitude smaller than what is seen when the spacecraft is allowed to drift before correcting errors.

However, this approach is undesirable for a number of reasons. First, mission constraints and thruster parameters generally do not allow for continual thrusting. Often power constraints, thermal constraints, or pointing requirements do not allow for the propulsion system to be on continuously. Second, thrusters generally do not have the ability to infinitely throttle and are restricted to some minimum thrust level that could make the control profile shown in Fig. 7b infeasible. Third, although the instantaneous ΔV is low, the total required ΔV is 7.90 m/s which is roughly double the 4.34 m/s used when the spacecraft was allowed to drift with the closed loop control (Section 4.3.1). Thus, this approach is impractical for use on-orbit.

4.4 Simulation Trajectory Evaluation

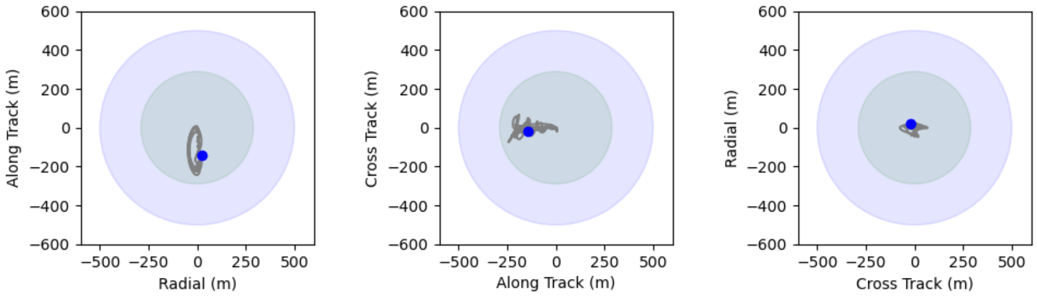
Fig. 8 shows an example of the trajectory produced at a simulation time of one day, using the switching strategies with an outer radius of 500 m and the L_1 norm.

The figure is shown in an LVLH frame centered on the desired trajectory so the displayed line indicates the perturbed trajectory of the spacecraft, relative to the desired. The initial state is indicated by the blue dot with the outer polytope shown in purple and the sphere shown in green.



(a) (b)

Fig. 7: Position error with the MPC constantly on (7a) and ΔV usage (7b)



(a) (b) (c)

Fig. 8: Trajectory for the L_1 norm at a simulation time of one day, purple circle represents the outer polytope, green circle represents the sphere, the blue circle is the spacecraft, and the grey line represents the spacecraft trajectory. 8a shows the along track vs. radial direction, 8b shows the cross track vs. along track, and 8c shows the radial vs. cross track orientation.

In Fig. 8, the spacecraft is initialized near the boundary of the sphere and immediately begins maneuvering until it fully executes its guidance plan. Once the control turns off, the spacecraft begins to drift. This is primarily seen in the along-track direction but the cross track axis accumulates noticeable error as well. Once the error accumulates to the point that the spacecraft is predicted to leave the sphere, the control activates again and the process repeats.

As an additional point of comparison, the distance between the 1st and 2nd and 2nd and 3rd spacecraft are shown in Fig. 9 and 10. Fig. 9 shows a simulation that uses the guidance and control algorithm as described in this paper. As seen in Fig. 9, the separation distances between the spacecraft are kept near the desired 6 km. In contrast, Fig. 10 shows the separation between the three spacecraft if they were not being controlled. The separation distance between vehicles one and two is unbounded in its growth and the two spacecraft are shown to drift apart. In contrast, vehicles 1 and 3 drift towards each other with a minimum separation distance of less than 15 m, at 4.7 days into the simulation, after which they pass each other and start drifting apart. This is a dangerously close approach and represents a high probability of collision for an actual system when accounting for measurement uncertainty in the spacecraft's positions. Note that both simulations have perturbed initial states. The drift in both cases is due to the mismatched orbits as well as environmental perturbations. It is clear from these figures that the guidance and control laws are necessary and successful in maintaining the desired separation between spacecraft.

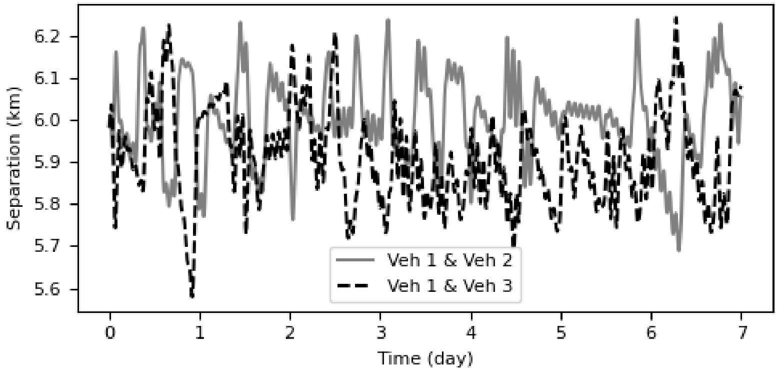


Fig. 9: Spacecraft separation distances for a controlled simulation.

4.5 Control Usage

An example of the control usage for all norms is shown in Fig. 11, 12, and 13 where the ΔV usage over time is plotted. The full time history is shown in the left plot of each pair while the right plot details the control usage indicated by the gray bar and arrow in the left plot. Note that while these plots correspond to the trajectories in Fig. 8, the trajectories shown in Fig. 8 only cover the first day of the sim while these control usage plots represent a week long simulation.

Examining the left of each pair of plots (Fig. 11a, 13a, 12a), it can be seen that each norm has approximately the same number of control events - roughly two or three events per day. As will be shown later, this is reflective of the overall control usage and remains generally consistent across all norms. Another observation can be made that the L_1 and L_∞ norms have higher

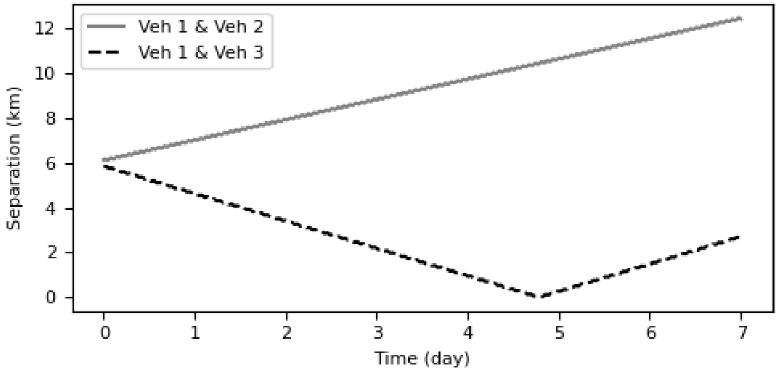
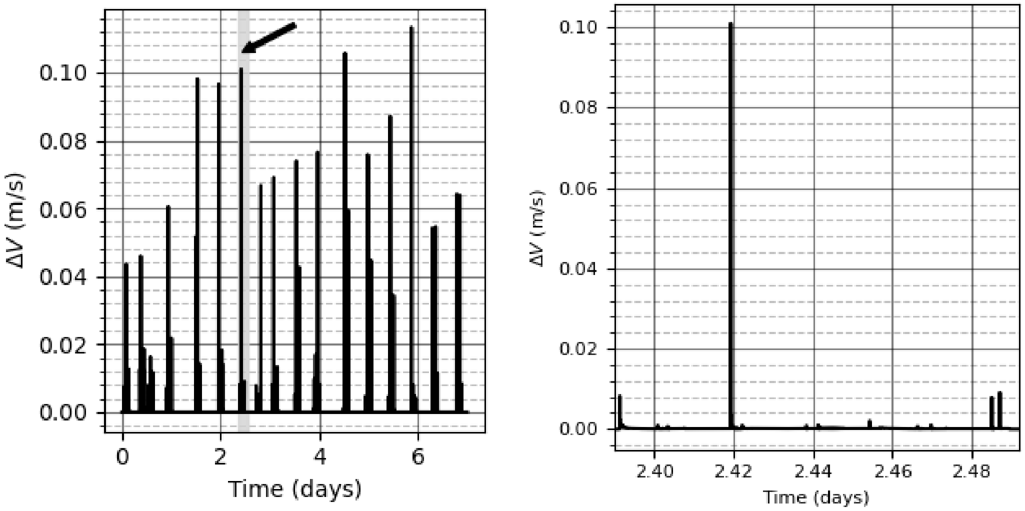


Fig. 10: Spacecraft separation distances for an uncontrolled simulation.



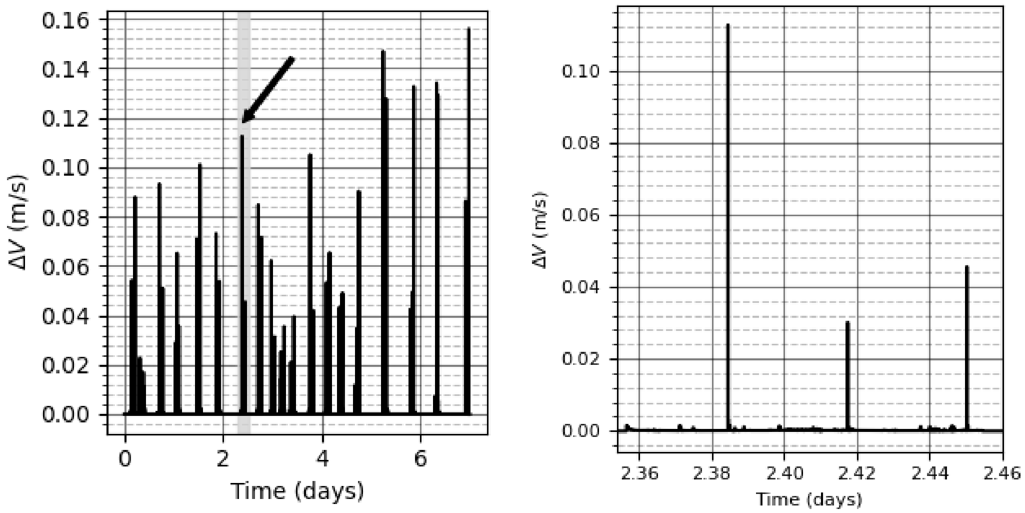
(a)

(b)

Fig. 11: ΔV usage with L_1 norm (11a) and with the section indicated by the arrow magnified (11b)

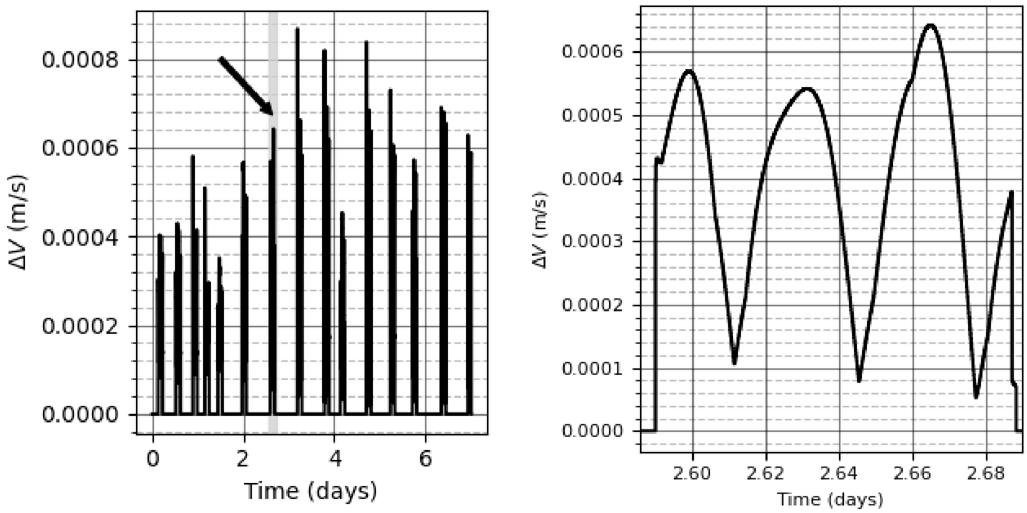
instantaneous control usage relative to the L_2 norm, with the difference being multiple orders of magnitude.

From the right hand plots (Fig. 11b, 13b, 12b), with higher time resolution, we can see the control usage over the course of a single maneuver. For the L_1 and L_∞ simulations, the results are quite similar. In both instances there are two peaks of high control usage with numerous smaller instances interspersed throughout the full maneuver. The periods of notable control usage tend to fall around multiples of half the orbital period. When compared with a Hohmann transfer, roughly impulsive controls spaced half an orbital period apart, intuitively this result makes sense as a low ΔV transfer trajectory.



(a) (b)

Fig. 12: ΔV usage with L_∞ norm (12a) and with the section indicated by the arrow magnified (12b)



(a) (b)

Fig. 13: ΔV usage with L_2 norm (13a) and with the section indicated by the arrow magnified (13b)

Looking at the L_2 norm in Fig. 13, we see a very different control pattern. Rather than impulsive type controls, the spacecraft uses a much smoother continuous control that oscillates in time. As with the other norms, these

Table 7: Initial ROE state dispersion for the Monte Carlo simulations

ROE State	Initial Dispersion (unitless)
δa	$\mathcal{U}(-1.454 \times 10^{-4}, 1.454 \times 10^{-4})$
$\delta\lambda$	$\mathcal{U}(-7.270 \times 10^{-4}, 7.270 \times 10^{-4})$
δe_x	$\mathcal{U}(-2.056 \times 10^{-4}, 2.056 \times 10^{-4})$
δe_y	$\mathcal{U}(-2.056 \times 10^{-4}, 2.056 \times 10^{-4})$
δi_x	$\mathcal{U}(-2.056 \times 10^{-4}, 2.056 \times 10^{-4})$
δi_y	$\mathcal{U}(-2.056 \times 10^{-4}, 2.056 \times 10^{-4})$

oscillations tend to have a period of roughly half the orbital period. There are some exceptions, particularly in what might be termed the transitory period when the controller first actives, but in general the pattern holds.

These difference in control usage may be considered significant when considering their application. If a given spacecraft has a high thrust, non-throttlable, on/off type propulsion system, using the L_1 or L_∞ norm might be more appropriate, whereas if the propulsion system is a smaller thrust, longer duration burning system, such as an electric propulsion system, the L_2 norm formulation would be more applicable.

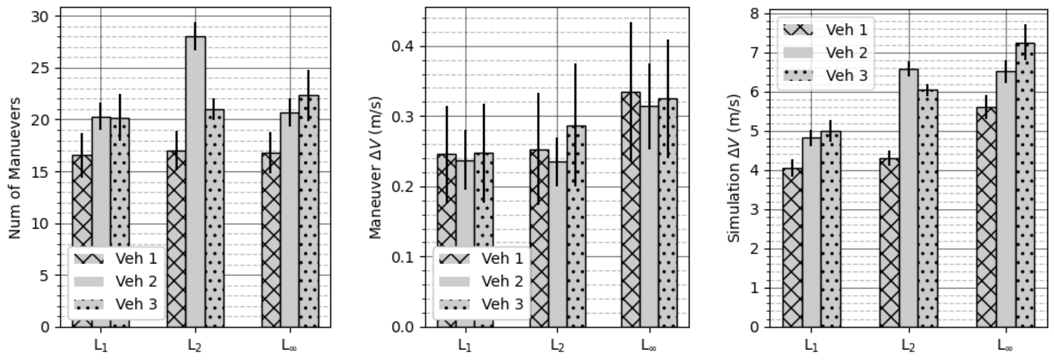
4.6 Cumulative Simulation Results

A series of Monte Carlo simulations were run to determine the overall characteristics of the guidance and control law. In each individual trial, the initial ROE states were randomly perturbed from the desired trajectory according to the uniform distributions shown in Table 7 and then propagated until one of the vehicles in the formation was about to exceed the inner sphere at which point the data collection began. In Cartesian coordinates, the initial position perturbations were found to range from a minimum of 1.85 m to 251.10 m with a mean initial error of 115.46 m. Each trial used one of the three norms, L_1, L_2, L_∞ and each norm type was run 100 times for a total of 300 trials.

The cumulative results of the Monte Carlo simulations are now presented. Each metric is broken down by both the norm definition used as well as the spacecraft orbit and encapsulates the full week-long simulation. The uncertainty bars on the data indicate a single standard deviation from the mean.

In Fig. 14a, the number of times a vehicle needed to maneuver is given. In this context, a maneuver is defined to be the period where the guidance and control laws are operating. There may be periods of time where the control law is active but determines that the optimal solution is to not apply any control, but we do not consider these as separate maneuver events.

For all vehicles across all norms, the number of maneuvers is generally the same except the second vehicle using the L_2 norm that shows a significantly higher number of maneuvers. With an average of 15 to 23 maneuvers over a week-long period, the spacecraft are maneuvering between 2.14 and 3.3 times a day. Since the number of maneuvers is linked to how far the spacecraft is allowed to drift, increasing the size of the keep-in volume should decrease the



(a) (b) (c)

Fig. 14: 14a shows the number of maneuvers used to control the spacecraft, 14b shows the average ΔV used per maneuver, and 14c shows the total ΔV used in each simulation of a seven day period.

number of maneuvers in any given time period. An evaluation of the number of maneuvers required is important, as the time spent maneuvering often is time that the spacecraft is unable to perform its primary mission.

The average ΔV used over the duration of each maneuver is shown in Fig. 14b. In other words, this is the amount of ΔV that is necessary to correct a spacecraft's trajectory once. The results show that the L_1 and L_2 norms use about the same ΔV per maneuver with the L_∞ using a greater amount. While there is variability, the amount of variability seems to be consistent across all norms and vehicles.

The total ΔV used over the course of each week-long simulation is shown in Fig. 14c. This shows the combined effects of the two previous metrics and encapsulates the total ΔV needed for station keeping operations. The L_1 norm has a relatively low and consistent ΔV usage for all vehicles. The L_∞ norm has the greatest ΔV usage. The L_2 norm shows inconsistent results with the first vehicle having usage comparable to the L_1 norm and the other vehicles having usage comparable to the L_∞ norm. While these results are reasonable for a CubeSat or small satellite mission, it is worth considering the implications on the mission. The average ΔV for the L_2 over the three spacecraft is a full meter per second higher than the L_1, and the L_∞ norm is a full two meters per second higher than the L_1. Over the course of a week these numbers may seem insignificant, but over the course of a mission lifetime of 1-3 years this difference would amount to 50-300 m/s of additional required ΔV. For CubeSat or small satellite missions an increase in a ΔV requirement of these magnitudes is significant and most likely mission prohibitive. The selection of norm used for a given mission should be considered during the concept and development phase of the mission.

Comparing Figs. 11a and 12a with Fig. 13a shows that the instantaneous ΔV used by the L_1 and L_∞ norms is significantly higher than that of the L_2 norm but only occur in brief spikes. Integrating these over the full duration of each maneuver or simulation, as seen in Figs. 14b and 14c, respectively, show the total ΔV usage. From these, it is clear that there is no overall difference in ΔV usage between the impulsive and continuous behavior.

Overall, the L_1 norm produces the lowest overall ΔV result. However, there may be cases where the L_2 norm should bear consideration. For example, due to the continuous low thrust control profile the L_2 norm generates, it may be well suited for missions with low impulse electric propulsion systems.

4.7 Potential Future Work

During development of this method, it was seen that the multiple sphere-based boundaries may be overly conservative and computationally intensive. An alternative approach would be to use the ROE states and dynamics to define the allowable boundaries. This would allow the boundaries for a repeating relative orbit to be described as a simple set of time-invariant state constraints. Additionally, defining the formation and boundaries would be simplified due to the time-invariant nature of the ROEs. These improvements are anticipated for future work.

5 Conclusion

This paper presents an MPC-based architecture used in conjunction with a sphere shaped polytope boundary constraint and the HCW equations to maintain a desired trajectory of a cluster of small satellites. These boundary constraints allow the satellites the freedom to drift and maneuver within the allowable bounds, while still maintaining flight safety and the mission formation. The surfaces allow multiple agents to be flown in a single formation with no overlapping regions. This allows the vehicles to execute avoidance strategies and maintain flight safety without continually maintaining the trajectories of other agents. The operational polytope boundaries enable the model predictive control framework to be used to make small maneuvers to correct perturbations from its desired trajectory. A fuel optimal guidance trajectory is also presented. This trajectory is used as a guide for the MPC to follow, lowering the overall fuel usage. The model predictive control framework combined with the convex polytope boundary enables a scalable method that can support clusters of satellites to safely achieve mission objectives while minimizing fuel usage. This framework also requires less overall ΔV when compared to a continuous controlled system and an LQR framework. This approach also reduces the terminal error when compared to an open-loop system.

This paper shows that if formation control is not performed the formation is not maintained and potential collision events may occur.

This paper also presents and compares the formulation of using L_1, L_2, and L_∞ norm objective functions for the MPC. The simulations show that the control formulation is able to maintain its position within the polytope volume with reasonable computation time requirements for all three norm formations. As would be expected, the computation time for solving the MPC problem increase with the desired time horizon. However, the L_2 norm showed the slowest computation time growth while the L_1 and L_∞ norms grew at similar rates. In the simulations, all the norms were able to run at near real-time rates or better despite not optimizing for computation time. This paper also shows that the L_1 norm produces the lowest total ΔV, over a seven day period, with the L_2 norm requiring a meter per second more than the L_1, and the L_∞ norm being a full two meters per second higher than the L_1 norm.

Comparing the number of times that each spacecraft was required to maneuver showed roughly equivalent performance between the three different norms. However, the L_1 norm required the lowest amount of ΔV, followed by the L_2 norm and then the L_∞ norm.

A Appendix

A.1 Convex Polytope Keep-in Volume Development

The current work presents the development of a sphere-shaped polytope switching surface that is used to generate the switching conditions. This approach alleviates the concern presented by [23] that a six-sided error box constraint could give initial conditions to the programming problem that result in higher fuel costs, such as when a trajectory ends near the corner of the error box with little room to maneuver.

Keep-in volumes can be leveraged to force the spacecraft to stay within a given area, allowing the formation to maintained. The agent spacecraft are able to drift and not required to constantly be maneuvering. The use of a convex polytope as the boundary constraint allows for a high degree of freedom in the possible constraints and for the formulation of the problem as a linear program. This work leverages the polytope boundary constraint developed by Smith et al., [25].

At each timestep k, the desired keep-in volume is defined by a convex polytope made of M faces. The desired state, $\mathbf{x}_{d,k}$, lies within the volume. As the desired state changes with time, the polytope must also be updated at each timestep. The M planes are each defined by a normal vector $\hat{\boldsymbol{\eta}}$, that is assumed to point towards the interior of the polytope, and by a point $\mathbf{p}$. If the vector $\mathbf{r}$ is considered an arbitrary point, and is on the border or interior of the polytope then the dot product will satisfy

$$\hat{\eta} \cdot (\mathbf{r} - \mathbf{p}) \geq 0. \tag{34}$$

A matrix constraint consisting of position and velocity vectors is used to maintain the spacecraft state, $\mathbf{x}_k$, inside the polytope at a given time k. This is formulated as

$$\mathbf{A}_{poly,k}\mathbf{x}_k \leq \mathbf{b}_{poly,k} \tag{35}$$

where

$$\mathbf{A}_{poly,k} = \begin{bmatrix} -\hat{\eta}_{1,k}^T & \mathbf{0}_{1\times 3} \\ -\hat{\eta}_{2,k}^T & \mathbf{0}_{1\times 3} \\ \vdots & \vdots \\ -\hat{\eta}_{M,k}^T & \mathbf{0}_{1\times 3} \end{bmatrix} \qquad \mathbf{b}_{poly,k} = \begin{bmatrix} -\hat{\eta}_{1,k}^T \mathbf{p}_{1,k} \\ -\hat{\eta}_{2,k}^T \mathbf{p}_{2,k} \\ \vdots \\ -\hat{\eta}_{M,k}^T \mathbf{p}_{M,k} \end{bmatrix} \tag{36}$$

with $\hat{\eta}_{i,k}$ and $\mathbf{p}_{i,k}$ being the normal and point associated with the i^{th} face of the polytope at time k.

As previously discussed, this work defines the polytope as a close approximation of a sphere, although any convex shape could be utilized.

Acknowledgments. This work was funded by Space Dynamics Laboratory.

Declarations

- Conflict of interest/Competing interests: On behalf of all authors, the corresponding author states that there is no conflict of interest.
- Ethics approval: Not applicable
- Consent to participate: Not applicable
- Consent for publication: Granted
- Availability of data and materials: Not applicable
- Code availability: No

References

[1] D'Amico, S., Ardaens, J.S., De Florio, S.: Autonomous Formation Flying Based on GPS - PRISMA Flight Results. Acta Astronautica **82**(1), 69–79 (2013). https://doi.org/10.1016/j.actaastro.2012.04.033

[2] Fridlund, C.V.M.: Darwin-The Infrared Space Interferometry Mission. In: ESA Bulletin 103, pp. 20–63 (2000)

[3] Mesbahi, M., Hadaegh, F.Y.: Formation Flying Control of Multiple Spacecraft via Graphs Matrix Inequalities and Switching. Journal of Guidance Control and Dynamics **24**(2), 369–377 (2001). https://doi.org/10.2514/2.4721

[4] Ticker, R.L., Azzolini, J.D.: 2000 Survey of Distributed Spacecraft Technologies and Architectures for NASA's Earth Science Enterprise in the 2010-2025 Time Frame. Technical Report 209964, NASA (2000)

[5] Martin, M., Klupar, P., Kilberg, S., Winter, J.: TechSat 21 and Revolutionizing Space Missions using Microsatellites. USU/AIAA Small Satellite Conference (2001)

[6] Bristow, J., Folta, D., Hartman, K.: A formation flying technology vision. Space 2000 Conference and Exhibition (2000). https://doi.org/10.2514/6.2000-5194

[7] Scharf, D.P., Hadaegh, F.Y., Ploen, S.R.: A Survey of Spacecraft Formation Flying Guidance and Control (Part II): Control. Proceedings of the American Control Conference **4**(Part 11), 2976–2985 (2004). https://doi.org/10.23919/acc.2004.1384365

[8] Krieger, G., Moreira, A., Fiedler, H., Hajnsek, I., Werner, M., Younis, M., Zink, M.: TanDEM-X: A Satellite Formation for High-Resolution SAR Interferometry. IEEE Transactions on Geoscience and Remote Sensing **45**(11), 3317–3340 (2007). https://doi.org/10.1109/TGRS.2007.900693

[9] Fiedler, H., Krieger, G., Werner, M., Reiniger, K.: The TanDEM-X Mission Design and Data Acquisition Plan. In: European Conference on Synthetic Apeture Radar 2006, pp. 16–18 (2006)

[10] Roth, N.H., Risi, B., Grant, C.C., Zee, R.E.: Flight Results from the CanX-4 and CanX-5 Formation Flying Mission. In: 29th Annual AIAA/USU Conference on Small Satellites, pp. 1–15 (2016). https://doi.org/10.1016/j.jcrysgro.2014.02.053

[11] Tapley, B.D., Bettadpur, S., Watkins, M., Reigber, C.: The Gravity Recovery and Climate Experiment: Mission Overview and Early Results. Geophysical Research Letters **31**(9), 1–4 (2004). https://doi.org/10.1029/2004GL019920

[12] Bertiger, W., Bar-Sever, Y., Bettadpur, S., Dunn, C.: GRACE: Millimeters and Microns in Orbit. In: ION GPS 2002 Conference, pp. 2022–2029 (2002). https://doi.org/10.1.1.475.4059

[13] Farahmand, M., Long, A., Hollister, J., Rose, J., Godine, D.: Magnetospheric Multiscale Mission Navigation Performance During Apogee-Raising and Beyond. Advances in the Astronautical Sciences **162**, 2738–2739 (2018)

[14] Winternitz, L.B., Bamford, W.A., Price, S.R., Carpenter, J.R., Long, A.C., Farahmand, M.: Global Positioning System Navigation Above 76,000 KM for NASA'S Magnetospheric Multiscale Mission. Journal of the Institute of Navigation **64**(2), 289–300 (2017). https://doi.org/10.1002/navi.198

[15] Wang, J., Zhang, D., Chen, G., Shao, X.: A New Fuel-Balanced Formation Keeping Reference Trajectories Planning Method. Aircraft Engineering and Aerospace Technology **90**(6), 927–936 (2018). https://doi.org/10.1108/aeat-03-2017-0084

[16] Di Mauro, G., Lawn, M., Bevilacqua, R.: Survey on Guidance Navigation and Control Requirements for Spacecraft Formation-Flying Missions. Journal of Guidance, Control, and Dynamics **41**(3), 581–602 (2018). https://doi.org/10.2514/1.G002868

[17] Li, J., Xi, X.: Fuel-optimal low-thrust reconfiguration of formation- flying satellites via homotopic approach. Journal of Guidance, Control, and Dynamics **35**(6), 1709–1717 (2012). https://doi.org/10.2514/1.57354

[18] Mazal, L., Gurfil, P.: Cluster Flight Algorithms for Disaggregated Satellites. Journal of Guidance, Control, and Dynamics **36**(1), 124–135 (2013). https://doi.org/10.2514/1.57180

[19] Rocco, E.M., De Oliveira E Souza, M.L., De Almeida Prado, A.F.B.: Station Keeping of Constellations Using Multi-Objective Strategies. Mathematical Problems in Engineering **2**(1) (2013). https://doi.org/10.1155/2013/476451

[20] Chernick, M., D'Amico, S.: New closed-form solutions for optimal impulsive control of spacecraft relative motion. Journal of Guidance, Control, and Dynamics **41**(2), 301–319 (2018). https://doi.org/10.2514/1.G002848

[21] Breger, L.S.: Model Predictive Control for Formation Flying Spacecraft. Master's thesis, Massachusetts Institute of Technology (2004)

[22] Lim, Y., Jung, Y., Bang, H.: Robust Model Predictive Control for Satellite Formation Keeping with Eccentricity/Inclination Vector Separation. Advances in Space Research **61**(10), 2661–2672 (2018). https://doi.org/10.1016/j.asr.2018.02.036

[23] Tillerson, M., How, J.P.: Advanced Guidance Algorithms for Spacecraft Formation-Keeping. Proceedings of the American Control Conference **4**, 2830–2835 (2002). https://doi.org/10.1109/ACC.2002.1025218

[24] Rawlings, J.B., Mayne, D.Q., Diehl, M.M.: Model Predictive Control: Theory, Computation, and Design, 2nd edn. Nob Hill Publishing,, Santa Barbara, California (2019). https://doi.org/10.1155/2012/240898

[25] Smith, T.K., Akagi, J., Droge, G.: Model Predictive Control Switching Strategy For Safe Small Satellite Cluster Formation Flying. Journal of Aerospace Systems (2023). https://doi.org/10.1007/s42401-023-00237-2

[26] Cheng, B., Li, Z.: Fully Distributed Event-Triggered Protocols for Linear Multi-Agent Networks. IEEE Transactions on Automatic Control **64**(4), 1655–1662 (2019)

[27] Cheng, B., Wu, Z., Li, Z.: Distributed Edge-Based Event-Triggered Formation Control. IEEE Transactions on Cybernetics, 1–12 (2019)

[28] Hu, Q., Shi, Y.: Event-Based Coordinated Control of Spacecraft Formation Flying Under Limited Communication. Nonlinear Dynamics **99**(3), 2139–2159 (2020). https://doi.org/10.1007/s11071-019-05396-6

[29] Smith, T., Akagi, J., Droge, G.: Spacecraft Formation Flying Control Switching Surface Based on Relative Orbital Elements. 2022 AAS/AIAA Astrodynamics Specialist Conference, 1–19 (2022)

[30] Ren, W., Beard, R.W.: Decentralized Scheme for Spacecraft Formation Flying via the Virtual Structure Approach. Journal of Guidance Control and Dynamics **27**(1), 73–82 (2004)

[31] Hill, G.W.: On the part of the motion of lunar perigee which is a function of the mean motions of the sun and moon. Acta Math **8**, 1–36 (1886). https://doi.org/10.1007/BF02417081

[32] Wiltshire, R.S., Clohessy, W.H.: Terminal guidance system for satellite rendezvous. Journal of The Aerospace Sciences **27**(9), 663–674 (1960). https://doi.org/10.2514/8.8704

[33] Curtis, H.: Orbital Mechanics for Engineering Students, 2nd edn. Elsevier,, Oxford, United Kingdom (2009)

[34] Borrelli, F., Bemporad, A., Morari, M.: Predictive Control for Linear and Hybrid Systems. Cambridge University Press,, Cambridge, United Kingdom (2017). https://doi.org/10.1017/9781139061759

[35] Brown, A.G., Ruschmann, M.C., Duffy, B., Ward, L., Hur-Diaz, S., Ferguson, E., Stewart, S.M.: Simulated Annealing Maneuver Planner for Cluster Flight. International Symposium on Space Flight Dynamics (1), 1–31 (2014)

[36] Hur-Diaz, S., O'Connor, B.: Cluster Flight Application On System F6. In: 24th International Symposium on Space Flight Dynamics, pp. 1–16 (2014)

[37] D'Amico, S.: Autonomous Formation Flying in Low Earth Orbit. PhD thesis, Delft University of Technology (2010)

[38] Sullivan, J., Grimberg, S., D'Amico, S.: Comprehensive survey and assessment of spacecraft relative motion dynamics models. Journal of Guidance, Control, and Dynamics **40**(8), 1837–1859 (2017). https://doi.org/10.2514/1.G002309

[39] Anon: U.S. Standard Atmosphere, 1976. Technical report (1976)

[40] Yamanaka, K., Ankersen, F.: New state transition matrix for relative motion on an arbitrary elliptical orbit. Journal of Guidance, Control, and Dynamics **25**(1), 60–66 (2002). https://doi.org/10.2514/2.4875

[41] Smith, T.K., Lewis, Z., Olsen, K., Bulcher, M.A., Whitmore, S.A.: A Miniaturized, Green, End-Burning, and Sandwich Hybrid Propulsion System. Journal of Propulsion and Power **38**(5), 701–713 (2022). https://doi.org/10.2514/1.B38623

[42] Gurobi Optimization, LLC: Gurobi Optimizer Reference Manual (2023). https://www.gurobi.com

CHAPTER 6

Model Predictive Control For Formation Flying Based on D'Amico Relative Orbital Elements

Model Predictive Control for Formation Flying Based on D'Amico Relative Orbital Elements

Tyson K. Smith[1] (✉), John Akagi[2], and Greg Droge[3]

1. Space Dynamics Laboratory, North Logan, Utah, 84341, United States
2. Space Dynamics Laboratory, North Logan, Utah, 84341, United States
3. Electrical and Computer Engineer Department, Utah State University, Logan, Utah, 84322, United States

✉ tyson.smith@sdl.usu.edu

Abstract

The desire to fly small spacecraft close together has been a topic of increasing interest over the past several years. This paper presents the development and analysis of a model predictive control based framework that is used with the D'Amico relative orbital elements (ROEs) to maintain the desired trajectories of a cluster of spacecraft while also allowing freedom to maneuver within some allowable bounds. Switching surfaces based on the ROE constraints contain the full state of the system, allowing for fuel reduction over other approaches that use the Hill-Clohessy-Wiltshire equations. The formation and boundary constraints are designed such that no two agents have overlapping regions, allowing the vehicles to maintain safety of flight without continually maintaining the trajectories of other agents. This framework allows for a scalable method that can support clusters of satellites to safely achieve mission objectives while minimizing fuel usage. This paper provides simulated results of the framework for a three spacecraft formation that demonstrates a 67% fuel reduction when compared to previous approaches.

Keywords

Formation Flying · Model Predictive Control · Relative Orbital Elements · Switching Surfaces

Nomenclature

$\boldsymbol{\omega}_E$	Angular Velocity of the Earth $(\mathrm{rad/s})$
$\delta\lambda$	Relative Mean Longitude
δa	Relative Semi-major Axis
$\delta e_x, \delta e_y$	x and y Relative Eccentricity Components
$\delta i_x, \delta i_y$	x and y Relative Inclination Components
$\delta v_r, \delta v_t, \delta v_n$	Velocity Changes in the Radial, Tangential, and Normal Axes $(\mathrm{m/s})$
$\hat{\mathbf{x}}$	Predicted ROE State
μ	Standard Gravitational Parameter of the Earth $(\mathrm{m^3 s^{-2}})$
Ω	Right Ascension of Ascending Node (rad)
ω	Argument of Perigee (rad)
ρ	Air Density $(\mathrm{kg/m^3})$
$\mathbf{A}$	Relative Orbital Element State Matrix
$\mathbf{A}_D$	ROE State Transition Matrix
$\mathbf{B}$	ROE Input Matrix
$\mathbf{v}$	Control Vector of Radial, Tangential, and Normal Velocity Changes $(\mathrm{m/s})$
$\mathbf{v}_{rel}$	Relative Velocity of the Spacecraft $(\mathrm{m/s})$
$\mathbf{x}_d$	Desired ROE State
$\mathbf{x}_e$	ROE State Error

A	Cross Sectional Area (m)
a	Semi-Major Axis (m)
C_d	Coefficient of Drag
e	Eccentricity
f	True Anomaly (rad)
g_0	Standard Gravity $(\mathrm{m/s^2})$
i	Inclination (rad)
I_{sp}	Specific Impulse (s)
J_2	J_2 Coefficient of the Earth
M	Mean Anomaly (rad)
m	Mass (kg)
n_c	Mean Motion of Reference Orbit $(\mathrm{rad/s^2})$
R_E	Radius of Earth
P	Terminal Error Cost Matrix
Q	State Error Cost Matrix
R	Control Usage Cost Matrix

1 Introduction

Spacecraft formation flight operations have become a topic of interest in recent years. There are dozens of missions that have flown or are currently in development that require some form of coordinated motion between two or more spacecraft, e.g., [1–11]. Formation flying unlocks a wide range of applications, such as various coordinated sensing and communication missions, e.g. [2, 12]. The coordinated flight also allows for many lower cost, small spacecraft to fly varied payloads that traditionally would have been flown on a single larger and more expensive spacecraft [13]. A key component to space formation flying is the requirement that vehicles limit fuel usage to increase the lifetime of their missions. Formations also require that the spacecraft's relative states are maintained, which requires some type of control for orbital adjustments. This must be done while respecting the computational capabilities of the individual platforms and the communication restrictions within the constellation. This paper presents an optimization-based feedback control strategy that allows each satellite in the formation to fly within a safe region of operation to maintain relative spacing with little to no communication required. A model predictive control (MPC) approach based on the D'Amico relative orbital elements (ROEs) is developed and analyzed with respect to overall fuel usage and computation requirements.

Several considerations arise when developing a formation flying control framework. These include: fuel usage, communication requirements between spacecraft, collision avoidance, overcoming disturbances, mission operational constraints, and computation time requirements. Creating a formation flying framework becomes a balancing act between these different objectives.

Various techniques have been studied previously to employ optimization strategies in an effort to reduce fuel consumption and overcome disturbances from perturbing forces [14–18]. While an optimal fuel saving maneuver may be found in theory, controller feedback is essential for overcoming disturbances from the invariably erroneous motion models. MPC is one such method for using optimal control techniques within a feedback loop to balance multiple desired outcomes [19]. The optimization problem is solved given the current state of the system, one or more control inputs from the resulting optimal control trajectory are executed, and the process is repeated with the new state used as the initial condition. Several researchers have implemented an MPC framework based on the Hill-Clohessy-Wiltshire (HCW) relative orbital dynamic equations [20–26] and were able to maintain control of a spacecraft within their desired tolerances. The HCW dynamics are attractive to use with MPC because they are a well understood, linear, time-invariant model. Additionally, the use of Cartesian coordinates provides an intuitive understanding of the spacecraft states.

However, using Cartesian coordinates, or the HCW equations, to define a spacecraft's state can result in some complications as well. For instance, given a Cartesian state, it is generally non-trivial to determine the full relative orbit of the spacecraft or attempt to match some orbital element of a second spacecraft. Additionally, using the standard Euclidean definition of distance between two states may not produce the most meaningful results. Consider, for example, a spacecraft whose position perfectly matches some desired location but has a velocity perpendicular to the desired velocity. The distance would be considered “close" in a Euclidean sense, but because of the direction of the velocity vector, these orbits are significantly different. The difficulty of matching the full relative orbital elements and velocities lead to the utilization of D'Amico's ROEs in this work.

Collision avoidance also needs to be considered when performing autonomous formation control. Lim et al., [16], guaranteed collision avoidance by developing an MPC framework that maintained the eccentric and inclination vectors of the D'Amico ROEs parallel to each other. Maintaining the ROE eccentricity and inclination vectors parallel to each other guarantees collision avoidance within an established formation, but it does not guarantee a collision free transfer trajectory nor does it guarantee the formation itself is maintained. DiMauro et al., [18], used a mixed integer linear programming (MILP) framework with a six-sided box as a keep out volume for the deputy spacecraft, but DiMauro found that the MILP approach increased the computation time to a degree that it was deemed not suitable for flight. Tillerson et al., [17], also used a six-sided Euclidean box as a switching condition, but postulated that using a higher dimensional shape could reduce fuel costs.

The authors of this paper developed several alternatives to reduce fuel costs due to the use of a six-sided error box [24, 26, 27]. [24] develop a model based on Cartesian coordinates that used a nested sphere based constraints to determine when to activate and deactivate the control. [26] built upon the work of [24] by adding a guidance trajectory for the controller to follow that further reduced fuel usage. [27] used a similar approach, but with the outer-most boundary being an approximation of ROE constraints in local-vertical local-horizontal (LVLH) space. In both approaches, the controller was found to be overly reactive to transient state errors and would immediately correct the error at the cost of fuel usage.

The major contributions of this work are the development and implementation of a guidance and model predictive control framework with the switching surfaces and dynamics based on the D'Amico ROEs. The amount of ΔV required to maintain the formation is reduced by combining an optimal guidance trajectory with an MPC feedback controller. The guidance trajectory finds a fuel-optimal trajectory that takes the spacecraft from its current state to the desired orbit path. The MPC framework provides a feedback control system with the ability to express constraints and weights in the objective that can provide for intuitive tuning to a desired behavior, such as overcoming disturbances and reducing fuel usage. The switching surface in this work is used as a boundary constraint. When the constraint is violated, a course correction maneuver is performed. This switching surface allows for the formation to be maintained as well as collision free flight within the formation. Using MPC and a switching surface based on the ROEs accounts for the interplay between the control inputs and the natural dynamic motion of the system that further reduce fuel usage when compared to a framework based on the HCW equations. Modeling the constraints and the control in ROE spaces allows the system to better match the desired orbit, not just a position state at a given time.

The remainder of the paper is outlined as follows. Section 2 gives an overview of the relative dynamics used for this work, background information into the model predictive control approach developed, as well as some discussion about switching control. Section 3 presents the control switching conditions, the guidance and control approach, and the MPC formulation. Section 4 presents simulation results for a three spacecraft formation executing the proposed guidance and control approach and compares it to previous work using the HCW equations. A significant decrease in required ΔV is obtained by using D'Amico's ROEs. Section 5 provides concluding remarks.

2 Background

This section presents background information relevant to the control scheme used for this work. The D'Amico ROEs, used for the guidance and control dynamics, are introduced, as well as an explanation of the general model predictive control framework. Finally, an explanation of the use of switching surfaces in spacecraft control is given.

2.1 Relative Dynamics Overview

Traditionally, the HCW equations have been used to describe the relative orbital dynamics of spacecraft, [28, 29]. More recently, however, relative orbital elements have been developed to describe relative dynamics in the context of Keplerian orbital elements. These ROEs allow the system to take advantage of the natural dynamics of the system. These are a set of states that define the orbit of a spacecraft relative to some reference orbit. Various ROE formulations have been proposed, notably a set created by Lovell and Spencer [30] and a set created by D'Amico [31]. For this work the D'Amico ROEs are chosen as they are derived directly from the standard orbital elements and are more amenable for the desired formation flying architecture as they alleviate the challenges of using Cartesian coordinates discussed in the previous section. As presented in this paper, the ROEs assume two-body motion with no perturbations, but can be adjusted to account for J_2 disturbances [32]. Although the ROE formulation used in this paper does not include the J_2 formulation, Section 4 includes J_2 perturbations in the simulated dynamics.

Given a set of standard orbital elements for the chief or reference orbit and a second set of standard orbital elements of a deputy or agent satellite, the ROEs, which describe the orbit of the deputy relative to the chief, can be determined. In the ROE states, the c and d subscripts indicate the elements of the chief and deputy, respectively. The orbital elements a, e, i, ω, Ω, and M are the semi-major axis, eccentricity, inclination, argument of perigee, right ascension of ascending node, and mean anomaly, respectively, [33]. The dimensionless ROEs, as defined by D'Amico [31], are given as

$$\mathbf{x} = \begin{bmatrix} \delta a \\ \delta\lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \end{bmatrix} = \begin{bmatrix} (a_d - a_c)/a_c \\ (f_d - f_c) + (\Omega_d - \Omega)\cos i_c \\ e_{x,d} - e_{x,c} \\ e_{y,d} - e_{y,c} \\ i_d - i_c \\ (\Omega_d - \Omega_c)\sin i_c \end{bmatrix}, \tag{1}$$

with

$$f_\square = \omega_\square + M_\square \ , \ e_{x,\square} = e_\square \cos\omega_\square \ , \ e_{y,\square} = e_\square \sin\omega_\square \ , \tag{2}$$

where the $\square$ subscripts indicate the states correspond to either the deputy or chief states.

The δa term represents the relative difference in semi-major axis and the $\delta\lambda$ term gives the relative mean longitude between the two spacecraft. The δe_x and δe_y terms can be composed into a single $\delta\mathbf{e}$ relative eccentricity vector where the magnitude gives the orbit size along the radial direction. Similarly, the δi_x and δi_y terms can create the $\delta\mathbf{i}$ relative inclination vector where the magnitude defines the size of the orbit along the normal axis. The $\delta\mathbf{e}$ and $\delta\mathbf{i}$ vectors can additionally be used to create passively safe orbits where the agent does not pass through the relative orbital plane in front of the leader [16]. Note that this relative state representation is nonsingular for circular orbits ($e_c = 0$), whereas it is singular for strictly equatorial orbits ($i_c = 0$) [18].

Assuming two body motion with no perturbations, the orbital elements defined in Eq. (2) are all constant except the mean argument of latitude, f, which increases at a constant rate, with μ representing the standard gravitation parameter, defined by

$$\dot{f} = \frac{df}{dt} = \sqrt{\frac{\mu}{a^3}}. \tag{3}$$

Defining $\Delta f = f_d - f_c$, then $\Delta \dot{f}$ represents the drift in relative mean argument of latitude that will occur when the spacecraft has a different semi-major axis than the reference orbit, and we define $\Delta a = a_d - a_c$. This can be approximated to the first order by the differencing of Eq. (3) for the two orbits as

$$\Delta \dot{f} = \frac{d(\Delta f)}{dt} = -\frac{3}{2}\sqrt{\frac{\mu}{a^5}}\Delta a = -\frac{3}{2}n\frac{\Delta a}{a} = -\frac{3}{2}n\delta a, \tag{4}$$

where n is the mean orbit motion and with the reasonable assumption that Δf and Δa are small compared to the inertial virtual leader orbit radius [31]. Assuming two body motion, the remainder of the inertial orbital elements are constant over time, and as such the other ROEs are also constant with respect to time. The full dynamics can then be represented as

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{3n}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta a \\ \delta \lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \end{bmatrix} = \mathbf{A}\mathbf{x}. \tag{5}$$

If f_M represents the mean argument of latitude of any given maneuver and the prefix Δ indicates the change in the specified ROE, the instantaneous changes of the ROEs ($\Delta\delta a$, $\Delta\delta\lambda$, $\Delta\delta e_x$, $\Delta\delta e_y$, $\Delta\delta i_x$, $\Delta\delta i_y$) as a result of impulsive ΔV applied along the tangential, radial, and normal axes (δv_t, δv_r, and δv_n, respectively) is [31],

$$\begin{array}{lclll} \Delta\delta a & = & & +\dfrac{2}{na}\delta v_t & \\ \Delta\delta\lambda & = & -\dfrac{2}{na}\delta v_r & & \\ \Delta\delta e_x & = & \dfrac{\sin f_M}{na}\delta v_r & +\dfrac{2\cos f_M}{na}\delta v_t & \\ \Delta\delta e_y & = & -\dfrac{\cos f_M}{na}\delta v_r & +\dfrac{2\sin f_M}{na}\delta v_t & \\ \Delta\delta i_x & = & & & +\dfrac{\cos f_M}{na}\delta v_n \\ \Delta\delta i_y & = & & & +\dfrac{\sin f_M}{na}\delta v_n \end{array} . \tag{6}$$

As can be seen, these are linear, time-varying and from Eq. 6 the input matrix, $\mathbf{B}$, which maps impulsive δv_r, δv_t, and δv_n velocity changes into instantaneous changes in the ROEs is found to be

$$\mathbf{B} = \begin{bmatrix} 0 & \frac{2}{na} & 0 \\ -\frac{2}{na} & 0 & 0 \\ \frac{sin f_M}{na} & \frac{2\cos f_M}{na} & 0 \\ -\frac{\cos f_M}{na} & \frac{2\sin f_M}{na} & 0 \\ 0 & 0 & \frac{\cos f_M}{na} \\ 0 & 0 & \frac{\sin f_M}{na} \end{bmatrix} . \tag{7}$$

The state transition matrix for $\mathbf{A}$ is found as $\mathbf{A}_D = e^{\mathbf{A}dt}$ where dt is the timestep. The control is represented as $\mathbf{v}$, with elements δv_r, δv_t, and δv_n, representing the impulsive radial, tangential, and normal velocity changes. An impulsive controls causes an instantaneous change in the ROE states and is described as

$$\mathbf{x}_i^+ = \mathbf{x}_i^- + \mathbf{B}\mathbf{v}_i, \tag{8}$$

where the “-" and “+" superscripts indicates the states before and after an instantaneous velocity change, respectively. Once the control is applied the state is propagated to the next timestep using the drifting dynamics

$$\mathbf{x}_{i+1}^{-} = \mathbf{A}_D \mathbf{x}_i^{+}. \tag{9}$$

Combining these so that the original state and the ROE change due to the control are propagated simultaneously gives

$$\mathbf{x}_{i+1}^{-} = \mathbf{A}_D (\mathbf{x}_i^{-} + \mathbf{B}\mathbf{v}_i). \tag{10}$$

The superscripts are now dropped since everything is in terms of the measured state at each timestep before any maneuver is applied.

$$\mathbf{x}_{i+1} = \mathbf{A}_D \mathbf{x}_i + \mathbf{A}_D \mathbf{B}\mathbf{v}_i. \tag{11}$$

The ROE model control inputs $\mathbf{v}_i$ rely on impulsive ΔV maneuvers as the inputs, but the simulation dynamics use a zero-order hold with accelerations applied from the thruster models. However, with the dynamics being discretized with a sufficiently small time step, an impulsive ΔV can be approximated as a constant acceleration by

$$\mathbf{v}_i = \mathbf{a}_i \; dt, \tag{12}$$

at each time step. Thus, the acceleration limits are converted to ΔV limits based on the thruster parameters and discretization time step, the ΔV control $\mathbf{v}_i$ at each step is solved for, and then each $\mathbf{v}_i$ of the solution is converted to an acceleration $\mathbf{a}_i$ to be applied to the system.

2.2 Model Predictive Control

Optimal control is a tool often used within spaceflight design to find the most fuel-efficient strategy to perform a correction maneuver. In general, there are two main limitations that often occur when using an optimal control solution in practice. First, even with a favorable optimization formulation, a horizon time that is long enough to produce desirable convergence characteristics may prove computationally prohibitive. Second, the model of the system is usually inaccurate or incomplete and the system may be impacted by external disturbances that can cause it to diverge from any calculated solution.

One approach to mitigate these concerns is to implement a feedback loop where the problem is repeatedly solved over a short time horizon. Once a solution is obtained, the first control in the solution is applied to the system and then the resulting state is measured. The problem can then be solved again, using the new measurement as the initial condition, and the first step of the new solution is applied to the system. This process of repeatedly solving the problem, implementing the first step of the solution, measuring the output, and repeating is referred to as Model Predictive Control (MPC).

While MPC generally loses any guarantee of optimality, it does provide the ability to express constraints, which is not common in many feedback control solutions. Furthermore, weights in the objective can provide intuitive “control knobs” for tuning to the desired behavior.

The L_1 norm is used for this work as it has previously been found to be more efficient for similar applications, [26], but other norms could also be used with this framework. Given an initial state of x_0, and a time horizon of N intervals, the L_1 objective function can be written as [34]

$$J = \sum_{k=1}^{N-1} ||Qx_k||_1 + \sum_{k=0}^{N-1} ||Ru_k||_1 + ||Px_N||_1, \tag{13}$$

where R, Q, and P are the weightings on control usage, state error, and terminal error, respectively.

2.3 Control Approaches

In general, a spacecraft is not able to maneuver continuously due to thruster duration constraints or mission requirements. This naturally lends itself to techniques for switched and hybrid systems where the controller changes in response to some trigger [35, 36], for example the violation of a position boundary constraint. For this work, a switching strategy is used to determine when to employ the MPC and when to allow the satellite to coast.

A strength of switching controllers is their simplicity. The control can be as simple as switching between two states, e.g., on/off. Another major advantage, specific to this work, is that the switching surfaces can be leveraged as operational boundary constraints. These constraints force the state to stay within a designated volume which, assuming non-overlapping boundaries for each spacecraft, guarantees the formation stays collision free. This also allows the spacecraft to drift while guaranteeing the spacecraft stays within the formation. This allows the spacecraft to stay within a safe region and be able to meet operational constraints.

Previous work in the area of spacecraft formation flying switching strategies were based on maintaining maximum and minimum relative distances [37] or six-sided error boxes [17]. Smith et al developed a framework that leverage a sphere-shaped switching condition and a polytope constraint based on the D'Amico ROEs, [24, 27], but found that in both cases the polytope formulation significantly impacted computation requirements. This work builds upon these previous works by using D'Amico ROEs to described the dynamics and the state constraints, alleviating the computation requirements in [24, 27]. This work also builds upon the work done in [26] by replacing the MPC model developed using the HCW equations and replacing it with one that uses the D'Amico ROEs. Methods of using Cartesian coordinates only accounted for position deviations while the ROE constraints contain the full state of the system. An ROE based switching surface and controller allows for the algorithm to take advantage of the natural dynamic motion of the system, further reducing fuel usage.

3 Control Framework Development

This work implements a guidance and control algorithm based on an MPC framework that keeps a spacecraft within some desired boundary constraints by using periodic course correction maneuvers. Within this, there are three key pieces that will be explained in this section: first, the boundary constraints and switching surface used to determine when maneuvers are necessary; second, the guidance law that finds a fuel optimal corrective trajectory; and third, the control law that follows the corrective trajectory.

To determine if a correction maneuver is required, the spacecraft's propagated state is compared to the switching surface. If the spacecraft is predicted to cross the surface, then the guidance trajectory plans a maneuver that will return the spacecraft to the desired trajectory. This maneuver plan is then passed to the MPC for execution, while correcting for any perturbations. Once the plan has been completed, the spacecraft re-enters its drifting state and again begins checking its propagated state over the drift horizon.

Note that for this work a maneuver is define as the entire period required to return the spacecraft to the desired trajectory and a single maneuver may consist of several thrust events. The structure of the algorithm is depicted in Algorithm 1 and the individual elements are described further in the following sections.

3.1 Switching Strategy

Smith et al. [26] developed a switching strategy based on the HCW equations. This work uses a similar approach but is adapted to work with the D'Amico ROEs. Since the spacecraft is unable to maneuver constantly, a strategy is required to select the periods where the spacecraft is drifting or maneuvering. The strategy chosen for this work is a switching surface that activates the controller when the spacecraft is predicted to reach the surface within some predetermined time frame, referred to as the drift horizon.

Algorithm 1: Guidance and control structure

```
$\hat{\mathbf{x}}_0 \leftarrow \mathbf{x}$                                   (Initialize From Current State)
$\hat{\mathbf{x}}_{k+1} = A_D \hat{\mathbf{x}}_k \quad k = 0, \ldots, N_{drift} - 1$          (Propagate Drift Trajectory)
$\mathbf{x}_{error,k} = \hat{\mathbf{x}}_k - \mathbf{x}_{d,k}, \quad k = 0, \ldots, N_{drift}$          (Calculate Drift Error)
if not control_on and $(\exists \mathbf{x}_{error,k} < \mathbf{x}_{min})$ or $(\exists \mathbf{x}_{error,k} > \mathbf{x}_{max})$, $k = 0, \ldots, N_{drift}$ then
    $control\_on \leftarrow true$
    $\min_{\mathbf{x}_g, \mathbf{v}_{nom}} \frac{1}{2} \sum_{k=0}^{N_g - 1} || \mathbf{v}_{nom,k} ||,$ s.t. $\mathbf{x}_{g,N_g} = \mathbf{x}_{d,N_g}$     (Create Guidance Trajectory)
    $mpc\_step \leftarrow 1$
if $control\_on$ then
    if $mpc\_step > N_g$ then
        $control\_on \leftarrow false$                         (Turn Off Control)
    else
        $\min_{\mathbf{x}_k, \mathbf{v}_k} \frac{1}{2} || \mathbf{v}_0 ||_{R,p} + \frac{1}{2} \sum_{k=1}^{N-1} \left( || \mathbf{x}_{e,k} ||_{Q,p} + || \mathbf{v}_k ||_{R,p} \right)$
            $+ \frac{1}{2} || \mathbf{x}_{e,N} ||_{P,p}$                      (Calculate MPC Control Law)
        $\mathbf{v} = \mathbf{v}_{nom,mpc\_step-1} + \mathbf{v}_0$                      (Get $\Delta V$ For Step)
        $\mathbf{u} = \mathbf{v}/dt$                                 (Convert to Acceleration)
        $\mathbf{u}_{ECI} \leftarrow lvlh2eci(\mathbf{u})$                (Convert Control from LVLH to ECI Frame)
        $mpc\_step = mpc\_step + 1$
else
    $\mathbf{u}_{ECI} = \mathbf{0}_{1 \times 3}$
```

The switching surface could be developed using any number of approaches such as, a six-sided error box [17] in the LVLH position states or an approximation of a sphere[24, 26]. However, the use of ROE state constraints is convenient for two main reasons. First, assuming a non-drifting relative orbit, the desired ROE state and the ROE state constraints are constant. Thus, there is no need to recalculate these bounds at each time step. In contrast, the LVLH states move over time as the spacecraft progresses along its orbit and the state constraints are required to be updated at each time step. Second, the ROE state constraints account for the full state of the spacecraft. State constraints in the LVLH frame account for maximum allowable positions, but generally do not account for the spacecraft's velocity vector. Since the ROE states inherently reflect the dynamics of the system, they provide a more comprehensive boundary between “acceptable" and “unacceptable" orbits.

Attempts were made to mitigate the LVLH velocity limitation in [24, 26] by comparing the spacecraft's predicted propagated trajectory against the switching surface. This allows the incorporation of both the spacecraft velocity and orbital dynamics into the switching strategy as both will affect the propagated trajectory. However, these methods that rely on Cartesian LVLH coordinates can only approximately compare the “closeness" of two orbits. In contrast, the use of ROEs allows for direct measuring of this closeness, which entirely negates the issue.

Since the ROEs do have a time-varying element ($\delta\lambda$), the spacecraft's current state is still propagated forward over some time-horizon so that corrections can be performed well before there is any immediate risk of the spacecraft leaving the allowable boundaries. However, by using ROEs, the full relative orbit can be quantified at each step and compared to the allowed boundaries.

During the drifting phase, the spacecraft state is propagated for N_{drift} steps of the drift horizon using the discrete form of the ROEs dynamics

$$\hat{\mathbf{x}}_{k+1} = \mathbf{A}_D \hat{\mathbf{x}}_k \quad k = 0, \ldots, N_{drift} - 1, \tag{14}$$

where $\hat{\mathbf{x}}_0$ is the current state of the spacecraft. The predicted error state is then compared to the allowable boundaries as $\hat{\mathbf{x}}_k - \mathbf{x}_{d,k} \leq \mathbf{x}_{max}$ or $\hat{\mathbf{x}}_k - \mathbf{x}_{d,k} \geq \mathbf{x}_{min}$, with element-wise comparison. If there is any step k where either of these conditions fail, then a guidance trajectory is calculated and the MPC is activated. Th drift horizon must be sufficiently long that a feasible guidance trajectory can be found which satisfies the constraints. If the horizon is too short then by the time the switching condition triggers, there may not be enough control authority to correct the spacecraft before it moves out of bounds. Furthermore, since a longer drift horizon means the spacecraft is not in any immediate danger of leaving the boundaries, the guidance trajectory has the opportunity to delay corrections until fuel optimal points.

Once the switching condition is activated and a guidance trajectory is found, the focus changes to determining when the spacecraft should cease maneuvering and reenter the drifting state. This is triggered by the spacecraft fully executing the plan associated with the guidance trajectory. Thus, the switching surface activates the guidance and control law when the spacecraft is anticipated to breach the allowable bounds and deactivates it when the spacecraft has finished its planned maneuver.

3.2 Guidance Trajectory

The guidance trajectory is tasked with finding a fuel-optimal trajectory that takes the spacecraft from the current state to the desired orbit within some specified time horizon. When the spacecraft's flight path is predicted to leave the keep-in volume, a single guidance trajectory is generated and then used by the control law.

The trajectory is bounded such that the nominal guidance trajectory, $\mathbf{x}_{nom}$ must be within some tolerance of the desired trajectory $\mathbf{x}_d$. Assuming a time-invariant boundary and a non-drifting desired relative orbit, the boundary constraints do not need to be recalculated every time a guidance trajectory is generated and can be expressed in terms of $\mathbf{x}_{max}$ and $\mathbf{x}_{min}$ This is a distinct contrast to the previous work [24, 26, 27] where the boundaries needed to be recalculated each time a guidance trajectory was found.

The trajectory and nominal control inputs, $\mathbf{v}_{nom}$, over N_g steps of the guidance trajectory are found as

$$\begin{array}{lll}
\min\limits_{\mathbf{x}_{nom},\mathbf{v}_{nom}} & \sum\limits_{k=0}^{N_g-1} |\mathbf{v}_{nom,k}|_1 & \\
\text{s.t.} & \mathbf{x}_{nom,k+1} = \mathbf{A}_D\mathbf{x}_{nom,k} + \mathbf{A}_D\mathbf{B}_k\mathbf{v}_{nom,k} & k = 0, 1, \ldots, N_g - 1 \\
& \mathbf{x}_{nom,N_g} = \mathbf{x}_{d,N_g} & \\
& \mathbf{x}_{nom,k} \leq \mathbf{x}_{max} & k = 1, 2, \ldots, N_g \\
& \mathbf{x}_{nom,k} \geq \mathbf{x}_{min} & k = 1, 2, \ldots, N_g \\
& \mathbf{v}_{nom,k} \leq \mathbf{v}_{max} & k = 0, 1, 2, \ldots, N_g - 1 \\
& \mathbf{v}_{nom,k} \geq \mathbf{v}_{min} & k = 0, 1, 2, \ldots, N_g - 1
\end{array}, \tag{15}$$

where the initial state of the guidance trajectory, $\mathbf{x}_{nom,0}$, is assumed to be the current state of the spacecraft. Note that the control input matrix $\mathbf{B}_k$ is time varying as a function of the mean argument of latitude although the state transition matrix is time invariant.

With this formulation, there is no dependence on any transient state errors ($\mathbf{x}_{e,k}, k = 1, \ldots, N - 1$) except to ensure that the spacecraft does not leave the allowable boundaries. However, there is a terminal constraint, $\mathbf{x}_{nom,N_g} = \mathbf{x}_{d,N_g}$, to ensure that the solution converges to the desired trajectory. This allows the guidance trajectory the option to select the fuel optimal locations for orbital transfer, e.g., at apogee, perigee, or the node line. If controls are applied at points besides these, successful orbital transfers are still possible but at the cost of additional fuel usage. In general, conservation of spacecraft fuel is significantly more important than convergence speed so adding the transient error and finding an appropriate weighting may increase complexity while giving minimal benefit.

The state constraints apply a maximum allowable deviation from the desired trajectory. These constraints are analogous to a high-dimensional box where each state element is constrained, similar to the six-sided error boxes used to constrain position in [17, 18]. Note that the given ROE constraints constrain the full state of the system. This reduces the concern about a spacecraft using large amounts of control if it gets “stuck" in the corner of a box since the MPC constraint is able to account for the full state, both position and velocity, of the system.

3.3 Model Predictive Control Law

The trajectory, $\mathbf{x}_{nom}$, and control profile, $\mathbf{v}_g$, calculated by the guidance trajectory are then passed to the MPC. While these can act as an open-loop control law that drive the spacecraft to the desired trajectory, an MPC is developed that tracks this nominal trajectory and rejects perturbations.

Once the guidance trajectory is found, the MPC tracks which step, n_g, it is on, beginning with $n_g = 0$. The MPC then selects the portion of the trajectory to track $\mathbf{x}_{track}$ over its N_c time horizon steps as

$$\mathbf{x}_{track,k} = \mathbf{x}_{g,n}, \quad n = n_g + k,\ k = 1, \dots, N_c, \tag{16}$$

$$\mathbf{v}_{guide,k} = \mathbf{v}_{nom,n}, \quad n = n_g + k,\ k = 0, \dots, N_c, \tag{17}$$

In the case when $n > N_g$ and $n > N_g - 1$ for the trajectory and control, respectively, these are set as

$$\mathbf{x}_{track,k} = \mathbf{x}_{d,k},\ k > N_g, \tag{18}$$

$$\mathbf{v}_{guide,k} = \mathbf{0}, \tag{19}$$

where $\mathbf{x}_{d,0}$ is the desired state of the spacecraft at the current moment.

The guidance trajectory and associated control inputs define the long-term fuel optimal correction maneuver the spacecraft should follow while the MPC calculates any corrections needed to keep the spacecraft close to the guidance trajectory in the face of perturbations which are not accounted for in the model. Since the MPC is designed to reject short-term perturbations, its time horizon can be significantly shorter than the guidance horizon which allows for decreased computation requirements.

A set of weighting parameters $\mathbf{Q}$, $\mathbf{P}$, and $\mathbf{R}$ are used to weight the MPC response to minimize transient state error, terminal state error, and control usage, respectfully. By weighting the state errors more than the control usage, the MPC tends to be more aggressive in rejecting perturbations at the cost of higher control usage. In contrast, weighting the control usage higher will push the MPC closer to an open-loop configuration where the guidance controls are followed with little change despite any deviations from the guidance trajectory. However, the boundary constraints are still enforced, so the MPC will still correct any deviations that would breach the boundary. The boundary constraints are identical to those used in both the switching surface and the guidance trajectory since the purpose is to maintain a safe spacecraft formation by enforcing the separation of the spacecraft in the formation.

The MPC which chooses the optimal trajectory, $\mathbf{x}_c$, and control perturbations, $\mathbf{v}_c$, can now be formulated as

$$\begin{aligned}
\min_{\mathbf{x}_c,\mathbf{v}_c} \quad & \sum_{k=0}^{N_c-1} |\mathbf{R}\mathbf{v}_{c,k}|_1 + \sum_{k=1}^{N_c-1} |\mathbf{Q}\mathbf{x}_{e,k}|_1 + |\mathbf{P}\mathbf{x}_{e,N}|_1 \\
\text{s.t.} \quad & \mathbf{x}_{c,k+1} = \mathbf{A}_D\mathbf{x}_{c,k} + \mathbf{A}_D\mathbf{B}\mathbf{v}_{c,k} + \mathbf{A}_D\mathbf{B}\mathbf{v}_{guide,k} \qquad k = 0, 1, \dots, N_c - 1 \\
& \mathbf{x}_{e,k} = \mathbf{x}_{nom,k} - \mathbf{x}_{track,k} \qquad k = 1, 2, \dots, N_c \\
& \mathbf{x}_{c,k} \le \mathbf{x}_{max} \qquad k = 1, 2, \dots, N_c \\
& \mathbf{x}_{c,k} \ge \mathbf{x}_{min} \qquad k = 1, 2, \dots, N_c \\
& \mathbf{v}_{c,k} \le \mathbf{v}_{max} - \mathbf{v}_{guide,k} \qquad k = 0, 1, 2, \dots, N_c - 1 \\
& \mathbf{v}_{c,k} \ge \mathbf{v}_{min} - \mathbf{v}_{guide,k} \qquad k = 0, 1, 2, \dots, N_c - 1
\end{aligned}. \tag{20}$$

Once the optimal profiles have been chosen, the final combined control is found as

$$\mathbf{v} = \mathbf{v}_{guide,0} + \mathbf{v}_{c,0}, \tag{21}$$

where $\mathbf{v}$ is an impulsive ΔV since the ROE model used for both the guidance trajectory and MPC use an impulsive ΔV as the inputs. However, it is assumed that the input to the system is acceleration. Over a short time, dt, a constant acceleration, $\mathbf{u}$, that approximates the impulsive $\mathbf{v}$ can be found as

$$\mathbf{u} = \frac{\mathbf{v}}{dt}, \tag{22}$$

which can then be applied to the spacecraft simulation.

Once the guidance and control loop has been entered, the MPC runs once at each time step to find the optimal input. This input is then applied to the system. The resulting spacecraft state is then used to update the initial state of the MPC. This process continues until the final step of the guidance trajectory has been completed at which point the guidance and control loop is exited and the spacecraft returns to its initial drifting state.

4 Numerical Simulation

This section discusses the simulation and parameters used to test the guidance and control law outlined in Section 3 and presents results from this simulation. A flowchart of the framework, for a single spacecraft, is presented in Fig. 1 beginning with the spacecraft obtaining or measuring its own state. If the spacecraft is in the drifting state with no control being applied, then the spacecraft propagates its state to determine if a guidance trajectory is needed. Otherwise, if there is already an active guidance trajectory, the spacecraft calculates the control needed to follow it. The control is then passed to the simulator to be applied to the system in the environmental model.

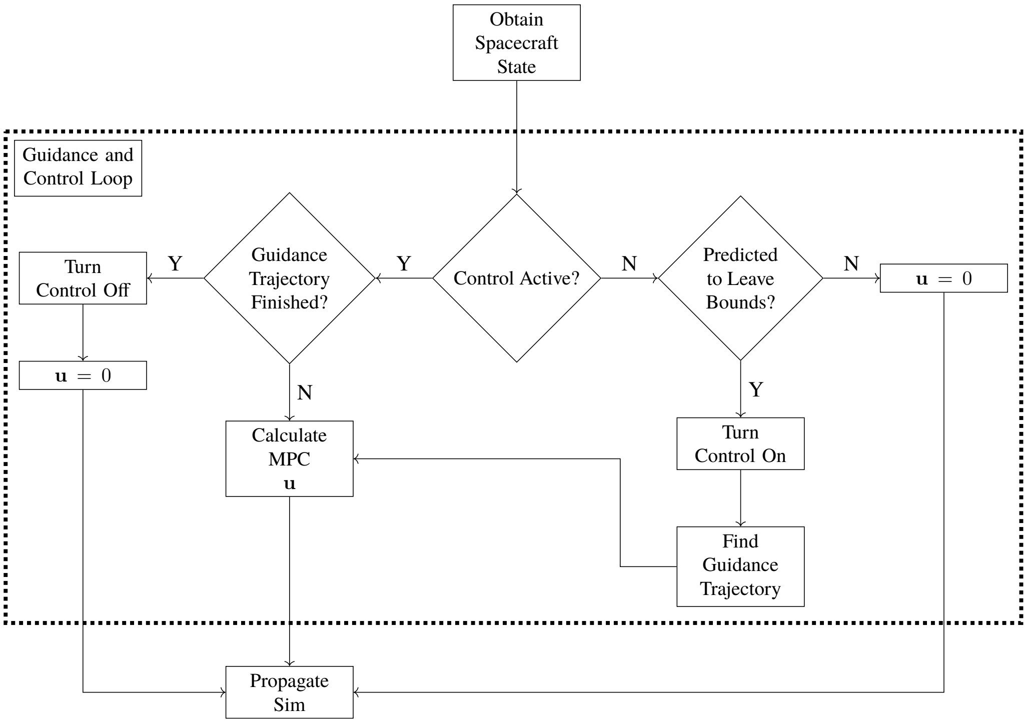


Fig. 1 Flowchart showing the simulation logic structuring and flow for a single spacecraft. The dashed line indicates the guidance and control loop that each spacecraft in the formation runs in parallel.

4.1 Simulation Overview

For each formation, the virtual leader's initial state is defined using Keplerian orbital elements and then converted to a Cartesian Earth Centered Inertial (ECI) state. The actual and desired location of each agent in the formation are initialized using D'Amico's ROEs and then converted to an ECI representation, using the leader as the reference.

To analyze the system, two-body dynamics with nonlinear perturbations in the ECI frame are used as the true states for each spacecraft and the reference orbit. The nonlinear dynamics are applied in the ECI frame as

$$F_d = -\frac{1}{2}\rho C_d A|\mathbf{v}_{rel}|\mathbf{v}_{rel}, \tag{23}$$

$$F_{J_2} = \frac{3}{2}\frac{J_2\mu R_E^2}{||\mathbf{r}_p||^4}\left[\frac{r_x}{||\mathbf{r}_p||}\left(5\frac{r_z^2}{||\mathbf{r}_p||^2}-1\right)\hat{i} + \frac{r_y}{||\mathbf{r}_p||}\left(5\frac{r_z^2}{||\mathbf{r}_p||^2}-1\right)\hat{j} + \frac{r_z}{||\mathbf{r}_p||}\left(5\frac{r_z^2}{||\mathbf{r}_p||^2}-3\right)\hat{k}\right], \tag{24}$$

$$\dot{m} = -\frac{||\mathbf{u}||_1 m}{g_0 I_{sp}}, \tag{25}$$

$$\dot{\mathbf{r}}_p = \mathbf{r}_v, \tag{26}$$

$$\dot{\mathbf{r}}_v = -\mu\frac{\mathbf{r}_p}{||\mathbf{r}_p||^3} + \frac{F_{J_2}}{m} + \frac{F_d}{m} + \mathbf{u}, \tag{27}$$

where F_d is the force due to drag, F_{J_2} is the force due to the J_2 perturbation [33], and $\mathbf{u}$ is the calculated control acceleration in the ECI frame, applied as a zero-order hold. The ECI state of the spacecraft is represented by $\mathbf{r}$ where subscripts indicate the position vector (p), a specific position axis (x, y, or z), or the velocity vector (v). The relative velocity of the spacecraft to the atmosphere is calculated as

$$\mathbf{v}_{rel} = \mathbf{r}_v - \boldsymbol{\omega}_E \times \mathbf{r}_p, \tag{28}$$

where $\boldsymbol{\omega}_E$ is the angular velocity of the Earth. As described in Section 2.1, the MPC calculates an impulsive ΔV maneuver in the LVLH frame, δv_r, δv_t, and δv_n. The LVLH acceleration is approximated as $\mathbf{a}_{LVLH} = \Delta V_{LVLH}/dt$ and then converted to the ECI frame to be applied to the system.

The desired orbits are updated in the ROE frame and the true spacecraft ROEs are found from the ECI states. The simulation then advances one time step and the process is repeated until the simulation duration has been reached.

4.2 Simulation Parameters

The parameters for the simulation and guidance and control algorithms are presented in this section.

4.2.1 Spacecraft Parameters

The simulation and spacecraft parameters used for this work are shown in Table 1. The physical parameters (mass, thrust, specific impulse (I_{sp}), etc.) are based on a 12U CubeSat design built by Space Dynamics Laboratory with a hybrid propulsion system [38]. The spacecraft is assumed to be able to independently apply a $0.5\,\mathrm{N}$ thrust along any of its three orthogonal axes.

The reference orbit is shown in Table 2 and is a nearly circular Low Earth Orbit. The desired orbits for each of the three spacecraft in the formation are shown in Table 3. Each spacecraft is in a non-drifting natural motion circumnavigating (NMC) orbit with the maximum radial and normal components creating a $1\,\mathrm{km} \times 2\,\mathrm{km}$ ellipse. Additionally, each spacecraft is separated from its neighbors by $6\,\mathrm{km}$ with the central spacecraft's NMC centered on the reference orbit position.

The allowable deviations on each spacecraft are shown in Table 4 and each spacecraft is subject to the same constraints. A rough understanding of these constraints in LVLH space can be obtained by

Table 1 Parameters for simulation

Parameter	Value
Spacecraft Wet Mass	$24\,\mathrm{kg}$
Spacecraft C_D	2.2
Spacecraft Area to Mass	$0.002\,\mathrm{m}^2/\mathrm{kg}$
Spacecraft Thrust	$0.5\,\mathrm{N}$
Spacecraft I_{sp}	$160\,\mathrm{s}$

Table 2 Reference Orbit of the virtual leader

Keplerian Orbital Element	Value
Semi-major Axis	$6\,878\,\mathrm{km}$
Eccentricity	1×10^{-4}
Inclination	25°
RAAN	45°
Argument of Perigee	0°
Initial True Anomaly	100°

Table 3 Relative Orbital Elements for the desired agent trajectory

ROE Element	Spacecraft 1	Spacecraft 2	Spacecraft 3
δa	0	0	0
$\delta\lambda$	0	8.723×10^{-4}	-8.723×10^{-3}
δe_x	1.454×10^{-4}	1.454×10^{-4}	1.454×10^{-4}
δe_y	0	0	0
δi_x	1.454×10^{-4}	1.454×10^{-4}	1.454×10^{-4}
δi_y	0	0	0

multiplying the values by the semi-major axis [31]. The δa constraints restricts the radial position error of the center of the spacecraft's relative orbit to around $\pm 100\,\mathrm{m}$. The $\delta\lambda$ term constrains the along track error of the relative orbit's center to $\pm 500\,\mathrm{m}$. The δe_x and δe_y constraints can be combined to form a $\delta e^2 = \delta e_x^2 + \delta e_y^2$ value that gives the maximum radial deviation as $\pm 200\,\mathrm{m}$ and the maximum along track position deviation as $\pm 400\,\mathrm{m}$. The δi value can similarly be found and gives the maximum normal position deviation as $\pm 200\,\mathrm{m}$.

Table 4 Relative Orbital Element Constraints for the desired trajectories

ROE Constraint	Spacecraft 1	Spacecraft 2	Spacecraft 3
δa	$\pm 1.45 \times 10^{-5}$	$\pm 1.45 \times 10^{-5}$	$\pm 1.45 \times 10^{-5}$
$\delta\lambda$	$\pm 7.27 \times 10^{-5}$	$\pm 7.27 \times 10^{-5}$	$\pm 7.27 \times 10^{-5}$
δe_x	$\pm 2.06 \times 10^{-5}$	$\pm 2.06 \times 10^{-5}$	$\pm 2.06 \times 10^{-5}$
δe_y	$\pm 2.06 \times 10^{-5}$	$\pm 2.06 \times 10^{-5}$	$\pm 2.06 \times 10^{-5}$
δi_x	$\pm 2.06 \times 10^{-5}$	$\pm 2.06 \times 10^{-5}$	$\pm 2.06 \times 10^{-5}$
δi_y	$\pm 2.06 \times 10^{-5}$	$\pm 2.06 \times 10^{-5}$	$\pm 2.06 \times 10^{-5}$

There is some interplay between the constraints that can be seen by both the $\delta\lambda$ and δe terms affecting the along track position of the spacecraft. Similarly, the δa and δe constraints both impact the radial spacecraft position. Thus, there is no single set of LVLH distances that describe the constraints on the spacecraft's allowable orbits. However, the above descriptions should give some idea as to the constrained space.

4.2.2 Monte Carlo Parameters

To obtain a better understanding of the efficacy of the system, the guidance and control laws were tested using a Monte Carlo simulation where the initial states of the spacecraft were perturbed but all other parameters, including the desired formation, remained the same. A total of 500 individual simulations were run where the initial ROE state error of each spacecraft selected from a uniform distribution centered on zero with the bounds equal to $\pm 10\%$ of the corresponding constraint. The parameters for the Monte Carlo simulation is shown in Table 5

Once the spacecraft initial states were determined, they were then propagated forward in time until one of the three spacecraft in the formation triggered the guidance and control law. This marked the point where data began being recorded and the start time of the simulation. Each individual simulation then

Table 5 Simulation and Monte Carlo Parameters

Parameter	Value
Simulation Timestep	$10\,\mathrm{s}$
Simulation Duration	$7\,\mathrm{days}$
Monte Carlo Simulations	500
Distribution	$\pm 10\%$

ran for a total of 7 days with a $10\,\mathrm{second}$ time step.

Ignoring any data prior to the first maneuver helps to compare the simulations evenly. Depending on the initialization, each the formation required different lengths of time to drift to the point where corrections are needed. By beginning the data collection only once a correction is required, the guidance and control laws have an impact over the full simulation time, for all simulations.

4.2.3 Guidance and Control Parameters

The parameters used for the guidance and control time horizons are shown in Table 6. The drift horizon approximately covers, a full orbit while the guidance horizon is approximately an orbit and a half in length. The length of the drift horizon is meant to give the spacecraft sufficient warning so that it is able to maneuver at the fuel-optimal points. The length of the guidance horizon defines the maximum convergence time for the spacecraft to return to the desired trajectory. The length (> 1 orbit) is designed so that the optimization has the option of choosing the points along the orbit that are traditional the fuel optimal locations to perform a maneuver (e.g., perigee, apogee, intersection of planes) although there is no explicit requirement constraining maneuvers to these points. The control horizon is shorter and a $5\,\mathrm{minute}$ duration is used since it is only correcting the short term perturbations.

Table 6 Guidance and control parameters used in the simulations

Parameter	Value
Guidance Horizon	820 Steps ($2.28\,\mathrm{hrs}$)
MPC Horizon	30 Steps ($5\,\mathrm{min}$)
Drift Horizon	600 Steps ($1.67\,\mathrm{hrs}$)
Boundary Margin	10%

Initial tests of the guidance and control loop showed that the guidance law generated trajectories that would closely match the allowable bounds. This is most apparent in the relative semi-major axis (δa) since changing this parameter also adjusts the relative argument of latitude ($\delta\lambda$) over time. Thus, maximizing the change δa state allows for faster changes in the $\delta\lambda$ state. However, when a state is close to the allowable boundary, slight perturbations can push them out of bounds which the MPC attempts to correct. This can result in the spacecraft continually oscillating about the allowable boundary. To avoid this, a margin is applied to the boundaries so that they are decreased by 10% for the switching surface, guidance law, and MPC.

Since the guidance trajectory uses a terminal constraint, no tuning is required within the guidance law. For the MPC, the weights used for this work are

$$Q = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \; R = \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & 0.01 \end{bmatrix}, \; S = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \qquad (29)$$

although additional tuning of Q, R, and S may provide lower ΔV results. Since the relative difference in

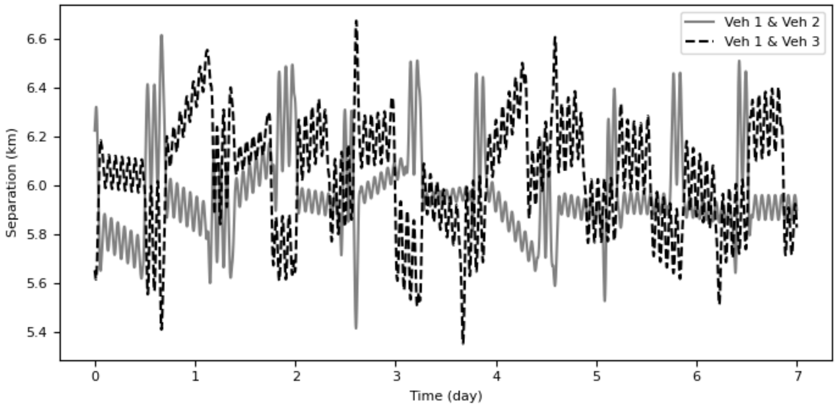


Fig. 2 Spacecraft separation distances using the guidance and control approach presented in the paper.

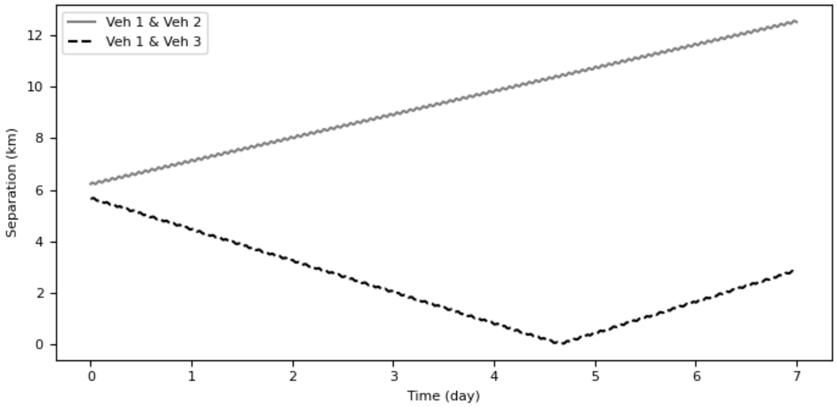


Fig. 3 Spacecraft separation distances when the spacecraft are uncontrolled.

semi-major axis (δa) drives the drift rate and along-track offset of the spacecraft ($\delta\lambda$), the δa element is weighted higher in the Q and S matrices to minimize the drift due to error in the semi-major axis.

4.3 Alternate Approach Comparisons

A sampling of alternative approaches to the guidance and control loop are touched on in this section and used as a point of comparison for the approach presented in Section 3.

4.3.1 Uncontrolled Formation

One approach maybe to simply place the spacecraft into their initial formation and then let them drift with the expectation that each spacecraft in the formation will experience similar perturbations and not experience any relative drift. However, the necessity of correction maneuvers can be seen when comparing Fig. 2 with Fig. 3. In Fig. 2, the distance between the spacecraft is shown where the spacecraft are using the guidance and control approach as presented above. With the formation defined according to Table 3, the desired separation between the first and second spacecraft and the first and third spacecraft is $6\,\mathrm{km}$. The plot shows that the spacecraft maintain this separation within about $600\,\mathrm{m}$. In contrast, Fig. 3 shows the separation between the three spacecraft if they were not being controlled. The separation distance between vehicles 1 and 2 is unbounded in its growth and the two spacecraft are shown to drift apart. Vehicles 1 and 3 drift towards each other. These vehicles experience a minimum separation distance of less than $15\,\mathrm{m}$, at

4.7 days into the simulation, after which they pass each other and start drifting apart. This is a dangerously close approach and represents a high probability of collision for an actual system when accounting for measurement uncertainty in the spacecraft's positions. Note that both simulations have perturbed initial states. The drift in both cases is due to the mismatched orbits as well as environmental perturbations. It is clear from these figures that the guidance and control laws are necessary and successful in maintaining the desired separation between spacecraft.

4.3.2 Open Loop Guidance Law

While the guidance and control law, as described above, is a closed-loop control, it can be run as an open-loop control by applying the controls associated with the guidance trajectory with no MPC corrections. In order to illustrate the differences, a single simulation was run where the two approaches were given initial starting conditions.

In Fig. 4 and 5, the ΔV applied at each time step is shown for the first spacecraft maneuver for the open and closed loop approaches, respectively. In both approaches, the guidance law generates identical controls which can be seen as the three ΔV spikes the two plots have in common. However, the closed-loop formulation has an additional number of small corrections that reject the perturbations that were not accounted for in the guidance law model.

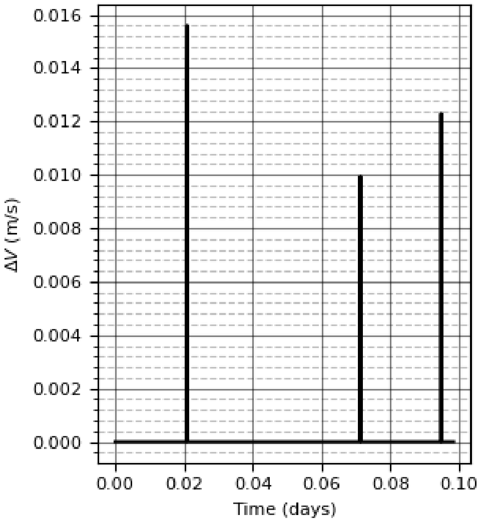


Fig. 4 ΔV using only the guidance portion of the control in an open-loop formulation for the first maneuver.

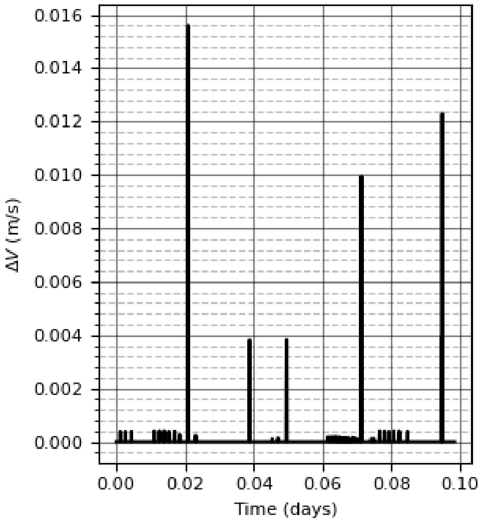


Fig. 5 ΔV usage over the first maneuver using the closed-loop guidance and control laws.

Both approaches saw a total of 12 correction maneuvers used over the 7-day simulation. The ΔV used per maneuver for both approaches is shown in Fig. 6 while the remaining position error after each maneuver finishes is shown in Fig. 7. In Fig. 6 and 7, the middle line is the median value, the box shows the span between the first and third quartile, the lines show the points within 1.5 times that range, and outliers are shown as circles. The open-loop control uses less ΔV with a median value of $0.088\,\mathrm{m/s}$ used over the course of each maneuver compared to the median value of $0.108\,\mathrm{m/s}$ used with the closed-loop control. Similarly, the total ΔV usage over the full 7-day simulation was found to be $1.023\,\mathrm{m/s}$ and 1.279 for the open- and closed-loops, respectively. This is an expected result since the closed-loop control uses additional ΔV to correct perturbations that the open-loop control ignores. The result of these corrections can be seen in Fig. 7 where the final position error of each maneuver is shown. The median terminal error for the open loop is $60.53\,\mathrm{m}$ while the median error for the closed loop is $33.89\,\mathrm{m}$. Thus, there is a trade-off between how much ΔV is used and how closely the spacecraft tracks the desired trajectory. Within the architecture as presented, the control law is closed-loop. However, by adjusting the weightings on the inner-loop controller so that the control usage is weighted significantly more than the state errors, an open-loop like behavior can be obtained. This would limit the extent to which perturbations are rejected but would keep the spacecraft within the allowable boundaries since the boundary constraint would be enforced.

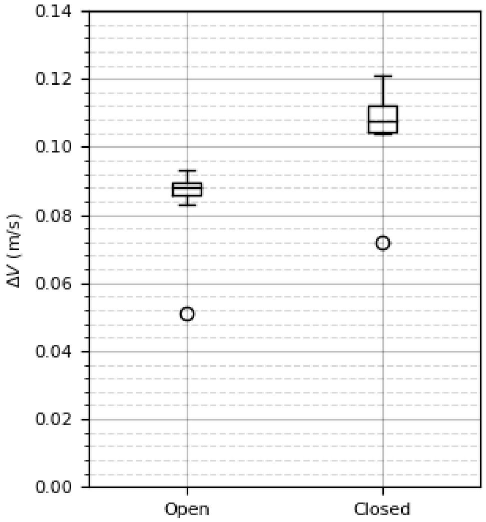


Fig. 6 A comparison of the ΔV used over each maneuver for the open- and closed-loop approaches.

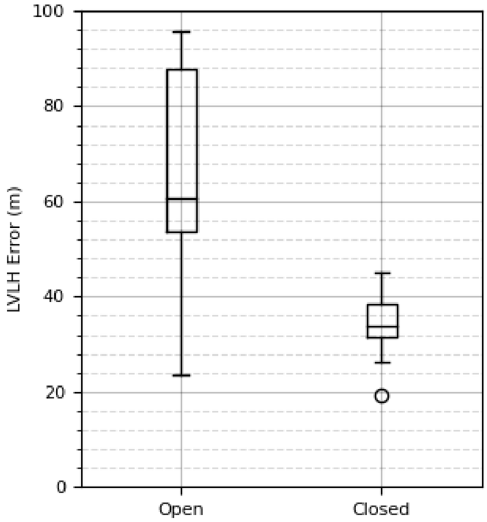


Fig. 7 Terminal position error of each maneuver for open- and closed-loop approaches

4.3.3 Constant Control

A final point of comparison is to have the spacecraft control always on and constantly correcting errors in its trajectory. In this case, as soon as the spacecraft finishes its guidance and control loop it immediately calculates a new guidance trajectory and the cycle continues. Again, a single simulation is used to illustrate the effects such a control approach has.

As seen in Fig. 8, the spacecraft is continually using control over the full simulation. However, in Fig. 9, which is a detailed portion of Fig. 8, the control usage is seen to be composed of individual, discrete impulses rather than the spacecraft using control constantly.

The LVLH position error between the actual and desired trajectories is shown in Fig. 10 for the constantly on control. Once the initial error is corrected, the spacecraft is always within $100\,\mathrm{m}$ of the desired trajectory. This is larger than the distance seen in [26] but this is because, in that case, the controller was trying to minimize the position error while in this case the controller is minimizing the ROE error.

Overall, the spacecraft used a total of $2.81\,\mathrm{m/s}$ of ΔV, which is greater than the control usage of $1.5\,\mathrm{m/s}$ from using the control formulation from this work. Regardless, the constant control usage would not be suitable for missions that require non-maneuvering periods to fulfill their objectives.

4.4 Individual Results

The results of an individual spacecraft are presented in this portion for a detailed explanation of the spacecraft's behavior. A more holistic review of the simulations is presented in the following section

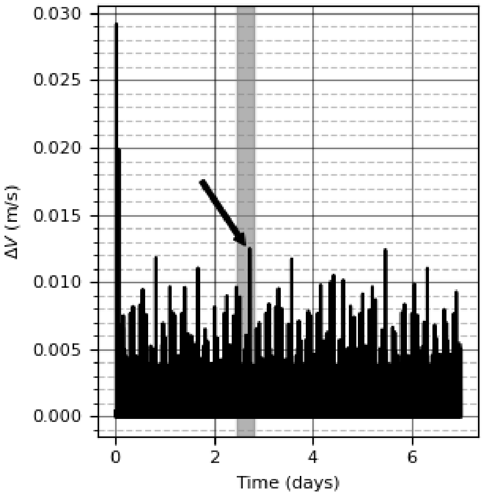


Fig. 8 ΔV usage with the guidance and control loop always on.

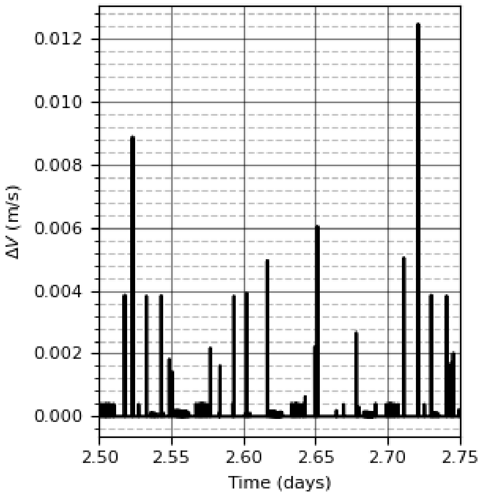


Fig. 9 Zoomed in view of the indicated grey zone in Fig. 8.

where the aggregate results for all simulations are presented. However, examining the results of a single simulation can give valuable insight into its performance.

In Fig. 11, the ΔV usage at each timestep is given over the full simulation. The control usage is divided into distinct periods of maneuvering and drifting, as would be expected from the switching surface. From this plot, we can see that the spacecraft performed a total of 12 maneuvers over the full simulation.

The gray bar near $t = 3\,\mathrm{days}$ indicates one period when the spacecraft was maneuvering back to the desired trajectory. This section of Fig. 11 is shown in detail in Fig. 12. Here, we see that the control is generally divided into distinct impulse-like segments with the largest controls being those planned by the guidance law and the smaller controls being those deemed necessary by the control law to reject perturbations. The exact behavior can be modified by adjusting the weights of the control law to determine the extent that the control attempts to reject perturbations.

In Fig. 13, the ROE errors are shown with the allowable bound depicted with the dashed lines. Additionally, the periods with the guidance trajectory and MPC active are shown by the gray bar. The primary sources of error are seen to be in the $\delta\lambda$ and δe_y components, although the control keeps them within bounds. As could be expected from an examination of the ROE dynamics, the δa component is used to correct the $\delta\lambda$ error. The δe_y error is corrected with the direct application of control. The accumulation of error in the $\delta\lambda$ state can be seen to be due to the small oscillations in the δa term that, due to a slight offset from zero, result in the spacecraft drifting along the $\delta\lambda$ component.

By way of comparison, Fig. 14 shows the LVLH position error over time for the same simulation. From the LVLH error, two observations can be made. First, there is no consistent LVLH error that triggers the guidance and control algorithm. While the maximum error across each maneuver and drift cycle roughly falls between 500 and 600 meters there is no hard bound in the LVLH error. Second, the LVLH error is not monotonically decreasing during the periods when the guidance and control is active. This results from the fact that the guidance trajectory waits for the fuel optimal points rather than immediately attempt to minimize errors. While both of these are understandable when viewed from an orbital dynamics perspective, they demonstrate the somewhat counter-intuitive nature of attempting to understand the ROE guidance and control laws in an LVLH frame.

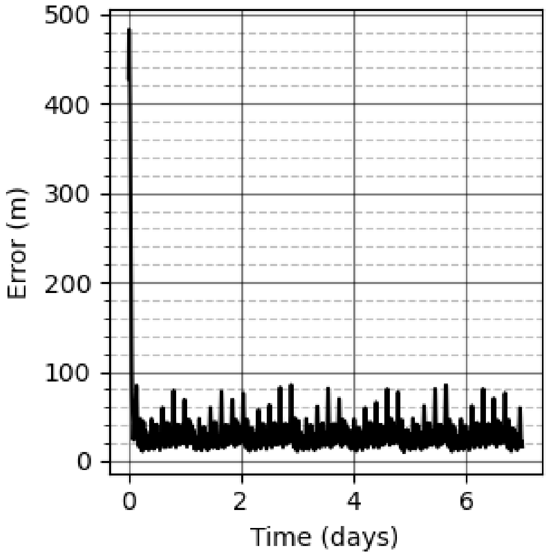


Fig. 10 LVLH position error with the guidance and control loop always on.

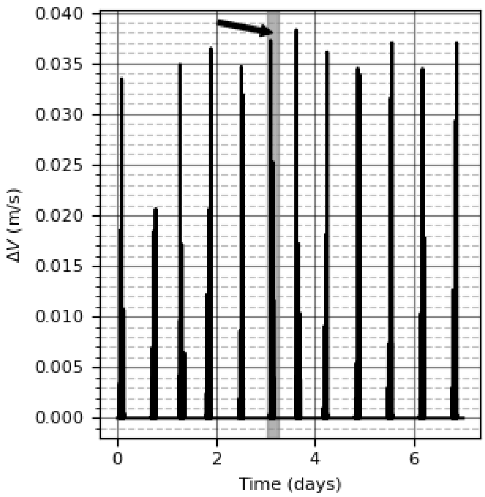


Fig. 11 ΔV usage for the spacecraft. The single maneuver indicated is detailed in Fig. 12

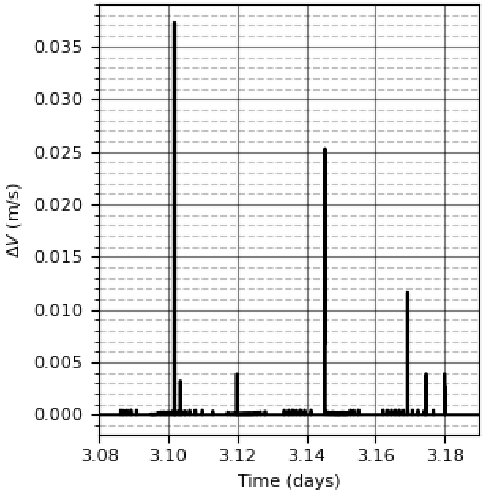


Fig. 12 Zoomed in view, grey zone, of Fig. 11.

4.5 Cumulative Results

This section provides the cumulative results that summarize the performance of the ROE based framework outlined in the previous sections. The results in this section represent Monte Carlo analysis of 500 simulations.

Examining the computation time required, the guidance and control law averaged 3.246 ± 0.168 s to run the initial step of each maneuver where the guidance trajectory is calculated with a maximum run time of 4.457 s. On the subsequent steps of the guidance and control law, where only the control law is run, the mean computation time was found to be 0.009 ± 0.001 s with a maximum computation time of

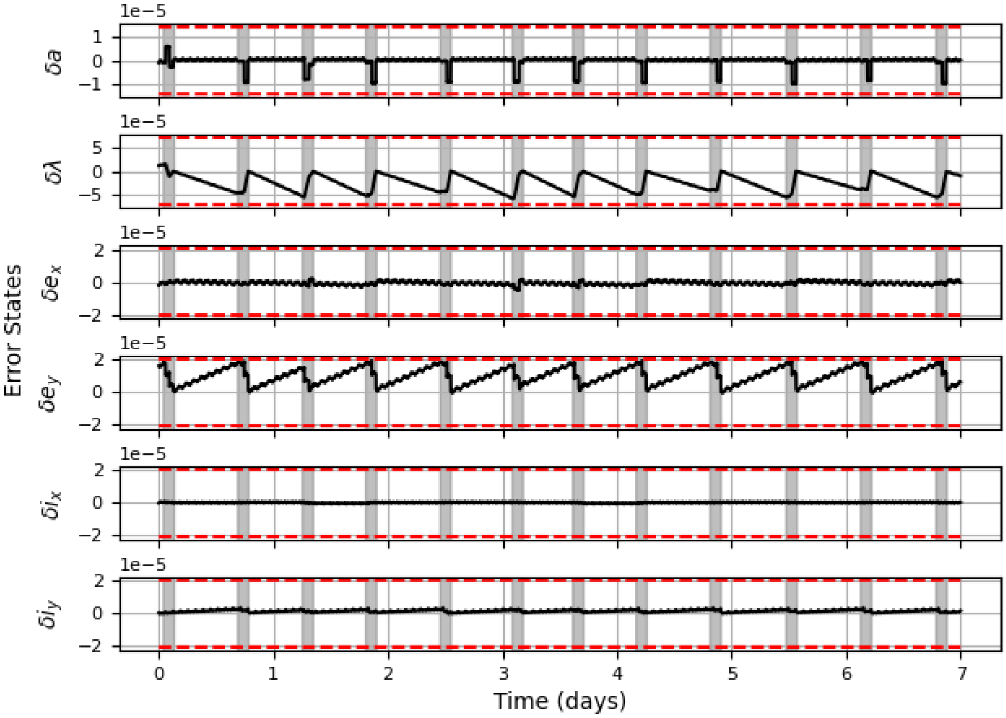


Fig. 13 ROE error over the simulation with the allowable bounds marked with the dashed lines. Gray zones indicate when the spacecraft guidance and control loop was active.

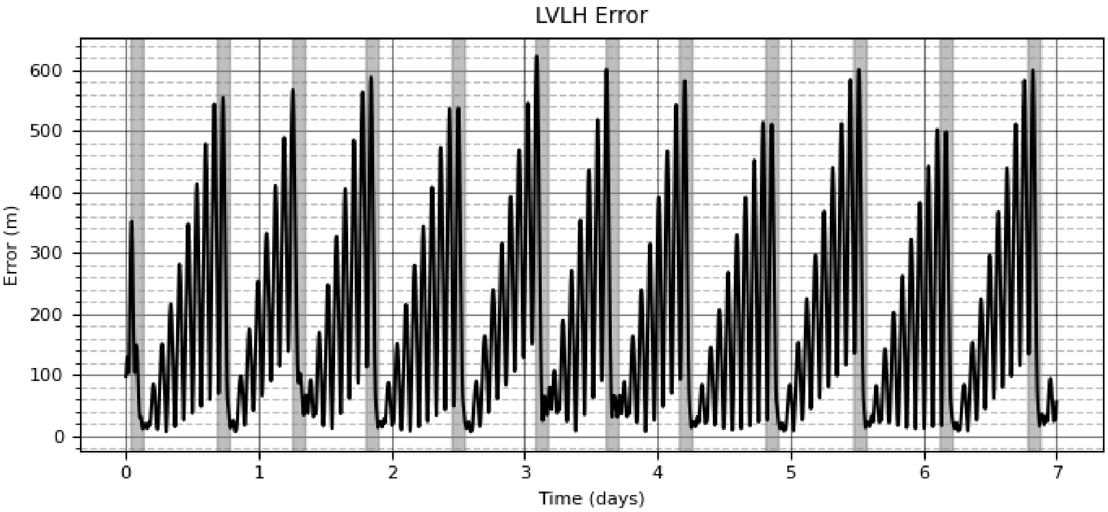


Fig. 14 LVLH error over the simulation. Gray zones indicate when the spacecraft guidance and control loop was active.

$1.775\,\mathrm{s}$ observed. These were all run using the Gurobi [39] solver.

In Fig. 15, the average number of maneuvers used by each vehicle is shown. A maneuver is defined to be the period that the control and guidance loop is active, regardless of whether the spacecraft is commanding a non-zero control input at every time during that period. The average number of maneuvers per vehicle were found to be 11.37 ± 0.48, 11.91 ± 0.29, and 12.10 ± 0.37, for the first, second, and third vehicles, respectively. This means that the spacecraft are maneuvering roughly 1.7 times per day and suggests that there is ample opportunity for the spacecraft to perform its primary mission during the drifting periods, note that each maneuver time is ~ 2.3 hours. However, the number of maneuvers used over any given period of time is a function of the allowable error on each spacecraft's trajectory since a larger keep-in volume allows the spacecraft to drift longer before needed a correction maneuver.

Fig. 16 shows the mean ΔV used during each error correction maneuver while Fig. 17 shows the total ΔV used over the full simulation. The per maneuver ΔV shows the spacecraft using 0.112 ± 0.012 m/s, 0.113 ± 0.014 m/s and 0.124 ± 0.014 m/s for each trajectory correction for the first, second, and third spacecraft, respectively. Additionally, the total ΔV for each vehicle was found to be 1.272 ± 0.033 m/s, 1.349 ± 0.058 m/s, and 1.497 ± 0.042 m/s for the first, second, and third vehicle, respectively. This indicates that it would take between 68 and 78 m/s to maintain this formation for a year-long mission. These numbers are well within the capability of a small satellite, which can hold between 100-200 m/s, depending on available volume [38].

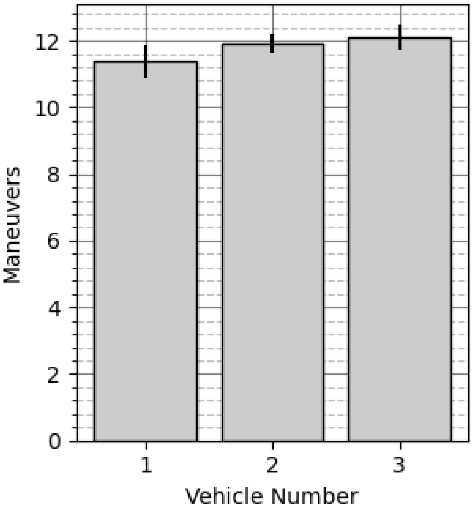


Fig. 15 Total number of maneuvers over the course of the simulation with the line indicating the $1-\sigma$ standard deviation.

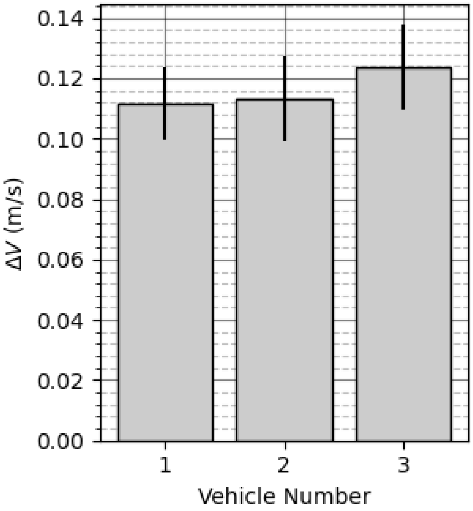


Fig. 16 Total ΔV used per maneuver for the three different desired orbits

4.6 HCW Comparison

The control architecture in this paper was also compared to the previous work done in [26] which used a similar approach based on the HCW equations. The two approaches attempt to mirror each other with the only difference being the model used for relative motion and boundary dynamics. For both approaches, the randomization of the initial states are the same, the same norm definition is used, and the formation is identical. However, there are some key differences that cannot be reconciled. With regard to the keep-in boundaries, a direct conversion between a spherical boundary and the ROE defined boundaries is not possible. However, the HCW model uses a boundary with radius $500\,\mathrm{m}$, which roughly correlates to the ROE boundaries shown in Table 4 and is demonstrated by the allowable LVLH errors seen in Fig.

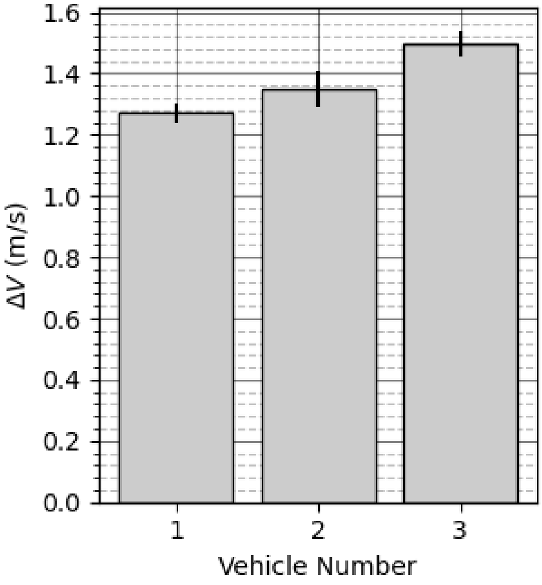


Fig. 17 Total ΔV used over the course of the simulation.

Table 7 Average number of maneuvers per vehicle over the 7-day simulations using ROEs and HCW algorithms.

Vehicle	ROE	HCW
1	11.37 ± 0.48	16.55 ± 2.19
2	11.91 ± 0.29	20.28 ± 1.32
3	12.10 ± 0.37	20.18 ± 2.24

Table 8 Average ΔV per maneuver for each vehicle over the 7-day simulations using ROEs and HCW algorithms.

Vehicle	ROE	HCW
1	0.112 ± 0.012 m/s	0.245 ± 0.069 m/s
2	0.113 ± 0.014 m/s	0.237 ± 0.042 m/s
3	0.124 ± 0.014 m/s	0.247 ± 0.070 m/s

14. Additionally, the weights on the MPC do not directly correlate due to the difference in error state dimensions, but the authors believe that the weights used for both formulations represent a near optimal result. The authors of this paper believe that while a perfect comparison of the two approaches may not be possible, the comparison presented in this section gives good insight into the performance of the two approaches.

The computation time difference between the two approaches is negligible. In Tables 7, 8, and 9, the number of maneuvers, ΔV per maneuver, and total ΔV are compared for the two approaches. For each metric, the ROE approach shows more favorable behavior. In Table 7, the HCW approach has roughly 1.5 times more maneuvers than the corresponding vehicle using the ROE approach. Since each maneuver is designed with an identical, fixed convergence duration, a vehicle using the ROE approach is able to devote a greater amount of its time to the actual mission. Moreover, the vehicles with the ROE approach use roughly half the amount of ΔV during each of their maneuvers (Table 8). Combining these two effects, the ROE approach shows a significant improvement in the total ΔV used over the course of the full seven day simulations (Table 9) and uses approximately a third of the ΔV used by the HCW approach.

Table 9 Average ΔV comparison for 7-day simulations using ROEs and HCW algorithms.

Vehicle	ROE	HCW
1	1.272 ± 0.033 m/s	4.062 ± 0.223 m/s
2	1.349 ± 0.058 m/s	4.815 ± 0.201 m/s
3	1.497 ± 0.042 m/s	4.994 ± 0.273 m/s

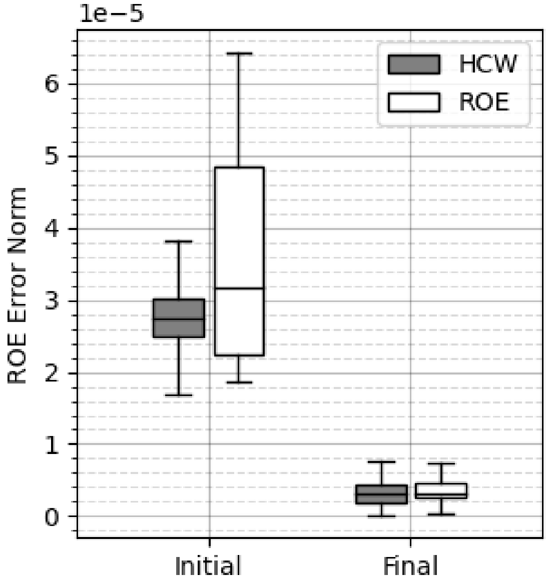


Fig. 18 2-norm of the ROE element-wise error for both approaches comparing the pre- and post-maneuver states. Outliers are removed for clarity.

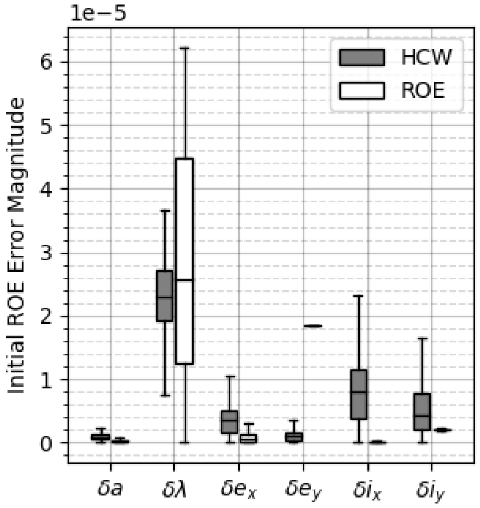


Fig. 19 ROE error magnitude distribution for both the HCW and ROE approaches at the end of maneuvers. Outliers are removed from the plot for clarity.

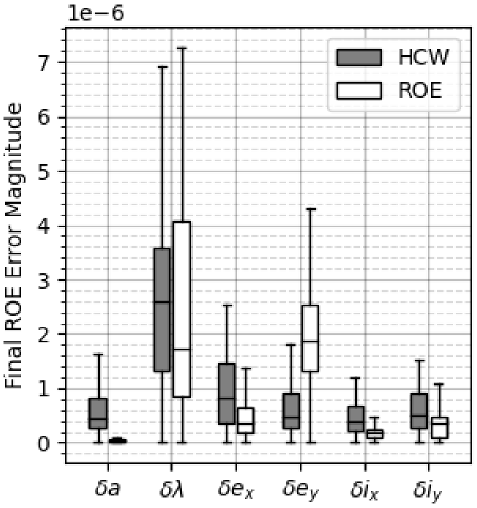


Fig. 20 ROE error magnitude distribution for both the HCW and ROE approaches at the beginning of maneuvers. Outliers are removed from the plot for clarity.

Figure 18 shows the 2-norm error of the different approaches in ROE state space immediately before and after maneuvers. In general, the errors are comparable for both approaches with the initial error being around 3×10^{-5}, although the distribution of initial error for the ROE model is larger than for the HCW model. The final errors are likewise similar although roughly an order of magnitude smaller at 3×10^{-6}. This shows that the ROE based approach is able to handle a larger range of errors than the HCW based approach, but uses significantly less fuel.

In Figures 19 and 20, the error magnitudes for the pre- and post-maneuver states, respectively, are shown. Of particular note is that the error is primarily dominated by the $\delta\lambda$ term, which is analogous to the along-track error. There is also noticeably higher error in the ROE model in the δe_y term although there is no obvious cause.

Overall though, the ROE approach presented in this paper gives comparable, or better, results than the HCW comparison. It is able to match the terminal error of the HCW approach despite a larger range of initial errors and notably less control usage.

5 Conclusions

This work presents a guidance and control architecture that uses the D'Amico ROEs to constrain a spacecraft to a given orbit relative to a reference. When the spacecraft predicts it will leave the allowable ROE constraints it enters a guidance and control loop and calculates the fuel-optimal trajectory that will return it to the desired orbit within some given time horizon. The spacecraft then uses a model predictive control to repeatedly solve for solutions, over a shorter horizon, that will balance deviations from the guidance trajectory with total control usage. Once the spacecraft has fully executed the guidance trajectory, it returns to a drifting state until it again determines if it will drift beyond the allowable bounds and the process repeats.

As presented in this paper, this framework requires less overall ΔV when compared to a continuous controlled system. This approach also reduces the terminal error when compared to an open-loop system. This work also shows that if formation control is not performed, the formation is not maintained, and potential collision events may occur.

The ROE based model predictive control framework presented as part of this work was analyzed through Monte Carlo simulation with a total of 500 1-week simulations with formations of three vehicles. The simulations showed that the spacecraft were able to stay within the allowable constraints without excessive propellant usage. The simulations also showed that the ROE formulation developed as part of this work requires fewer maneuvers and 33% the ΔV of previously published architectures based on the HCW equations.

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

References

[1] Bristow J, Folta D, Hartman K. A Formation Flying Technology Vision. *Space 2000 Conference and Exhibition*, 2000, doi:10.2514/6.2000-5194.

[2] Scharf DP, Hadaegh FY, Ploen SR. A Survey of Spacecraft Formation Flying Guidance and Control (Part II): Control. *Proceedings of the American Control Conference*, 2004, 4(Part 11): 2976–2985, doi:10.23919/acc.2004.1384365.

[3] Di Mauro G, Lawn M, Bevilacqua R. Survey on Guidance Navigation and Control Requirements for Spacecraft Formation-Flying Missions. *Journal of Guidance, Control, and Dynamics*, 2018, 41(3): 581–602, doi:10.2514/1.G002868.

[4] D'Amico S, Ardaens JS, De Florio S. Autonomous Formation Flying Based on GPS - PRISMA Flight Results. *Acta Astronautica*, 2013, 82(1): 69–79, doi:10.1016/j.actaastro.2012.04.033.

[5] Roth NH, Risi B, Grant CC, Zee RE. Flight Results from the CanX-4 and CanX-5 Formation Flying Mission. In *29th Annual AIAA/USU Conference on Small Satellites*, SSC15-I-4, 2016, 1–15, doi:10.1016/j.jcrysgro.2014.02.053.

[6] Bertiger W, Bar-Sever Y, Bettadpur S, Dunn C. GRACE: Millimeters and Microns in Orbit. In *ION GPS 2002 Conference*, September, 2002, 2022–2029, doi:10.1.1.475.4059.

[7] Tapley BD, Bettadpur S, Watkins M, Reigber C. The Gravity Recovery and Climate Experiment: Mission Overview and Early Results. *Geophysical Research Letters*, 2004, 31(9): 1–4, doi:10.1029/2004GL019920.

[8] Krieger G, Moreira A, Fiedler H, Hajnsek I, Werner M, Younis M, Zink M. TanDEM-X: A Satellite Formation for High-Resolution SAR Interferometry. *IEEE Transactions on Geoscience and Remote Sensing*, 2007, 45(11): 3317–3340, doi:10.1109/TGRS.2007.900693.

[9] Fiedler H, Krieger G, Werner M, Reiniger K. The TanDEM-X Mission Design and Data Acquisition Plan. In *European Conference on Synthetic Apeture Radar 2006*, May, 2006, 16–18.

[10] Farahmand M, Long A, Hollister J, Rose J, Godine D. Magnetospheric Multiscale Mission Navigation Performance During Apogee-Raising and Beyond. *Advances in the Astronautical Sciences*, 2018, 162: 2738–2739.

[11] Winternitz LB, Bamford WA, Price SR, Carpenter JR, Long AC, Farahmand M. Global Positioning System Navigation Above 76,000 KM for NASA'S Magnetospheric Multiscale Mission. *Journal of the Institute of Navigation*, 2017, 64(2): 289–300, doi:10.1002/navi.198.

[12] Mann A. Starlink: SpaceX's Satellite Internet Project. *https://www.space.com/spacex-starlink-satellites.html*, 2020.

[13] Engberg B, Twiggs R. Operations Planning and Formation Flying : Analyzing Resource Usage in Formation Assembly. *Annual AIAA/USU Conference on Small Satellites*, 1999, (SSC99–VIII–2).

[14] Rocco EM, De Oliveira E Souza ML, De Almeida Prado AFB. Station Keeping of Constellations Using Multi-Objective Strategies. *Mathematical Problems in Engineering*, 2013, 2(1), doi:10.1155/2013/476451.

[15] Breger LS. Model Predictive Control for Formation Flying Spacecraft. Master's thesis, Massachusetts Institute of Technology, 2004.

[16] Lim Y, Jung Y, Bang H. Robust Model Predictive Control for Satellite Formation Keeping with Eccentricity/Inclination Vector Separation. *Advances in Space Research*, 2018, 61(10): 2661–2672, doi:10.1016/j.asr.2018.02.036.

[17] Tillerson M, How JP. Advanced Guidance Algorithms for Spacecraft Formation-Keeping. *Proceedings of the American Control Conference*, 2002, 4: 2830–2835, doi:10.1109/ACC.2002.1025218.

[18] Di Mauro G, Spiller D, Rafano Carnà SF, Bevilacqua R. Minimum-Fuel Control Strategy for Spacecraft Formation Reconfiguration via Finite-Time Maneuvers. *Journal of Guidance, Control, and Dynamics*, 2019, 42(4): 752–768, doi:10.2514/1.G003822.

[19] Rawlings JB, Mayne DQ, Diehl MM. *Model Predictive Control: Theory, Computation, and Design*. 2nd edition, Santa Barbara, California: Nob Hill Publishing, 2019, doi:10.1155/2012/240898.

[20] Di Cairano S, Park H, Kolmanovsky I. Model Predictive Control approach for guidance of spacecraft rendezvous and proximity maneuvering. *International Journal of Robust and Nonlinear Control*, 2012, 22(August): 1293–1427, doi:10.1002/rnc.

[21] Weiss A, Baldwin M, Erwin RS, Kolmanovsky I. Model predictive control for spacecraft rendezvous and docking: Strategies for handling constraints and case studies. *IEEE Transactions on Control Systems Technology*, 2015, 23(4): 1638–1647, doi:10.1109/TCST.2014.2379639.

[22] Buckner C, Lampariello R. Tube-Based Model Predictive Control for the Approach Maneuver of a Spacecraft to a Free-Tumbling Target Satellite. *Proceedings of the American Control Conference*, 2018, 2018-June: 5690–5697, doi:10.23919/ACC.2018.8431558.

[23] Dong K, Luo J, Dang Z, Wei L. Tube-based robust output feedback model predictive control for autonomous rendezvous and docking with a tumbling target. *Advances in Space Research*, 2020, 65(4): 1158–1181, doi:10.1016/j.asr.2019.11.014.

[24] Smith TK, Akagi J, Droge G. Model Predictive Control Switching Strategy For Safe Small Satellite Cluster Formation Flying. *Journal of Aerospace Systems*, 2023, doi:10.1007/s42401-023-00237-2.

[25] Dong K, Luo J, Limon D. A novel stable and safe model predictive control framework for autonomous

rendezvous and docking with a tumbling target. *Acta Astronautica*, 2022, 200(March): 176–187, doi:10.1016/j.actaastro.2022.08.012.

[26] Smith T, Akagi J, Droge G. Satellite Cluster Flight Using Guidance Trajectory and Model Predictive Control. *Journal of the Astronautical Sciences*, Under Review.

[27] Smith T, Akagi J, Droge G. Spacecraft Formation Flying Control Switching Surface Based on Relative Orbital Elements. *2022 AAS/AIAA Astrodynamics Specialist Conference*, 2022: 1–19.

[28] Hill GW. On the Part of the Motion of Lunar Perigee Which is a Function of the Mean Motions of the Sun and Moon. *Acta Math*, 1886, 8: 1–36, doi:10.1007/BF02417081.

[29] Wiltshire RS, Clohessy WH. Terminal Guidance System for Satellite Rendezvous. *Journal of The Aerospace Sciences*, 1960, 27(9): 663–674, doi:10.2514/8.8704.

[30] Lovell TA, Spencer DA. Relative orbital elements formulation based upon the clohessy-wiltshire equations. *Journal of the Astronautical Sciences*, 2014, 61(4): 341–366, doi:10.1007/s40295-014-0029-6.

[31] D'Amico S. Autonomous Formation Flying in Low Earth Orbit. Ph.D. thesis, Delft University of Technology, 2010.

[32] D'Amico S, Montenbruck O. Proximity operations of formation-flying spacecraft using an eccentricity/inclination vector separation. *Journal of Guidance, Control, and Dynamics*, 2006, 29: 554–563, doi:10.2514/1.15114.

[33] Curtis H. *Orbital Mechanics for Engineering Students*. 2nd edition, Oxford, United Kingdom: Elsevier, 2009.

[34] Borrelli F, Bemporad A, Morari M. *Predictive Control for Linear and Hybrid Systems*, Cambridge, United Kingdom: Cambridge University Press, 2017, doi:10.1017/9781139061759.

[35] Liberzon D, Morse AS. Basic problems in stability and design of switched systems. *IEEE control systems magazine*, 1999, 19(5): 59–70.

[36] Shorten R, Wirth F, Mason O, Wulff K, King C. Stability Criteria for Switched and Hybrid Systems. *SIAM Review*, 2007, 49(4): 545–592, doi:10.1137/05063516X.

[37] Mazal L, Gurfil P. Cluster Flight Algorithms for Disaggregated Satellites. *Journal of Guidance, Control, and Dynamics*, 2013, 36(1): 124–135, doi:10.2514/1.57180.

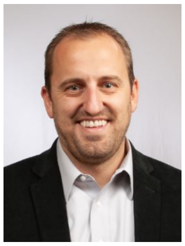
[38] Smith TK, Lewis Z, Olsen K, Bulcher MA, Whitmore SA. A Miniaturized, Green, End-Burning, and Sandwich Hybrid Propulsion System. *Journal of Propulsion and Power*, 2022, 38(5): 701–713, doi:10.2514/1.B38623.

[39] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2023.

Author biography

Tyson Smith is an aerospace engineer at the Space Dynamics Lab, specializing in guidance, navigation, and controls. Tyson is also a PhD student in the Electrical Engineering Department at Utah State University. tyson.smith@sdl.usu.edu

John Akagi is a PhD student in the Mechanical and Aerospace Engineering Department at Utah State University studying nonlinear spacecraft controls. He also works at the Space Dynamics Laboratory as part of the Guidance, Navigation, and Controls team. john.akagi@sdl.usu.edu



Greg Droge is an assistant professor in the Electrical and Computer Engineering Department at Utah State University specializing in control systems. greg.droge@usu.edu

CHAPTER 7
CONCLUSION

The current command and control of satellite constellations largely consists of developing control schedules for a single satellite at a time by a mission planning team working on the ground. Future desires to have hundreds and even thousands of satellites in a constellation make the single-satellite planning approach infeasible. Satellites that have the ability to autonomously formation fly, can make this future desire a reality. But certain challenges exist when trying to solve the problem of autonomous formation flight, they include; fuel consumption, overcoming disturbances or perturbing forces, data distribution between spacecraft, collision avoidance, mission operational constraints, and computation feasibility. This research addresses these challenges through providing a decentralized, optimization-based control that can be used for autonomous formation flight of small satellites. Through developing a switched, model predictive control framework based on a virtual structure architecture. This architecture uses both the HCW equations and the D'Amico ROEs to describe the relative motion of the system. This framework allows for minimal communication from operators and between satellites while conserving fuel consumption. It also enables required operational drift while staying within safe regions of operation for coordinated flight

The major contributions of this research are; the development of a virtual structure, model predictive control architecture based on the Hill-Clohessey-Wiltshire equations control. Development of a virtual structure, model predictive control architecture based on the D'Amico relative orbital elements. Development of switching control strategies based on a spherical polytope approximation that is centered on each satellite, and a switching strategy that is based on the D'Amico ROEs. These architectures were analysed and compared and contrasted to identity advantages and disadvantages regarding fuel usage and control computation time.

This framework requires less overall ΔV when compared to a continuous controlled system and an LQR framework. This approach also reduces the terminal error when compared to an open-loop system. This work shows that if formation control is not performed, the formation is not maintained, and potential collision events may occur.

This research presents and compares the formulation of using L_1, L_2, and L_∞ norm objective functions for the MPC. The simulations show that the control formulation is able to maintain its position within the polytope volume with reasonable computation time requirements for all three norm formations. As would be expected, the computation time for solving the MPC problem increase with the desired time horizon. However, the L_2 norm showed the slowest computation time growth while the L_1 and L_∞ norms grew at similar rates. In the simulations, all the norms were able to run at near real-time rates or better despite not optimizing for computation time. This research also shows that the L_1 norm produces the lowers total ΔV, over a seven day period, with the L_2 norm requiring a meter per second more than the L_1, and the L_∞ norm being a full two meters per second higher than the L_1 norm.

Comparing the number of times that each spacecraft was required to maneuver showed roughly equivalent performance between the three different norms. However, the L_1 norm required the lowest number of manuvevers, followed by the L_2 norm and then the L_∞ norm. This research also shows that the ROE formulation developed as part of this work requires 1/3 the ΔV of architecture based on the HCW equations.

REFERENCES

[1] A. Manno, *Intercorrelated Satellite Observations Related to Solar Events.* Astrophysics and Space Science Library, 1969.

[2] A. Labeyrie, "Stellar Interferometry Methods," *Annual Review of Astronomy and Astrophysics*, vol. 16, no. 1, pp. 77–102, 1978.

[3] G. B. Sholomitsky, O. F. Prilutsky, and V. G. Rodin, "Infrared space interferometer," *28th Int. Astro. Fed. Congress Praha Czechoslovakia*, 1977.

[4] R. V. Stachnik and D. Y. Gezari, "SAMSI: an orbiting spatial interferometer for micro-arcsecond astronomical observations." *Kilometric Optical Arrays in Space*, vol. 226, no. May 1985, pp. 35–42, 1985. [Online]. Available: https://ui.adsabs.harvard.edu/{#}abs/1985ESASP.226...35S/abstract

[5] A. Labeyrie, G. Schumacher, and E. Savaria, "Flute or trio: Different approaches to optical arrays in space," *Advances in Space Research*, vol. 2, no. 4, pp. 11–22, 1982.

[6] A. B. DeCou, "Multiple spacecraft optical interferometry preliminary feasibility assessment," *JPL Tech. Report D-8811*, 1991.

[7] M. M and S. Leszkiewicz, "Efficient spacecraft formation keeping with consideration of ballistic coefficient control," *AIAA Aerospace Science*, 1988.

[8] C. Scolese, D. Folta, and F. Bordi, "Field of view location and formation flying for polar orbiting missions," *AAS/AIAA Spaceflight Mechanics Meeting*, pp. 949–966, 1991.

[9] P. K. Wang and F. Y. Hadaegh, "Coordination and Control of Multiple Microspacecraft Moving in Formation," *Journal of the Astronautical Sciences*, vol. 44, no. 3, pp. 315–355, 1996.

[10] J. Bristow, D. Folta, and K. Hartman, "A formation flying technology vision," *Space 2000 Conference and Exhibition*, September 2000.

[11] D. P. Scharf, F. Y. Hadaegh, and S. R. Ploen, "A Survey of Spacecraft Formation Flying Guidance and Control (Part II): Control," *Proceedings of the American Control Conference*, vol. 4, no. Part 11, pp. 2976–2985, 2004.

[12] S. D'Amico, J. S. Ardaens, and S. De Florio, "Autonomous Formation Flying Based on GPS - PRISMA Flight Results," *Acta Astronautica*, vol. 82, no. 1, pp. 69–79, 2013.

[13] N. H. Roth, B. Risi, C. C. Grant, and R. E. Zee, "Flight Results from the CanX-4 and CanX-5 Formation Flying Mission," in *29th Annual AIAA/USU Conference on Small Satellites*, no. SSC15-I-4, 2016, pp. 1–15.

[14] N. J. Cole and S. R. Talbot, "Guidance , Navigation , and Control for Commercial and Scientific Applications of Formation Flying," *32nd Annual AIAA/USU Conference on Small Satellites*, 2018. [Online]. Available: https://digitalcommons.usu.edu/smallsat/2018/all2018/305

[15] W. Bertiger, Y. Bar-Sever, S. Bettadpur, and C. Dunn, "GRACE: Millimeters and Microns in Orbit," in *ION GPS 2002 Conference*, no. September, 2002, pp. 2022–2029.

[16] B. D. Tapley, S. Bettadpur, M. Watkins, and C. Reigber, "The Gravity Recovery and Climate Experiment: Mission Overview and Early Results," *Geophysical Research Letters*, vol. 31, no. 9, pp. 1–4, 2004.

[17] G. Krieger, A. Moreira, H. Fiedler, I. Hajnsek, M. Werner, M. Younis, and M. Zink, "TanDEM-X: A Satellite Formation for High-Resolution SAR Interferometry," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 11, pp. 3317–3340, 2007.

[18] H. Fiedler, G. Krieger, M. Werner, and K. Reiniger, "The TanDEM-X Mission Design and Data Acquisition Plan," in *European Conference on Synthetic Apeture Radar 2006*, no. May, 2006, pp. 16–18.

[19] M. Farahmand, A. Long, J. Hollister, J. Rose, and D. Godine, "Magnetospheric Multiscale Mission Navigation Performance During Apogee-Raising and Beyond," *Advances in the Astronautical Sciences*, vol. 162, pp. 2738–2739, 2018.

[20] L. B. Winternitz, W. A. Bamford, S. R. Price, J. R. Carpenter, A. C. Long, and M. Farahmand, "Global Positioning System Navigation Above 76,000 KM for NASA'S Magnetospheric Multiscale Mission," *Journal of the Institute of Navigation*, vol. 64, no. 2, pp. 289–300, 2017.

[21] S. Bandyopadhyay, G. P. Subramanian, R. Foust, D. Morgan, S. J. Chung, and F. Y. Hadaegh, "A review of impending small satellite formation flying missions," *53rd AIAA Aerospace Sciences Meeting*, no. January, pp. 1–17, 2015.

[22] E. M. Rocco, M. L. De Oliveira E Souza, and A. F. B. De Almeida Prado, "Station Keeping of Constellations Using Multi-Objective Strategies," *Mathematical Problems in Engineering*, vol. 2, no. 1, 2013.

[23] L. S. Breger, "Model Predictive Control for Formation Flying Spacecraft," Master's thesis, Massachusetts Institute of Technology, 2004.

[24] Y. Lim, Y. Jung, and H. Bang, "Robust Model Predictive Control for Satellite Formation Keeping with Eccentricity/Inclination Vector Separation," *Advances in Space Research*, vol. 61, no. 10, pp. 2661–2672, 2018.

[25] M. Tillerson and J. P. How, "Advanced Guidance Algorithms for Spacecraft Formation-Keeping," *Proceedings of the American Control Conference*, vol. 4, pp. 2830–2835, 2002.

[26] G. Di Mauro, D. Spiller, S. F. Rafano Carnà, and R. Bevilacqua, "Minimum-Fuel Control Strategy for Spacecraft Formation Reconfiguration via Finite-Time Maneuvers," *Journal of Guidance, Control, and Dynamics*, vol. 42, no. 4, pp. 752–768, 2019.

[27] J. B. Rawlings, D. Q. Mayne, and M. M. Diehl, *Model Predictive Control: Theory, Computation, and Design*, 2nd ed. Santa Barbara, California: Nob Hill Publishing, 2019.

[28] S. Di Cairano, H. Park, and I. Kolmanovsky, “Model Predictive Control approach for guidance of spacecraft rendezvous and proximity maneuvering,” *International Journal of Robust and Nonlinear Control*, vol. 22, no. August, pp. 1293–1427, 2012. [Online]. Available: http://onlinelibrary.wiley.com/doi/10.1002/rnc.1553/abstract

[29] A. Weiss, M. Baldwin, R. S. Erwin, and I. Kolmanovsky, “Model predictive control for spacecraft rendezvous and docking: Strategies for handling constraints and case studies,” *IEEE Transactions on Control Systems Technology*, vol. 23, no. 4, pp. 1638–1647, 2015.

[30] C. Buckner and R. Lampariello, “Tube-Based Model Predictive Control for the Approach Maneuver of a Spacecraft to a Free-Tumbling Target Satellite,” *Proceedings of the American Control Conference*, vol. 2018-June, pp. 5690–5697, 2018.

[31] K. Dong, J. Luo, Z. Dang, and L. Wei, “Tube-based robust output feedback model predictive control for autonomous rendezvous and docking with a tumbling target,” *Advances in Space Research*, vol. 65, no. 4, pp. 1158–1181, 2020. [Online]. Available: https://doi.org/10.1016/j.asr.2019.11.014

[32] K. Dong, J. Luo, and D. Limon, “A novel stable and safe model predictive control framework for autonomous rendezvous and docking with a tumbling target,” *Acta Astronautica*, vol. 200, no. March, pp. 176–187, 2022. [Online]. Available: https://doi.org/10.1016/j.actaastro.2022.08.012

[33] C. V. M. Fridlund, “Darwin-The Infrared Space Interferometry Mission,” in *ESA Bulletin 103*, no. august, 2000, pp. 20–63.

[34] M. Mesbahi and F. Y. Hadaegh, “Formation Flying Control of Multiple Spacecraft via Graphs Matrix Inequalities and Switching,” *Journal of Guidance Control and Dynamics*, vol. 24, no. 2, pp. 369–377, 2001.

[35] R. L. Ticker and J. D. Azzolini, “2000 Survey of Distributed Spacecraft Technologies and Architectures for NASA’s Earth Science Enterprise in the 2010-2025 Time Frame,” NASA, Tech. Rep. 209964, 2000.

[36] M. Martin, P. Klupar, S. Kilberg, and J. Winter, “TechSat 21 and Revolutionizing Space Missions using Microsatellites,” *USU/AIAA Small Satellite Conference*, August 2001.

[37] A. Mann, “Starlink: Spacex’s satellite internet project,” *https://www.space.com/spacex-starlink-satellites.html*, January 2020.

[38] B. Engberg and R. Twiggs, “Operations Planning and Formation Flying : Analyzing Resource Usage in Formation Assembly,” *Annual AIAA/USU Conference on Small Satellites*, no. SSC99–VIII–2, 1999.

[39] A. C. Boley and M. Byers, “Satellite mega-constellations create risks in Low Earth Orbit, the atmosphere and on Earth,” *Scientific Reports*, vol. 11, no. 1, pp. 1–8, 2021. [Online]. Available: https://doi.org/10.1038/s41598-021-89909-7

[40] H. Schaub, S. R. Vadali, J. L. Junkins, and K. T. Alfriend, “Spacecraft formation flying control using mean orbit elements,” *Journal of the Astronautical Sciences*, vol. 48, no. 1, pp. 69–87, 2001.

[41] W. Kang, A. Sparks, and S. Banda, “Coordinated control of multisatellite systems,” *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 2, pp. 360–368, 2001.

[42] H. Curtis, *Orbital Mechanics for Engineering Students*, 2nd ed. Oxford, United Kingdom: Elsevier, 2009.

[43] V. Manikonda, P. O. Arambel, M. Gopinathan, R. K. Mehra, and F. Y. Hadaegh, “Model Predictive Control-Based Approach for Spacecraft Formation Keeping and Attitude Control,” *Proceedings of the American Control Conference*, vol. 6, pp. 4258–4262, June 1999.

[44] M. Mesbahi and F. Y. Hadaegh, “Formation flying control of multiple spacecraft via graphs, matrix inequalities, and switching,” *IEEE Conference on Control Applications - Proceedings*, vol. 2, no. May 2014, pp. 1211–1216, 1999.

[45] ——, “A Robust Control Approach for the Formation Flying of Multiple Spacecraft,” *European Control Conference, ECC 1999 - Conference Proceedings*, pp. 2933–2938, 1999.

[46] J. P. Desai, J. Ostrowski, and V. Kumar, “Controlling Formations of Multiple Mobile Robots,” *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 4, no. May, pp. 2864–2869, 1998.

[47] N. Deo, *Graph Theory with Applications to Engineering and Computer Science.* Prentice-Hall, 1974.

[48] P. K. C. Wang, “Navigation strategies for multiple autonomous mobile robots moving in formation,” *Journal of Robotic Systems*, vol. 8, no. 2, pp. 177–195, 1991.

[49] P. K. Wang, F. Y. Hadaegh, and K. Lau, “Synchronized Formation Rotation and Attitude Control of Multiple Free-Flying Spacecraft,” *1997 Guidance, Navigation, and Control Conference*, no. October 2014, pp. 1582–1589, 1997.

[50] F. Y. Hadaegh and W.-m. Lu, “ADAPTIVE CONTROL OF FORMATION FLYING SPACECRAFT FOR INTERFEROMETRY Fred,” *Jet Propulsion*, 1998.

[51] K. Yamanaka, “Simultaneous translation and rotation control law for formation flying satellites,” *AIAA Guidance, Navigation, and Control Conference*, no. 2000-4440, 2000.

[52] P. K. Wang, F. Y. Hadaegh, and K. Lau, “Synchronized Formation Rotation and Attitude Control of Multiple Free-Flying Spacecraft,” *Journal of Guidance, Control, and Dynamics*, vol. 22, no. 1, pp. 28–35, 1999.

[53] F. Hadaegh and B. Kang, “Rule-based estimation and control of formation flying spacecraft,” 2001. [Online]. Available: http://trs-new.jpl.nasa.gov/dspace/handle/2014/13348

[54] B. J. Lurie, “Multi-Mode Synchronized Control for Formation Flying Interferometer,” in *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2003.

[55] A. Pant, P. Seiler, T. J. Koo, and K. Hedrick, “Mesh stability of unmanned aerial vehicle clusters,” *Proceedings of the American Control Conference*, vol. 1, pp. 62–68, 2001.

[56] G. Q. Xing, S. A. Parvez, and D. Folta, “Design and implementation of synchronized autonomous orbit and attitude control for multiple spacecraft formation using gps measurement feedback,” *AAS/AIAA Spacefight Mechanics Magizine*, pp. 115–134, 2000.

[57] A. Robertson, G. Inalhan, and J. P. How, “Formation control strategies for a separated spacecraft interferometer,” *Proceedings of the American Control Conference*, vol. 6, no. July, pp. 4142–4147, 1999.

[58] R. P. Aguilera, P. Acuna, G. Konstantinou, S. Vazquez, and J. I. Leon, “Chapter 2 - basic control principles in power electronics: Analog and digital control design,” in *Control of Power Electronic Converters and Systems*, F. Blaabjerg, Ed. Academic Press, 2018, pp. 31–68.

[59] R. Olfati-Saber and R. M. Murray, “Distributed structural stabilization and tracking for formations of dynamic multi-agents,” *Proceedings of the IEEE Conference on Decision and Control*, vol. 1, pp. 209–215, 2002.

[60] J. Lawton, R. W. Beard, and F. Y. Hadaegh, “Elementary attitude formation maneuvers via leader-following and behavior-based control,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, no. February 2000, 2000, pp. 1–14.

[61] A. R. Riah, “Formation Control of Multi-Robot using Virtual Structures with a Linear Algebra Approach,” *JAREE (Journal on Advanced Research in Electrical Engineering)*, vol. 4, no. 1, 2020.

[62] D. Zhou, Z. Wang, and M. Schwager, “Agile Coordination and Assistive Collision Avoidance for Quadrotor Swarms Using Virtual Structures,” *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 916–923, 2018.

[63] Y. Liu, J. Gao, C. Liu, F. Zhao, and J. Zhao, “Reconfigurable formation control of multi-agents using virtual linkage approach,” *Applied Sciences (Switzerland)*, vol. 8, no. 7, 2018.

[64] Y. Xu, P. Liu, X. Zhang, C. Zha, and Z. Tian, “Formation Control and Obstacle Avoidance for Multi-agent Systems in Unknown Environment,” *Proceedings of the 2019 IEEE International Conference on Unmanned Systems, ICUS 2019*, pp. 925–930, 2019.

[65] L. Juan, Z. Xu, Z. Honghan, and D. Xue, "Trajectory Tracking Control of Multi-AUVs Formation based on Virtual Leader," *Proceedings of 2019 IEEE International Conference on Mechatronics and Automation, ICMA 2019*, pp. 291–296, 2019.

[66] M. A. Lewis and K.-h. Tan, "High Precision Formation Control of Mobile Robots Using Virtual Structures," *Autonomous Robots*, vol. 403, pp. 387–403, 1997.

[67] R. W. Beard and F. Y. Hadaegh, "Constellation Templates : An Approach To Autonomous Formation Flying," in *World Automation Congress*, December 1997.

[68] ——, "Constellation templates: An approach to autonomous formation flying," *World Automation Congress*, 1998.

[69] ——, "Fuel Optimized Rotation for Satellite Formations in Free Space." in *American Control Conference*, October 1999.

[70] R. W. Beard, J. Lawton, and F. Y. Hadaegh, "A Coordination Architecture for Spacecraft Formation Control," *IEEE Transactions on Control Systems Technology*, vol. 9, no. 6, pp. 777–790, 2001.

[71] M. Tillerson, L. Breger, and J. P. How, "Distributed Coordination and Control of Formation Flying Spacecraft," *Proceedings of the American Control Conference*, vol. 2, pp. 1740–1745, 2003.

[72] W. Ren and R. W. Beard, "Decentralized Scheme for Spacecraft Formation Flying via the Virtual Structure Approach," *Journal of Guidance Control and Dynamics*, vol. 27, no. 1, pp. 73–82, 2004.

[73] B. J. Young, "Mobile Robots: Coordination and Control," Ph.D. dissertation, Brigham Young University, 2000.

[74] R. W. Beard, "A feedback architecture for formation control," *American Control Conference*, October 1999.

[75] G. W. Hill, "On the part of the motion of lunar perigee which is a function of the mean motions of the sun and moon," *Acta Math*, vol. 8, pp. 1–36, 1886.

[76] R. S. Wiltshire and W. H. Clohessy, "Terminal guidance system for satellite rendezvous," *Journal of The Aerospace Sciences*, vol. 27, no. 9, pp. 663–674, 1960.

[77] S. D'Amico, "Autonomous Formation Flying in Low Earth Orbit," Ph.D. dissertation, Delft University of Technology, 2010.

[78] T. Balch and R. C. Arkin, "Behavior-based formation control for multirobot teams," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 6, pp. 926–939, 1998.

[79] L. Mazal and P. Gurfil, "Cluster Flight Algorithms for Disaggregated Satellites," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 1, pp. 124–135, 2013.

[80] J. P. How and M. Tillerson, "Analysis of the impact of sensor noise on formation flying control," *Proceedings of the American Control Conference*, vol. 5, no. February 2001, pp. 3986–3991, 2001.

[81] G. Inalhan, M. Tillerson, and J. P. How, "Relative dynamics and control of spacecraft formations in eccentric orbits," *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 1, pp. 48–59, 2002.

[82] F. Bauer, J. Bristow, D. Folta, K. Hartman, D. Quinn, and J. How, "Satellite formation flying using an innovative autonomous control system (autocon) environment," in *AIAA Guidance, Navigation, and Control Conference*, 1997, pp. 657–666.

[83] D. Folta, L. Newman, and T. Gardner, "Foundations of formation flying for mission to planet earth and new millennium," *AIMAAS Astrodynamics Conference*, pp. 656–666, 1996.

[84] D. Folta and D. Quinn, "A universal 3-d method for controlling the relative motion of multiple spacecraft in any orbit," in *AIMAAS Astrodynamcis. Specialist Conference*, 1998.

[85] D. C. Folta and D. Quinn, "A 3-D method for autonomously controlling multiple spacecraft orbits," *IEEE Aerospace Conference Proceedings*, vol. 1, no. February 2014, pp. 61–68, 1998.

[86] B. Cheng and Z. Li, "Fully Distributed Event-Triggered Protocols for Linear Multi-Agent Networks," *IEEE Transactions on Automatic Control*, vol. 64, no. 4, pp. 1655–1662, 2019.

[87] B. Cheng, Z. Wu, and Z. Li, "Distributed Edge-Based Event-Triggered Formation Control," *IEEE Transactions on Cybernetics*, pp. 1–12, 2019.

[88] Q. Hu and Y. Shi, "Event-Based Coordinated Control of Spacecraft Formation Flying Under Limited Communication," *Nonlinear Dynamics*, vol. 99, no. 3, pp. 2139–2159, 2020.

[89] T. A. Lovell and D. A. Spencer, "Relative orbital elements formulation based upon the clohessy-wiltshire equations," *Journal of the Astronautical Sciences*, vol. 61, no. 4, pp. 341–366, 2014.

[90] S. D'Amico and O. Montenbruck, "Proximity operations of formation-flying spacecraft using an eccentricity/inclination vector separation," *Journal of Guidance, Control, and Dynamics*, vol. 29, pp. 554–563, 2006.

[91] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*. Cambridge, United Kingdom: Cambridge University Press, May 2017.

[92] T. Sugar and V. Kumar, "Decentralized control of cooperating mobile manipulators," in *IEEE International Conference on Robotics and Automation*, May 1998, pp. 2916–2921.

[93] D. Swaroop and J. K. Hedrick, "String stability of interconnected systems," *IEEE Transactions on Automatic Control*, vol. 41, no. 3, pp. 349–357, 1996.

[94] D. Swaroop, "A note about the stability of a string of LTI systems," *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, vol. 124, no. 3, pp. 472–475, 2002.

[95] P. Seiler, A. Pant, and J. K. Hedrick, "Preliminary investigation of mesh stability for linear systems," *In American Society of Mechanical Engineers, Dynamic Systems and Control*, vol. 67, pp. 359–364, 1999.

[96] J. K. Hedrick, I. M. Tomizuka, and P. Varaiya, "Control Issues in Automated Highway Systems," *IEEE Control Systems*, vol. 14, no. 6, pp. 21–32, 1994.

[97] A. Pant, P. Seiler, and K. Hedrick, "Mesh stability of look-ahead interconnected systems," *IEEE Transactions on Automatic Control*, vol. 47, no. 2, pp. 403–407, 2002.

[98] M. Anderson and A. Robbins, "Formation flight as a cooperative game," *Guidance, Navigation, and Control Conference and Exhibit.*, 1998.

[99] K. Sugihara and I. Suzuki, "Distributed Algorithms for Formation of Geometric Patterns with Many Mobile Robots," *Journal of Robotic Systems*, vol. 13, no. 3, pp. 127–139, 1996.

[100] N. E. Leonard and E. Fiorelli, "Virtual Leaders, Artificial Potentials and Coordinated Control of Groups," in *IEEE Conference Decision and Control*, 2001, pp. 2968–2973.

[101] X. Yun, G. Alptekin, and O. Albayrak, "Line and circle formation of distributed physical mobile robots," *Journal of Robotic Systems*, vol. 14, no. 2, pp. 63–76, 1997.

[102] Z. Wang, F. Khorrami, and W. Grossman, "Robust Adaptive Control of Formation Keeping for a Pair of Satellites," *Proceedings of the American Control Conference*, vol. 2, no. June 2000, pp. 834–838, 2000.

[103] C. R. McInnes, "Autonomous ring formation for a planar constellation of satellites," *Journal of Guidance, Control, and Dynamics*, vol. 18, no. 5, pp. 1215–1217, 1995.

[104] G. B. Palmerini, "Guidance strategies for satellite formations," *Advances in the Astronautical Sciences*, vol. 103, no. PART 1, pp. 135–145, 2000.

[105] H. Yamaguchi and J. W. Burdick, "Asymptotic stabilization of multiple nonholonomic mobile robots forming group formations," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 4, no. May, pp. 3573–3580, 1998.

[106] B. J. Young, R. W. Beard, and J. M. Kelsey, "A control scheme for improving multivehicle formation maneuvers," *Proceedings of the American Control Conference*, vol. 2, no. February 2001, pp. 704–709, 2001.

[107] O. Albayrak, "Line and Circle Formation of Distributed Autonomous Mobile Robots with Limited Sensor Range," Ph.D. dissertation, Naval Postgraduate School, 1996.

[108] Q. Chen and J. Y. Luh, "Coordination and control of a group of small mobile robots," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 2315–2320, 1994.

[109] H. Zhang and P. Gurfil, "Cooperative orbital control of multiple satellites via consensus," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 5, pp. 2171–2188, 2018.

[110] J. R. Lawton, R. W. Beard, and B. J. Young, "A decentralized approach to formation maneuvers," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 6, pp. 933–941, 2003.

[111] J. R. Lawton and R. W. Beard, "Synchronized multiple spacecraft rotations," *Automatica*, vol. 38, no. 8, pp. 1359–1364, 2002.

[112] J. Lawton, "A Behavior-Based Approach to Multiple Spaceraft Formation Flying," *PhD Disertation*, 2000.

[113] F. Y. Hadaegh, A. R. Ghavimi, S. Gurkirpal, and M. Quadrelli, "A Centralized Optimal Controller for Formation Flying Spacecraft," in *International Conference on Robotic Intelligence Technology*, 2000, pp. 1–6.

[114] R. S. Smith and F. Y. Hadaegh, "Control Topologies for Deep Space Formation Flying Spacecraft," in *American Control Conference*, 2002, pp. 2836–2841.

[115] J. L. Speyer, "Computation and Transmission Requirements for a Decentralized Linear-Quadratic-Gaussian Control Problem," *IEEE Transactions on Automatic Control*, vol. AC-24, no. 2, pp. 266–269, 1979.

[116] D. Folta and R. Carpenter, "Formation Flying with Decentralized Control in Libration Point Orbits," *NASA Final Report*, no. January, pp. 1–19, 2000.

[117] Y. Ulybyshev, "Long-term formation keeping of satellite constellation using linear-quadratic controller," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 1, pp. 109–115, 1998.

[118] R. Olfati-Saber and R. M. Murray, "Graph rigidity and distributed formation stabilization of multi-vehicle systems," *Proceedings of the IEEE Conference on Decision and Control*, vol. 3, no. December, pp. 2965–2971, 2002.

[119] ——, *Distributed cooperative control of multiple vehicle formations using structural potential functions.* IFAC, 2002, vol. 15.

[120] A. Lamy and S. Pascal, "Station Keeping Strategies for Constellations of Satellites," *AAS/NASA International Symposium on Spaceflight Dynamics*, pp. 819–833, 1993.

[121] E. Lansard, E. Frayssinhes, and J. L. Palmade, "Global design of satellite constellations: A multi-criteria performance comparison of classical walker patterns and new design patterns," *Acta Astronautica*, vol. 42, no. 9, pp. 555–564, 1998.

[122] F. Graziani, G. Palmerini, and P. Teofilatto, "Design and control strategies for global coverage constellations," *Journal of Brazilian SOC. Mech. Sci.*, pp. 181–187, 1994.

[123] J. Lawton, R. W. Beard, and F. Y. Hadaegh, "Adaptive control approach to satellite formation flying with relative distance constraints," *Proceedings of the American Control Conference*, vol. 3, no. February, pp. 1545–1549, 1999.

[124] S. Badesha, G. Heyler, P. Sharer, and T. Strikwerda, “Development of formation deployment and initialization concepts,” *Flight Mechanics*, pp. 333–343, 1999.

[125] B. Shahbazi, M. Malekzadeh, and H. R. Koofigar, “Robust Constrained Attitude Control of Spacecraft Formation Flying in the Presence of Disturbances,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 5, pp. 2534–2543, 2017.

[126] S. A. Schweighart and R. J. Sedwick, “Development and analysis of a high fidelity linearized J2 model for satellite formation flying,” *AIAA Space 2001 Conference and Exposition*, pp. 1–15, 2001.

[127] A. Sparks, “Linear control of spacecraft formation flying,” in *AIAA Guidance, Navigation, and Control Conference*, Denver, CO, 2000.

[128] L. Zhao, J. Yu, and P. Shi, “Command Filtered Backstepping-Based Attitude Containment Control for Spacecraft Formation,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–10, 2019.

[129] L. C. L., H. W. M., and B. E. V., “Orbital formation keeping with differential drag,” *Journal of Guidance Control Dynamics*, vol. 12, no. 1, pp. 108–113, Jan–Feb 1989.

[130] V. Kapila, A. G. Sparks, J. M. Buffington, and Q. Yanl, “Spacecraft Format ion Flying: Dynamics and Control,” in *American Control Conference*, June 1999, pp. 4137–4141.

[131] Q. Yan, G. Yang, V. Kapila, and M. S. de Queiroz, “Nonlinear dynamics and output feedback control of multiple spacecraft in elliptical orbits,” *Proceedings of the American Control Conference*, vol. 2, no. June, pp. 839–843, 2000.

[132] R. Vassar and R. Sherwood, “Formation keeping for a pair of satellites in a circular orbit,” *Journal of Guidance, Control, and Dynamics,*, vol. 8, no. 2, pp. 235–242, 1985.

[133] C. Chao and H. Bemstein, “Onboard stationkeeping of geosynchronous satellites using a global positioning system receiver,” *Journal of Guidance, Control, and Dynamics*, vol. 17, no. 4, pp. 778–786, 1994.

[134] S. R. Starin, R. K. Yedavalli, and A. G. Sparks, “Spacecraft formation flying maneuvers using linearquadratic regulation with no radial axis inputs,” *AIAA Guidance, Navigation, and Control Conference and Exhibit*, no. April 2014, 2001.

[135] G. Q. Xing, S. A. Parvez, and D. C. Folta, “Implementation of Autonomous GPS Guidance and Control for the Spacecraft Formation Flying,” in *American Control Conference*, June 1999, pp. 4163–4167.

[136] J. R. Carpenter, “A preliminary investigation of decentralized control for satellite formations,” *IEEE Aerospace Conference Proceedings*, vol. 7, pp. 63–74, 2000.

[137] M. S. De Queiroz, V. Kapila, and Q. Yan, “Adaptive nonlinear control of multiple spacecraft formation flying,” in *Journal of Guidance, Control, and Dynamics*, vol. 23, 2000, pp. 385–390.

[138] L. Mazal and P. Gurfil, “Closed-Loop Distance-Keeping for Long-Term Satellite Cluster Flight,” *Acta Astronautica*, vol. 94, no. 1, pp. 73–82, 2014.

[139] L. Mazal, D. Pérez, R. Bevilacqua, and F. Curti, “Spacecraft Rendezvous by Differential Drag Under Uncertainties,” *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 8, pp. 1721–1733, 2016.

[140] F. Y. Hadaegh, S. J. Chung, and H. M. Manohara, “On development of 100-gram-class spacecraft for swarm applications,” *IEEE Systems Journal*, vol. 10, no. 2, pp. 673–684, 2016.

[141] S. H. and A. K., “Impulsive spacecraft formation flying control to establish specific mean orbit elements,” *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 4, pp. 739–745, 2001.

[142] Z. Tan, P. M. Bainunr, and A. Strong, “Maintaining constant distance between satellites in coplanar elliptic orbits,” *Journal of the Astronautical Sciences*, vol. 50, no. 1, pp. 53–69, 2002.

[143] H. Schaub and K. T. Alfriend, “Hybrid Cartesian and orbit element feedback law for formation flying spacecraft,” *Astrodynamics Specialist Conference*, vol. 25, no. 2, pp. 184–191, 2000.

[144] L. Breger and J. P. How, “Gauss’s Variational Equation-Based Dynamics andControl for Formation Flying Spacecraft,” in *2nd International Symposium on Formation Flying Missions and Technologies*, vol. 2, 2007, pp. 1–12. [Online]. Available: http://acl.mit.edu/papers/BregerHow{_}Paper.pdf

[145] M. K. Ben Larbi and E. Stoll, “SPACECRAFT FORMATION CONTROL USING ANALYTICAL INTEGRATION OF GAUSS ’ VARIATIONAL EQUATIONS,” *reserchgate.net*, 2009.

[146] M. Leomanni, E. Rogers, and S. B. Gabriel, “Explicit model predictive control approach for low-thrust spacecraft proximity operations,” *Journal of Guidance, Control, and Dynamics*, vol. 37, no. 6, pp. 1780–1790, 2014.

[147] D. A. Spencer, “Automated Trajectory Control for Proximity Operations Using Relative Orbital Elements,” Ph.D. dissertation, Georgia Institute of Technology, 2015.

APPENDICES

APPENDIX A

Extended Formation Flying Literature Survey

At the onset of this research a literature survey of the different formation flying control architectures was conducted. This Appendix presents the additional information of that survey not found in Chapter 2.

A.1 Formation Flying Control Architecture

In the literature their are five basic formation architectures; Leader/Follower, Cyclic, Behavioral, Multiple-Input Multiple-Output (MIMO), and Virtual Structures. Additional literature, not discussed in Chapter 2 is now presented in this section, including details about architecture stability.

Leader/Follower

Other studies, not mentioned in Chapter 2, that implemented a Leader/Follower architecture in the mobile robotics community are; Sugar and Kumar used a Leader/Follower architecture control a group of robots to cooperatively move a box [92]. Desai used feedback linaerization techniques to derive tracking control laws for non-holonomic robots [46].

A Discussion on Leader/Follower Stability: Recall that a Leader/Follower system is built off of a hierarchically connected system, which is considered to be "mesh stable" if it is asymptotically Lyapunov stable and if any perturbations to the leader do not grow as they filter threw each of the followers. In other words, the peak tracking errors of all the followers are uniformly bounded by the peak tracker error of the first follower. Mesh stability is derived from string stability. Lyapunov stability is considered to be less restrictive then both mesh and string stability.

Let us consider a dynamic system that has the form shown in Equation A.1. where x_i is a state of the ith subsystem, and the state is given by Equation A.2. For a Leader/Follower system x_i represents the tracking error of the follower.

$$\dot{x}_i = f_i\left(x_i, x_{i-1}, ..., x_1\right) \tag{A.1}$$

$$x = \begin{bmatrix} {x_1}^T \\ {x_2}^T \\ {x_3}^T \end{bmatrix} \tag{A.2}$$

Asymptotic String Stability: Asymptotic string stability originated with the automated highway system [93]. Asymptotic string stability requires that there exits $\delta > 0$ such that $\sup_i \mid\mid x_i(0) \mid\mid < \delta \implies \sup_i \sup_t \mid\mid x_i(t) \mid\mid < \varepsilon$ for all $\varepsilon > 0$ and that $\mid\mid x_i(t) \mid\mid \to 0$ for all i. *The stability condition states that the infinite chain of subsystem must be uniformly bounded.* The peak error must not grow as you go through the system. Given an infinite number of subsystems the state would become unbounded. Darbha did derive sufficient conditions for asymptotic string stability. The function f_i can be nonlinear and non-autonomous but must also be identical. The subsystem is required to be exponentially stable and the subsystem connections must be sufficiently weak in terms of the constraints associated with the Lyapunov function and the Lipschitz constants of f_i [93].

Spatial Asymptotic String Stability: Spatial asymptotic string stability requires that the subsystem errors go to zero as you move through the system *spatially*, Equation A.3

$$\lim_{i\to\infty} \sup_t \mid\mid x_i(0) \mid\mid = 0 \implies \lim_{i\to\infty} \sup_t \mid\mid x_i(t) \mid\mid = 0 \tag{A.3}$$

Spatial asymptotic string stability requires the uniformity to be bounded. Although where an asymptotic string stable system is only required for uniformity in the $\mathcal{L}_\infty$ norm, spatial asymptotic string stability requires uniformity in an $\mathcal{L}_p$ norm. Darbha did derive sufficient conditions for spatial asymptotic string stability for systems with function f_i that is linear and identical. The subsystem connections must be sufficiently weak [94]. Spatial asymptotic string stability assumes that the dynamics are time-invariant.

Monotonic String Stability: Monotonic string stability is a hierarchically connected system with an infinite number of subsystems. Where a spatial asymptotic string stability requires that the error goes to zero, a monotonic string stable system is required for the peak errors to decay *monotonically* as it moves through the system, Equation A.4

$$\sup_t \| x_i(t) \| \leq \sup_t \| x_{i-1}(t) \| \tag{A.4}$$

There is always a uniform bound on the system states, even if the peak error increases from one subsystem to the next. A monotonic string stable system dynamics does not need to be identical. The idea of monotonic string stability comes from the concept of line vehicles following each other and are treated as systems with a finite hierarchically connected systems where the subsystems are arranged in a two-dimensional grid [55] [95] [96]. Monotonic string stability assumes that the dynamics are time-invariant.

Pant defined that mesh systems and hierarchically connected systems are the same through a change in variables in the subsystem indices. And that a finite hierarchically connected system is mesh stable if it is asymptotically Lyapunov stable and if $\sup_t \| x_i(t) \| \leq \max_{k<i-1} \sup_t \| x_k(t) \|$. Note that for monotonic string stability the current subsystems peak error must be less than the previous subsystems peak error. Mesh stability simply just requires the peak error to be less than the max peak error of all the previous subsystems. Mesh stability is considered recursive. The peak tracking error of all the followers must be bounded by the leader [97].

Cyclic

Wang presented a strategies called *Multi-neighbor* where each spacecraft control is designed to operate with respect to the center of mass of a subset of the neighboring spacecraft. The cycle dependency directed graph is built when two spacecraft are neighbors with one another. Each spacecraft controls itself with respect to the center of mass created between him and his neighbor [48]. [78] and [98] studied what is referred to as a *centroid* strategy, which is where a controller is set to control itself with respect to the center of mass of the entire formation. It is worth noting that centroid strategies have only been studied through simulation.

Sugihara and Suzuki studied cyclic algorithms for taking robots in random locations and having them form regular geometric shapes. They used a rule-based controller to have the robots form lines, circles, and polygons. The methodology used for setting up the circles are that the robot moves toward or away from the robot farthest away until it gets to some distance away, then move away from the closest robot [99]. This approach is similar to the theory of flocking or schooling in animals, which is the idea that they are attracted to their most distant neighbors, repulsed by the closes neighbor, and align their velocity with the velocities of their neighbors [100]. Yun built upon this work by modifying the algorithm to handle data from actuators, sensors, and collision avoidance algorithms [101]. [102] used a similar technique that used two Cyclic algorithms where potential fields are constructed using rules similar to the farthest/nearest neighbor approach. If the spacecraft is more than a given distance away from another spacecraft, the spacecraft is attracted to the neighbor, if it is less than some distance than it is driven away from the neighbor. Wang presented a stability proof for this approach with the formation equilibria characterized for four spacecraft. McInnes also used this potential field approach to space out the spacecraft in a circular orbit, stability analysis was provided [103]. Palmerini built off of McInnes work and looked at different potential functions to form the spacecraft arrangements [104]. Yamaguchi and Burdick used potential strategies where robots were attracted to two assigned neighbors [105]. Young and Beard developed a Cyclic algorithm that is similar to a virtual structure type architecture, the difference being that a goal state is prescribed instead of the motion of the robot/spacecraft, and the feedback gain the control law is dependent on the tracking errors of the spacecraft control laws. When a spacecraft falls out of formation the control gain decreases, which slows the virtual structure, allowing the spacecraft to reduce their tracking errors and reconstruct the formation [73] [106].

Cyclic Architecture Stability: Most Cyclic algorithms have only been studied through simulation due to the fact the the stability analysis is difficult. This difficulty comes because the cycles in the dependency directed graph add higher levels of feedback to the individual spacecraft feedback controllers. That being said, potential field-based cyclic algorithms usually have an associated stability proof since the potential function is the basis for the Lyapunov function [98], [101].

Advantages/Disadvantages to a Cyclic Architecture: Cyclic algorithms typically perform better than a Leader/Follower architecture since non-hierarchical connections between individual spacecraft controllers are allowed [78, 106] and the control effort is distributed more evenly [60]. Cyclic architecture do not have a coordinating agent nor instability that results from single point failures and thus the algorithm can be completely decentralized [103]. Similar to a Behavioral architecture the formation geometry is formed by the interactions of the individual controllers. A disadvantage to the Cyclic architecture is that the data requirements are similar to a Minimum Input Minimum Output (MIMO) architecture, although Cyclic architectures are typically more robust than MIMO architectures as the polygon algorithm allows the formation to adjust as the formation size changes without needing to redesigning the entire controller. Also the stability of these types of architectures is poorly understood [11].

Behavioral

Anderson and Robbins [98] give a great example of a behavioral architecture using a velocity commanded aircraft that uses a behavioral approach to perform collision avoidance, obstacle avoidance, formation maintenance, and move-to-goal behaviors. They include a specific velocity vector and weighting for each of the behaviors, the velocity of each aircraft is set to the summed value of its behavioral velocities. A Leader/Follower algorithm with a repulsive potential field centered on each spacecraft is a behavioral algorithm consisting of formation maintenance and collision avoidance as a behavior. A Formation maintenance behavior could be made up of lower level actions, but is still considered a single behavior. Control laws for individual behaviors can be considered to be formation flying control algorithms [11]. The control action for each vehicle is set up as a weighted average of the control strategies to achieve each behavior. To achieve convergence, behavioral control laws feedback the relative position between the spacecraft, compared to other strategies that feed forward the acceleration of the leader. When compared to other architectures behavioral architectures typically require less information passed between the spacecraft in the formation. Convergence analysis and bounds on the formation keeping errors for this architecture is lacking in the literature [60]. For a behavioral architecture with a formation, the formation maintenance behavior is required [78, 98].

McInnes used a behavioral architecture to maintain a constellation of satellites evenly distributed in the same orbit around earth. Lyapunov control functions were used for collision avoidance and to maintain the satellites separation distance [103]. The majority of the behavioral architecture literature is based on formation flying aircraft or mobile robots. Anderson and Robbins developed control laws that mimic the instinctive behavior of birds and applied it to aircraft [98]. Balch and Arkin derived formation keeping control strategies for mobile robots that are based on an a weighted average approach for several behaviors; formation maintenance, goal seeking, and collision avoidance. For the formation maintenance behavior they used Leader/Follower and cyclic strategies similar to the centroid strategies developed by [48], To this behavior they added obstacle avoidance and collision avoidance [78]. [99] took robots aligned in random locations and developed a simple cyclic algorithm to cause them to form different geometric shapes. [101] and [107] used a behavioral architecture to have robots create circle and line formations. A critical piece to the development of a behavioral architecture is analysis proving that the behavioral controls do not interfere with one another. [108] applied the same control laws from [101] to control a formation of robots to transport objects. Zhang and Gurfil use consensus theory to characterize the properties of the control objective of a multi-agent system [109].

Behavioral Architecture Stability: Lawton, Young, and Beard introduced the idea of the potential for coupled dynamics. Lawton et. al. state that their is a coupling between the formation maintenance behavior and the move-to-goal behavior. They determined that formation maintenance behavior can be achieved by coupling goal state tracking errors, i.e, if all the robots have the same tracking error value with respect to their goal states, then the robots are in formation. Each robot uses a feedback linearized controller to track its move-to-goal and to track the error state of its two nearest neighbors. It is worth mentioning that by itself the idea of tracking the error of two nearest neighbors is considered a Cyclic centroid strategy, for which a general stability proof does not exist. But interesting enough Lawton et. al. stabilize the Cyclic algorithm through coupling it with the move-to-goal behavior [110]. Lawton, Young, and Beard applied the idea of coupled dynamics to rotation motion as well, using rate feedback and passivity based controllers [111], compared Leader/Follower coupled dynamics in terms of control effort and tracking errors [60], and performed decomposition of an individual spacecraft current attitudes into eigenaxis and off eigenaxis components [112]. Stability proofs can be found in [60].

Multiple-Input Multiple-Output

A Multiple-Input, Multiple-Output (MIMO) architecture is when the formation is treated like a multiple-input, multiple-out put plant in a traditional control algorithm. In other words the controller is built around the dynamics of the entire formation instead of being based on a single spacecraft.

Hadaegh, Ghavimi, Singh, and Quadrelli formulated a minimal state space realization for the relative error states for a formation where all the relative spacecraft positions are constant, this type of formation is considered to be a *rigid formation.* They also designed an LQR controller for this formation [113]. [114] went on to develop an algebraic method for deriving control topologies based on linear dependencies to relative position specifications. And looked at controlling the inertial position and velocity of the formation and the unobservable states to minimize fuel usage. Speyer looked at a decentralized LQR problem, where the resulting state feedback gain matrix is equivalent to a standard, centralized LQR controller. Each spacecraft is required to estimate the state of the entire formation, using an estimator that operates on a reduced set of measurement, a locally-optimal estimator, and information communicated from other spacecraft from the formation. The data passed between spacecraft is compressed using an augmented local estimator [115]. Folta et. al. use this same decentralized LQR algorithm to fly formations at the Lagrange points [116].

Directed graphs have been used with MIMO architectures, to specify the desired formation geometry and to enforce a given spacecraft control interdependency, where the spacecraft are in circular orbits with the same semi-major axis. The directed graph directs the constant angular offsets that are to be maintained between the spacecraft [117].

Rigid and unfoldable directed graphs are used to specify the geometry of a formation. Algebraic constraints on the spacecraft locations in the directed graphs are used to generate a potential function. The gradient of this potential function constitutes the basis of the formation controller. This approach requires interdependecies in the control as the spacecraft involved in the algebraic constraint enforce it. This directed graph approach is considered to be MIMO because in order to design the formation controller a directed graph must specifies the entire formation [118]. Olfati-Saber and Murry present ideas for merging, separating and constructing rigid and unfoldable directed graphs of formations [119].

Advantages/Disadvantages to the MIMO Architecture: There are two key advantages to the MIMO architecture, these are that optimality can be guaranteed and that stability follows directly from MIMO synthesis techniques. For a MIMO architecture, the entire formation state is available for controller synthesis, this guarantees the optimality conditions are met. Since each spacecraft in the formation needs to know the state of the whole formation, MIMO architectures also have the highest data requirements. MIMO algorithms are not robust against local failures, a local failure can have a global effect. If one spacecraft fails the entire formation can go unstable. Also in order to change the size of the formation, i.e. add or subtract spacecraft, the entire controller must be redesigned [11].

Virtual Structures

Scharf, Hadaegh, and Ploen, split virtual structures into two categories; Iterated Virtual Structures (IVS) and Guidance Virtual Structures (GVS). IVS fits a formation *template* or structure to current spacecraft positions at each time step. Each spacecraft then tracks desired states with respect to this template. Coupling of spacecraft states happens through the template fitting step [11]. A Walker constellation formation fitting template was considered by [120, 121]. Several papers in the literature looked at different fitting algorithms for non-holonomically constrained robots and perturbed the fitted template to meet the formation desired states [66, 104, 122]. Using a Least-squares fit has been investigated to find the virtual center of the formation. This virtual center is considered as the virtual "leader" spacecraft and minimizes the tracking errors of all the spacecraft in the formation. This algorithm is not considered Leader/Follower since all the states are coupled through the fitting template step [71]. GVS contains an initial formation temple or structure fitting step, but then a prescribed motion of the structure is used to generate the desired trajectories. An adaptive controller with saturation constraints for a GVS design was presented in [123]. [69] and [74] used GVS to plan optimal formation rotations. [124] used a pattern matching methodology. Scharf, Hadaegh, and Ploen, make the claim that GVS by itself is not a formation flying control, due to the fact that the states are not coupled. But if the virtual structure is referenced to a real spacecraft then GVS is a type of Leader/Follower formation flying algorithm where the reference trajectories are provided by the virtual structure. They also state that GVS is the basis for a Cyclic architecture [11]. [125] looked at a spacecraft formation flying control problem using a virtual structure algorithm, modeling the effects of external disturbances, model uncertainties, sensor noises, and actuator saturation. A controller based on µ-synthesis is used to overcome the environmental disturbances. An H_∞ based linear matrix inequality controller is used to obtain a control law with lower order, using the linearized model with uncertainties. Also they used an adaptive controller, based on the Lyapunov stability theorem, to overcome a broader range of model uncertainties, which also guarantees the stability.

A.2 Relative Motion Models and Their Respective Control Approaches

HCW Equations

Formations that are required to orbit the earth, will need to consider additional disturbances, such as; gravitation spherical harmonics, atmospheric drag, and solar radiation pressure. Leader/Follower architectures have been applied to earth orbiting missions. Their are several different approaches that have been used in the past. The majority of the literature use some form of linear quadratic (LQ) control for the follower tracking. These controllers use a variation of the Hill-Clohessy-Wiltshire (HCW) equations. The HCW equations linearize the two-body gravitational dynamic equations. These equations were "first" published in 1960 by Clohessy and Wiltshire [76] [42], but it was later discovered that these equations were just another form of the previously described equations produced by Hill in 1877 [75]. These equations were later updated to account for effects due to the oblateness of the earth, also known as J2 effects [126]. Sparks designed a decoupled LQ controller that included angular velocity of the reference frame, using the modified version of the HCW equations that include J_2 effects [127]. Sparks also designed an LQ controller that looks at the frequency of thruster firings with respect to the ΔV required to overcome disturbances. Redding developed a trajectory planer that uses a feed-forward controller and gives a non-equilibrium point control offset. The discrete LQ controller also performs disturbance rejection. Zhoa, et. al. [128] investigated the problem of adaptive finite time attitude containment control for multiple spacecrafts in formation with an unknown external disturbance.

Leonard, Hollister, and Bergman used differential drag combined with switching curves such that the follower tracks the motion of the leader [129]. Kapila discretized the dynamics and derived a discrete-time LQ pulse based control law [130]. Yan added a periodic gain to this controller [131]. Chao and Vassar both developed discrete-time LQ controllers for in-plane and out-of-plane motion, Vassar for Low earth orbits (LEO) and Chao for Geostationary orbits (GEO) [132, 133]. Chao also designed an LQ controller that does not use radial thrusting [133]. Starin did similar work [134]. Xing designed an LQG controller using

GPS data [135]. Capenter [136] built an LQ controller that used the decentralized estimator of Speyer [115]. Yan and De Queiroz derived adaptive control laws to coordinate the motion of the spacecraft based on the nonlinear dynamic equations [137] [131]. Mazal and Gurfil looked at using a Leader/Follower method to develop a flight algorithm for cluster management using Lambert targeting [79]. They later looked at maintaining max and min distances with this same approach [138]. Mazal et. al. also implemented a similar LQR controller to demonstrate spacecraft rendezvous using differential drag [139]. [140] looked at developing a Leader/Follower architecture for formations of 100s to 1,000s of 100 gram spacecraft, for Swarm applications. [88] used a Leader/Follower architecture to control a formation under limited communication, where the information that needs to be shared between spacecraft for the controller only happens after a specified control event.

Keplerian Orbital Elements

Several papers use Keplerian orbital elements as opposed to Cartesian coordinates. Schaub et al. developed a control law using mean orbital elements where the reference trajectory is the leaders orbital elements with some given offest. Schaub et al compared the mean orbital element controller to a controller that uses Cartesian coordinates in an inertial reference frame [141]. Tan, Bainum, and Strong developed a similar controller that uses the osculating orbital elements. They considered a formation in an elliptical orbit [142]. [143] took a similar approach using osculating orbital element difference, but it was then mapped to the desired Cartesian position and velocity vector through a linearized transformation. When compared to a control law using mean orbital elements this controller gave a 20 meter increase in tracking accuracy from using mean orbital elements. [40] develops a method to control an individual osculating orbital elements without affecting the other elements.

Gauss' Variation Equations

Gauss' Variation Equations (GVE) have also been used to implement Leader/Follower architectures. Berger and How developed a linear time varying model for the GVEs and used these equations in a model predictive controller [23]. They later built on this model to take into account J_2 perturbation [144]. BenLarbi and Stoll derived a control concept for far range formation flying applications based on GVEs [145]

Relative Orbital Elements

Additional research has implemented Leader/Follower approaches using relative orbital elements (ROEs). D'Amico developed the Guidance, Navigation and Control algorithms for the PRISMA spacecraft based on ROEs [77]. Leomanni, Rogers and Gabriel applied techniques to improve computationally efficiency of model predictive control when applied to a low thrust rendezvous problem [146]. Spence developed a methodology for automated trajectory control of a spacecraft about a non-maneuvering target using ROEs [147].

APPENDIX B

Hill-Clohessy Wiltshire Equations Derivation

This appendix presents a derivation of the HCW equations.

B.1 HCW Equations Background

If we assume that we have two spacecraft in orbit around the earth, it is advantages to view the relative motion of this system. This motions is typically viewed in a rotating and translating reference frame. This frame is referred to as the local-vertical local horizontal (LVLH) frame. And is describe by having its origin at the center of one of the spacecraft, we will referred to this spacecraft as the Target. The x-axis of this frame is along the targets radial axis, the z-axis is aligned with the velocity direction and orthogonal to the orbit plane, and the y-axis is in the orbit plane in the direction of the local horizontal.

$$\hat{i} = \frac{\bar{r}_T}{r_T} \tag{B.1}$$

$$\hat{j} = \hat{k} \times \hat{i} \tag{B.2}$$

$$\hat{k} = \frac{\bar{r}_T \times \bar{v}_T}{r_T v_T} \tag{B.3}$$

The transformation matrix to rotate from the initial frame to the LVLH frame then becomes

$$T_I^{LVLH} = \begin{pmatrix} \hat{i}^T \\ \hat{j}^T \\ \hat{k}^T \end{pmatrix} \tag{B.4}$$

The angular velocity in the inertial frame is the angular velocity of the target position vector $\bar{r}_T$

$$\bar{\Omega} = \frac{\bar{r}_T \times \bar{v}_T}{r_T^2} \tag{B.5}$$

and from this we can see that the angular acceleration is

$$\dot{\bar{\Omega}} = \frac{2\left(\bar{r}_T \bullet \bar{v}_T\right)}{r_T^2}\bar{\Omega} \tag{B.6}$$

Since the rotating frame is centered at the target vehicle and the frame rotates with angular velocity and angular acceleration we know that

$$\bar{r}_C = \bar{r}_T + \bar{r}_{rel} \tag{B.7}$$

$$\bar{v}_C = \bar{v}_T + \bar{\Omega} \times \bar{r}_{rel} + \bar{v}_{rel} \tag{B.8}$$

$$\bar{a}_C = \bar{a}_T + \bar{\Omega} \times \bar{\Omega} \times \bar{r}_{rel} + \dot{\bar{\Omega}} \times \bar{r}_{rel} + 2\bar{\Omega} \times \bar{v}_{rel} + \bar{a}_{rel} \tag{B.9}$$

The relative position, velocity, and acceleration in the LVLH frame can now be calculated by

$$\bar{r}_{rel_LVLH} = T_I^{LVLH}\bar{r}_{rel_I} \tag{B.10}$$

$$\bar{v}_{rel_LVLH} = T_I^{LVLH}\bar{v}_{rel_I} \tag{B.11}$$

$$\bar{a}_{rel_LVLH} = T_I^{LVLH}\bar{a}_{rel_I}. \tag{B.12}$$

Linearization of the Equations of Relative Motion

Recall that the relative motion of two bodies are

$$\ddot{\mathbf{r}} = -\frac{\mu}{\mathbf{r}^{3/2}}\mathbf{r} \tag{B.13}$$

If we re-write this equation with respect to our Target/Chaser problem the equation becomes

$$\ddot{\mathbf{r}}_T = -\frac{\mu}{\mathbf{r}_T{}^{3/2}}\mathbf{r}_T \tag{B.14}$$

$$\ddot{\mathbf{r}}_C = -\frac{\mu}{\mathbf{r}_C{}^{3/2}}\mathbf{r}_C. \tag{B.15}$$

If the chaser position is

$$\bar{r}_C = \bar{r}_T + \bar{r}_{rel} \tag{B.16}$$

then from the two-body dynamics equation we get

$$\ddot{\bar{r}}_T + \ddot{\bar{r}}_{rel} = -\frac{\mu}{\mathbf{r}_C{}^{3/2}}\mathbf{r}_C \tag{B.17}$$

$$\ddot{\bar{r}}_T + \ddot{\bar{r}}_{rel} = -\frac{\mu}{[(\bar{r}_T + \bar{r}_{rel}) \bullet (\bar{r}_T + \bar{r}_{rel})]^{3/2}}(\bar{r}_T + \bar{r}_{rel}) \tag{B.18}$$

and if we assume the relative position is small,i.e.

$$\frac{r_{rel}}{r_T} << 1 \tag{B.19}$$

we can linearize this equation about

$$F(r_T, r_{rel}) = [(\bar{r}_T + \bar{r}_{rel}) \bullet (\bar{r}_T + \bar{r}_{rel})]^{-3/2} |_{r_{rel}=0} \tag{B.20}$$

using Taylor series expansion, which gives us

$$[(\bar{r}_T + \bar{r}_{rel}) \bullet (\bar{r}_T + \bar{r}_{rel})]^{-3/2} = [(\bar{r}_T \bullet \bar{r}_T) + 2(\bar{r}_T \bullet \bar{r}_{rel}) + (\bar{r}_{rel} \bullet \bar{r}_{rel})]^{-3/2} \tag{B.21}$$

$$= \frac{1}{r_T} - \frac{3x}{r_T^4} + \frac{6x^2}{r_T^5} - \frac{3y^2}{r_T^5} - \frac{3z^2}{2r_T^5} - \frac{10x^3}{r_T^6} + \frac{15xy^2}{2r_T^6} + \frac{15xz^2}{r_T^6} + ... \tag{B.22}$$

plugging this back into the equation above

$$\ddot{\bar{r}}_T + \ddot{\bar{r}}_{rel} = -\mu \left(\frac{1}{r_T} - \frac{3x}{r_T^4} + \frac{6x^2}{r_T^5} - \frac{3y^2}{r_T^5} - \frac{3z^2}{2r_T^5} - \frac{10x^3}{r_T^6} + \frac{15xy^2}{2r_T^6} + \frac{15xz^2}{r_T^6} + ... \right) (\bar{r}_T + \bar{r}_{rel}) \tag{B.23}$$

and then if we neglect the terms with order greater then one, we get

$$\ddot{\bar{r}}_T + \ddot{\bar{r}}_{rel} = -\frac{\mu \bar{r}_T}{r_T^3} - \frac{\mu}{r_T^3} \left(\bar{r}_{rel} - \frac{3}{r_T^2} \left(\bar{r_T} \bullet \bar{r}_{rel} \right) \bar{r_T} \right) \tag{B.24}$$

or

$$\ddot{\bar{r}}_T + \ddot{\bar{r}}_{rel} = -\frac{\mu}{\left(\bar{r}_T + \bar{r}_T \right)^{\frac{3}{2}}} \bar{r}_T - \frac{\mu}{\mathbf{r}_T{}^3} \left[I_{3\times3} - 3\frac{\bar{r}_T \mathbf{r}_T^T}{r_T^2} \right] \bar{r}_{rel}. \tag{B.25}$$

Then if we use the two-body dynamics equation of the target, we can simplify the above equation to

$$\ddot{\bar{r}}_{rel} = -\frac{\mu}{\mathbf{r}^3{}_T} \left[I_{3\times3} - 3\frac{\bar{r}_T \mathbf{r}_T^T}{r_T^2} \right] \bar{r}_{rel} \tag{B.26}$$

Or

$$\bar{a}_C = -\frac{\mu}{\mathbf{r}_T}{}^3 \left[I_{3\times3} - 3\frac{\bar{r}_T \mathbf{r}_T^T}{r_T^2} \right] \bar{r}_{rel}. \tag{B.27}$$

These equations describe the relative motion in the inertial frame not the rotation LVLH frame.

B.2 Hill-Clohessey-Whiltshire Equations

As mentioned in the previous section, the acceleration for the chaser vehicle can be described by

$$\bar{a}_C = \bar{a}_T + \bar{\Omega} \times \bar{\Omega} \times \bar{r}_{rel} + \dot{\bar{\Omega}} \times \bar{r}_{rel} + 2\bar{\Omega} \times \bar{v}_{rel} + \bar{a}_{rel}. \tag{B.28}$$

If we change the left hand side of this equation to the linearized form presented in the last section,

$$-\frac{\mu}{\mathbf{r}_T{}^3}\left[I_{3\times3} - 3\frac{\bar{r}_T \mathbf{r}_T^T}{r_T^2}\right]\bar{r}_{rel} = \bar{a}_T + \bar{\Omega} \times \bar{\Omega} \times \bar{r}_{rel} + \dot{\bar{\Omega}} \times \bar{r}_{rel} + 2\bar{\Omega} \times \bar{v}_{rel} + \bar{a}_{rel}. \tag{B.29}$$

Now if we assume that the above Target is in a circular orbit, then that means

$$\dot{\bar{\Omega}} = 0 \tag{B.30}$$

$$\frac{\mu}{\mathbf{r}_T{}^3} = \bar{\Omega}^2. \tag{B.31}$$

That means the equation above becomes

$$-\bar{\Omega}^2\left[I_{3\times3} - 3\frac{\bar{r}_T \mathbf{r}_T^T}{r_T^2}\right]\bar{r}_{rel} = \bar{a}_T + \bar{\Omega} \times \bar{\Omega} \times \bar{r}_{rel} + 0 \times \bar{r}_{rel} + 2\bar{\Omega} \times \bar{v}_{rel} + \bar{a}_{rel} \tag{B.32}$$

Or

$$-\bar{\Omega}^2\left[I_{3\times3} - 3\frac{\bar{r}_T \mathbf{r}_T^T}{r_T^2}\right]\bar{r}_{rel} = \bar{a}_T + \bar{\Omega} \times \bar{\Omega} \times \bar{r}_{rel} + 2\bar{\Omega} \times \bar{v}_{rel} + \bar{a}_{rel}. \tag{B.33}$$

Recall that the these vectors in the LVLV frame are described as

$$\bar{r}_{rel} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \tag{B.34}$$

$$\bar{v}_{rel} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} \tag{B.35}$$

$$\bar{a}_{rel} = \begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} \tag{B.36}$$

$$\bar{\Omega} = \begin{pmatrix} 0 \\ 0 \\ \Omega \end{pmatrix} \tag{B.37}$$

$$\bar{r}_T = \begin{pmatrix} r_T \\ 0 \\ 0 \end{pmatrix}. \tag{B.38}$$

Substituting these vector forms into the equation above we end up with

$$-\begin{pmatrix} 0 \\ 0 \\ \Omega \end{pmatrix}^2 \left[I_{3\times3} - 3\frac{\bar{r}_T\mathbf{r}_T^T}{r_T^2}\right] \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \bar{a}_T + \begin{pmatrix} 0 \\ 0 \\ \Omega \end{pmatrix} \times \begin{pmatrix} 0 \\ 0 \\ \Omega \end{pmatrix} \times \begin{pmatrix} x \\ y \\ z \end{pmatrix} + 2\begin{pmatrix} 0 \\ 0 \\ \Omega \end{pmatrix} \times \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} + \begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} \tag{B.39}$$

$$-\begin{pmatrix} 0 \\ 0 \\ \Omega \end{pmatrix}^2 \left[I_{3\times3} - 3\frac{\bar{r}_T\mathbf{r}_T^T}{r_T^2}\right] \begin{pmatrix} x \\ y \\ z \end{pmatrix} = -\frac{\mu}{\begin{pmatrix} r_T \\ 0 \\ 0 \end{pmatrix}^{3/2}} \begin{pmatrix} r_T \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \Omega \end{pmatrix} \times \begin{pmatrix} 0 \\ 0 \\ \Omega \end{pmatrix} \times \begin{pmatrix} x \\ y \\ z \end{pmatrix} + 2\begin{pmatrix} 0 \\ 0 \\ \Omega \end{pmatrix} \times \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} + \begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} \tag{B.40}$$

Or

$$\ddot{x} - 3\Omega^2 x - 2\Omega\dot{y} = 0 \tag{B.41}$$

$$\ddot{y} + 2\Omega\dot{x} = 0 \tag{B.42}$$

$$\ddot{z} + \Omega^2 z = 0 \tag{B.43}$$

The HCW equations are linear constant coefficient differential equations, thus their solutions can be obtained in closed form. The in-plane motion (x/y) is coupled, but the out of plane motion (z) is clearly decoupled with the in plane motion. The differential equation of the out of plane motion is a second order harmonic oscillator.

If we write the HCW equations in the state space form

$$\dot{x} = Ax + Bu \tag{B.44}$$

where

$$x = \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \tag{B.45}$$

and

$$\dot{x} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix}. \tag{B.46}$$

The Jacobian matrices for the HCW equations can be written as

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3\Omega^2 & 0 & 0 & 0 & 2\Omega & 0 \\ 0 & 0 & 0 & -2\Omega & 0 & 0 \\ 0 & 0 & -\Omega^2 & 0 & 0 & 0 \end{bmatrix} \tag{B.47}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{B.48}$$

The closed form of the HCW equations can be described by

$$x(t) = x_0(4 - 3\cos\Omega t) + \frac{\dot{x}_0}{\Omega}\sin\Omega t + \frac{2\dot{y}_0}{\Omega}(1 - \cos\Omega t) \tag{B.49}$$

$$y(t) = 6x_0(\sin\Omega t - \Omega t) + y_0 + \frac{2\dot{x}_0}{\Omega}(\cos\Omega t - 1) + \frac{\dot{y}_0}{\Omega}(4\sin\Omega t - 3\Omega t) \tag{B.50}$$

$$z(t) = z_0 \cos \Omega t + \frac{\dot{z}_0}{\Omega} \sin \Omega t \tag{B.51}$$

$$\dot{x}(t) = 3x_0\Omega \sin \Omega t + \dot{x}_0 \cos \Omega t + 2\dot{y}_0 \sin \Omega t \tag{B.52}$$

$$\dot{y}(t) = 6x_0\Omega \left(\cos \Omega t - 1\right) - 2x_0 \sin \Omega t + \dot{y}_0(4 \cos \Omega t - 3) \tag{B.53}$$

$$\dot{z}(t) = \dot{z}_0 \cos \Omega t - z_0\Omega \sin \Omega t. \tag{B.54}$$

When in matrix form, the HCW equations can be written in the state space form

$$\begin{bmatrix} x(t) \\ y(t) \\ z(t) \\ \dot{x}(t) \\ \dot{y}(t) \\ \dot{z}(t) \end{bmatrix} = \Phi_{6\times6} \begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ \dot{x}_0 \\ \dot{y}_0 \\ \dot{z}_0 \end{bmatrix} \tag{B.55}$$

where the state transition matrix becomes

$$\Phi_{6\times6} = \begin{bmatrix} 4 - 3\cos \Omega t & 0 & 0 & \frac{1}{\Omega} \sin \Omega t & \frac{2}{\Omega}(1 - \cos \Omega t) & 0 \\ 6\left(\sin \Omega t - \Omega t\right) & 1 & 0 & \frac{2}{\Omega}(\cos \Omega t - 1) & \frac{1}{\Omega}(4 \sin \Omega t - 3\Omega t) & 0 \\ 0 & 0 & \cos \Omega t & 0 & 0 & \frac{1}{\Omega} \sin \Omega t \\ 3\Omega \sin \Omega t & 0 & 0 & \cos \Omega t & 2 \sin \Omega t & 0 \\ 6\Omega\left(\cos \Omega t - 1\right) & 0 & 0 & -2 \sin \Omega t & 4 \cos \Omega t - 3 & 0 \\ 0 & 0 & -\Omega \sin \Omega t & 0 & 0 & \cos \Omega t \end{bmatrix} \tag{B.56}$$

or in block form

$$\begin{bmatrix} \bar{r}_{rel}(t) \\ \bar{v}_{rel}(t) \end{bmatrix} = \begin{bmatrix} \Phi_{rr} & \Phi_{rv} \\ \Phi_{vr} & \Phi_{vv} \end{bmatrix} \begin{bmatrix} \bar{r}_{rel}(t_0) \\ \bar{v}_{rel}(t_0) \end{bmatrix} \tag{B.57}$$

$$\dot{x} = \begin{bmatrix} \Phi_{rr} & \Phi_{rv} \\ \Phi_{vr} & \Phi_{vv} \end{bmatrix} \begin{bmatrix} \bar{r}_{rel}(t_0) \\ \bar{v}_{rel}(t_0) \end{bmatrix} + a. \tag{B.58}$$

These equations are named Hill-Clohessey-Wiltshire (HCW) equations. The HCW equations can be written with the control inputs u_x, u_y, u_z

$$\ddot{x} - 3\Omega^2 x - 2\Omega\dot{y} = u_x \tag{B.59}$$

$$\ddot{y} + 2\Omega\dot{x} = u_y \tag{B.60}$$

$$\ddot{z} + \Omega^2 z = u_z \tag{B.61}$$

CURRICULUM VITAE

Tyson K. Smith

Published Journal Articles

- Smith, T., Akagi, J., Droge, G. “Model Predictive Control Switching Strategy For Safe Small Satellite Cluster Formation Flying.”, *Journal of Aerospace Systems*, July 2023

- Shoop, T., Munoz, J., Gunther, J., Geller, D., Smith, T. “Formation Design for Optimal Relative Navigation in GPS Denied Environments.”, *The Journal of Astronautical Sciences*, Vol. 70, No. 2, February 2023

- Smith, T., Lewis, Z., Olsen, K., Bulcher, A. M., Whitmore, S. A. “Test Results of a Miniaturized Green End-Burning Hybrid Propulsion System for CubeSats.”, *AIAA Journal of Propulsion and Power*, Vol. 38, No. 5, May 2022

- Whitmore, Stephen A., Smith, Tyson K., “Interim Access to the International Space Station”, *AIAA Journal of Spacecraft and Rockets*, Vol. 47, No. 3, May-June 2010

- Whitmore, S A., Smith, T. K., “Launch and Deployment Analysis for a Small, MEO, Technology Demonstration Satellite”, *AIAA Journal of Spacecraft and Rockets*, Vol. 46, No. 2, March-April 2009

Published Conference Papers

- Martineau, R., Smith, T., Felt, M., Weston, C., Rusch, B., Fowler, R., Brown, E., Farmer, A., Neilsen, T., Thompson, C., Kirkman, A., Fluckiger, P., Morrison, S., Wittner, D., Picha, F., Weiss, J., Daniel, N., Wood, S., Glaser, M. “Lessons Learned

During the Implementation of a Cold Gas Propulsion System for the SunRISE Mission." *Small Satellite Conference 2023*, SSC23-VI–04.

- Smith, T., Akagi, J., Droge, G. "Spacecraft Formation Flying Control Switching Surface Based on Relative Orbital Elements." *2022 AAS/AIAA Astrodynamics Specialist Conference*, 1–19.

- Smith, T. K., Lewis, Z., Olsen, K., Thibaudeau, R., Whitmore, S. A. "A Miniaturized Hydrogen Peroxide/ABS Based Hybrid Propulsion Systems for CubeSats" *Small Satellite Conference 2022*, SSC22–X–02.

- Shoop, T., Munoz, J., Gunther, J., Geller, D., Smith, T. "Formation Design For Optimal Relative Navigation in GPS Denied Environments." *45th Annual AAS Guidance and Controls Conference*, Breckenridge, CO, February 2022

- Whitmore, S. A., Smith T. K. "3-D Printing of High-Performance Green Hybrid Propulsion (HPGHP) Solutions." *AIAA Wasatch Aerospace and Systems Engineering Mini-Conference April 2021*

- Smith, T., Lewis, Z., Olsen, K., Bulcher, A. M., Whitmore, S. A. "End-Burning Hybrid Propulsion System for CubeSats." *September 2020 NASA In-Space Chemical Propulsion TIM and JANNAF LPS AMP Additive Manufacturing for Propulsion Applications TIM.*

- Cox, S., Smith, T., Jones, T. L., Droge, G., Jones, A. S. "Power-Optimal Slew Maneuvers in Support of Small Satellite Earth Imaging Missions". *AIAA Small Satellite Conference 2020*, SSC20-P3-1.

- Smith, T., Lewis, Z., Olsen, K., Bulcher, A. M., Whitmore, S. A. "A Miniaturized Green End-Burning Hybrid Propulsion System for CubeSats." *AIAA Small Satellite Conference 2020*, SSC20-IX-09.

- Shuster, S., Geller, D., Smith, T., “A Performance Analysis of On-Board Numerical Propagators”, *41st Annual AAS Guidance and Controls Conference*, Breckenridge, CO, February 2018

- Whitmore, Stephen A., Smith, Tyson K., “Interim Access to the International Space Station”, *AIAA Space 2009 Conference and Exposition*, Pasadena, CA, September 2009

- Whitmore, S A., Smith, T. K., “Launch and Deployment Analysis for a Small, MEO, Technology Demonstration Satellite”, *46th AIAA Aerospace Sciences Meeting and Exhibit Conference*, Reno, NV, January 2008

Published Peer Reviewed Technical Reports

- Smith T., Skulsky, E., “State of the Art Small Spacecraft Technology: GNC Chapter”, NASA/TP—2020, *Small Spacecraft Systems Virtual Institute*, Ames Research Center, Moffit Field, CA

- Marshall W., el al., “State of the Art Small Spacecraft Technology: Propulsion Chapter”, NASA/TP—2020, *Small Spacecraft Systems Virtual Institute*, Ames Research Center, Moffit Field, CA

Thesis

- Smith, Tyson Karl, “Interim Access to the International Space Station” (2009). Master’s Thesis, Utah State University.

Patents

- Smith, T., Lewis, Z., Olsen, K., Bulcher, A. M., Whitmore, S. A. Miniaturized Green End-Burning Hybrid Propulsion System for CubeSats, No.: US 11,724,829 B2, Date Aug 15, 2023

- Smith, T., Lewis, Z., Olsen, K., Bulcher, A. M., Whitmore, S. A. Miniaturized Green End-Burning Hybrid Propulsion System for CubeSats, Patenet No.: US 11,407,531 B2, Date Aug 9, 2022

Patents Pending

- Smith, T., Akagi J., Droge G., 2022, Model Predictive Control for Spacecraft Formation, 17/891,968

- Smith, T., Akagi J., Droge G., 2022, Model Predictive Control for Spacecraft Formation, 17/689,038

- Smith, T., Lewis, Z., Olsen, K., Bulcher, A. M., Whitmore, S. A. 2021, Miniaturized Green End-Burning Hybrid Propulsion System for CubeSats, 17/391,843

- Cox, S., Smith, T., Jones, T. L., 2021, Power Enhanced Slew Maneuvers, 17/389,246