(Preprint) AAS 23-148

POST-DOCKING SPACECRAFT DYNAMICS USING BAUMGARTE STABILIZATION

João Vaz Carneiro*, Andrew Morell* and Hanspeter Schaub†

There are many challenges with spacecraft life extension missions and debris or defunct-satellite removal operations. Many envisioned space exploration concepts rely heavily on on-orbit servicing, assembly, and manufacturing (OSAM). These procedures are very complex due to the coupled motion between the servicer and the target spacecraft. Simulating the physical interactions between multiple satellites is challenging and typically entails developing specific models for the servicer and the target spacecraft. This process is time-consuming and must be repeated for each mission. This paper proposes a modular approach to accurately describe the dynamics of two spacecraft docked to each other. Taking advantage of classic control techniques, the dynamics of two docked spacecraft, each with its own reaction wheels or flexible structures, can be simulated by numerically enforcing the constraints between these two complex bodies. This numerical approach to model constrained motion between two complex vehicles shows promising results, with constraint violations that are multiple orders of magnitude smaller than the size of each spacecraft and the ability to apply orbital and attitude maneuvers while enforcing the constraints successfully.

INTRODUCTION

Many spacecraft currently in Earth's orbit have their missions end not due to catastrophic faults or faulty sensors and instruments, but because they run out of fuel to make orbital and attitude corrections. Some of these high-value spacecraft, especially those in geostationary orbit (GEO), are very useful for communications, meteorology, and navigation. Due to their altitude, they are not significantly impacted by atmospheric drag and take a long time to fall into a lower orbit without spending fuel. Extending the life of spacecraft already in orbit, especially those occupying important orbits, as well as also disposing of defunct satellites, is essential to reducing space-based service costs and minimizing the amount of orbital debris around Earth.

Given these issues and the increased interest in space commercialization, space sustainability has become a major focus of a multitude of private companies and government programs alike.[1] One of the earliest and most extensive examples of on-orbit servicing was the Hubble Space Telescope servicing missions performed by NASA.[2] Private on-orbit servicing missions have been successfully completed by Northrop Grumman's subsidiary SpaceLogistics which started providing life extension for communications satellites in 2020 and 2021 with its Mission Extension Vehicle.[3] Entities

*Graduate Research Assistant, Ann and H.J. Smead Department of Aerospace Engineering Sciences, University of Colorado, Boulder, 431 UCB, Colorado Center for Astrodynamics Research, Boulder, CO, 80309.

†Professor and Department Chair, Schaden Leadership Chair, Ann and H.J. Smead Department of Aerospace Engineering Sciences, University of Colorado, Boulder, 431 UCB, Colorado Center for Astrodynamics Research, Boulder, CO, 80309. AAS Fellow, AIAA Fellow.

currently developing on-orbit servicing technologies include NASA's OSAM-1 mission[4] (previously Restore-L[5]), the European Space Agency's robotic debris removal mission ClearSpace-1,[6] and OrbitFab's Rapid Attachable Fluid Transfer Interface[7] to name a few. Some of these missions focus on removing inactive spacecraft, while others focus on servicing and/or refueling out-of-commission satellites. Besides fixing spacecraft issues, new modules such as a space domain awareness (SDA) sensor can be added to spacecraft to adapt them as technology evolves faster than their orbital lifetime. While they have different mission requirements, all of these scenarios involve a servicer spacecraft reaching a satellite in orbit and docking with it, whether it be to provide a space tug or servicing.

Successfully designing these missions requires extensive spacecraft simulations to guarantee the fulfillment of all mission requirements. However, simulating these complex dynamics is far from trivial, as each spacecraft can have active attitude control devices, moving hinged solar panels, thrusters, etc. Analytically solving for the post-docking dynamics of each spacecraft is a complex and lengthy process that requires precise knowledge of each spacecraft's components, making it a mission-specific process. Previous work has focused on creating simulations based on actual missions[8,9] by creating a digital twin of the mission hardware. This is expensive and specific to these missions, lacking the generality needed for early-mission design. A common approach is using multiple software packages to simulate these complex dynamics, such as MATLAB* and STK†.

This paper puts forward a general numerical dynamics solution to the problem of two docked spacecraft. Instead of analytically solving for the contribution of each spacecraft on the other, the docking constraint is implemented using a numerical proportional-integral-derivative (PID) controller inspired by the Baumgarte stabilization method[10] used in numerical simulation of multi-body systems. This method has seen specific use in robotics applications, and many papers that study and explore it give robotic arms as an example of a use-case scenario.[11,12] The use of this constraint stabilization method to model docked complex spacecraft has not been considered to date. The kinematic relationship between the two spacecraft is defined through holonomic constraints. Previous work in Ref. 13 has focused on analytically solving for the kinematic relationships between the servicer and the target satellites and then using a second-order sliding mode controller to impose these constraints.

The proposed force and torque solution is general so that it can be applied to multiple scenarios and different connection types. Three different connection types are explored, progressively restricting the system's degrees of freedom. This ensures that the method works for multiple different constraints of increased rigidity. One of the challenges with this approach is that to satisfy the constraints, high gains need to be used for the constraint forces and torques. While the post-docking maneuvers are usually slow to avoid stressing the connections between the servicer and target spacecraft, these high constraint forces and torques lead to very stiff differential equations. Nonetheless, with the appropriate time step and gain selection, the approach is shown to work in a variety of scenarios.

The outcome is a general formulation for the dynamics formulation of two spacecraft docked in orbit, which is applicable to a number of different scenarios. In particular, single-point solutions are avoided, which only apply to a single configuration of docking spacecraft. Rather, the goal is to be able to use two separate spacecraft models, each with complex component dynamics, and en-

*https://www.mathworks.com/products/matlab.html

†https://www.ansys.com/products/missions/ansys-stk

force docking constraints without having to re-derive problem-specific system equations of motion. Another benefit of this approach is that it yields a time history of the forces and torques needed to satisfy the constraints. When solved analytically, these forces and torques are not directly computed because the constraints are embedded in the equations of motion, and the internal forces and torques do not directly appear. With these, a first-cut approximation of the structural forces and torques can be computed, which is very useful to understand if the two spacecraft can withstand the stresses felt post-docking.

This paper is organized as follows. The problem statement is defined, the general scenario is discussed, and the different constraints are explained. A mathematical overview of the constraint definition is given, along with the method of enforcement through Baumgarte stabilization. Each constraint is then formulated analytically. Finally, a numerical analysis is done, which includes gain analysis and maneuver application.

PROBLEM FORMULATION

This problem formulation aims to be realistic while keeping it general enough to be open to multiple implementations. The general scenario consists of a two-spacecraft configuration after a docking procedure shown in Figure 1. The satellites are attached through a rigid arm, with increas-

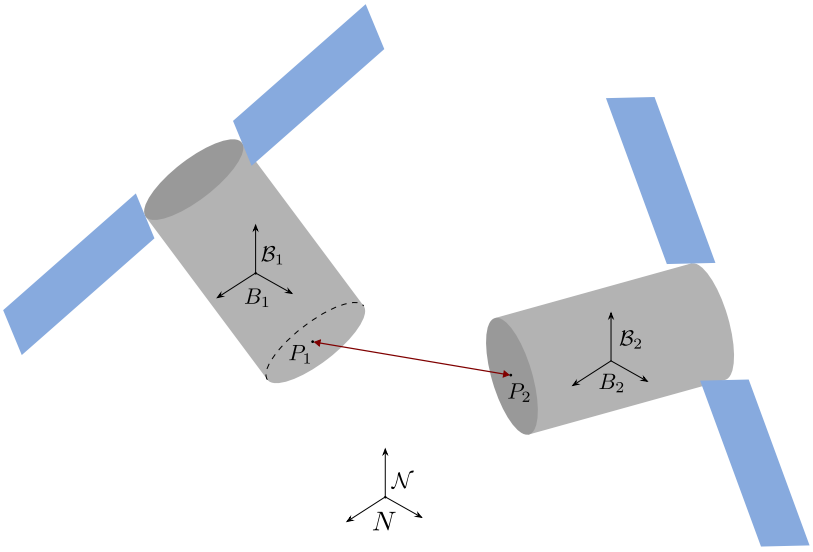


Figure 1. Generalized 6-DOF configuration.

ing levels of rigidity at the hinge points. Both spacecraft are orbiting Earth in a low Earth orbit (LEO) regime, and each spacecraft is a 6-degree-of-freedom system that includes translation and rotation.

The inertial frame $\mathcal{N}$ has its origin in N, spacecraft 1's body frame $\mathcal{B}_1$ is centered at its center of mass B_1, and spacecraft 2's body frame $\mathcal{B}_2$ is centered at its center of mass B_2. Setting the spacecraft's origin at the corresponding center of mass simplifies the equations of motion, as the center of mass is assumed to be constant in the body frame. Spacecraft 1 has mass m_1 and moment of inertia about its center of mass $[I_{1,B_1}]$. Similarly, spacecraft 2 has mass m_2 and moment of inertia $[I_{2,B_2}]$. Spacecraft 1 and 2 each have connection points P_1 and P_2, which are attached to each other by a rigid arm. The connection points are assumed to be fixed to their respective spacecraft hub.

Multiple constraints are studied in this work. The scenarios gradually change from a simple one-degree-of-freedom translation constraint to a full translational and rotational lock between the two bodies. This approach allows the stabilization method to be progressively tested in more constrained scenarios. These tend to be more realistic because, in practice, it is uncommon to have uncontrolled spherical joints in spacecraft operations.

Length Constraint

The length constraint consists of a rigid arm connecting points P_1 and P_2 with a spherical hinge at each end. Both spacecraft must keep a fixed distance l between themselves, although they can rotate about each connection point. This is the simplest translational restriction between two connected bodies, as it is a one-degree-of-freedom constraint. It is enforced by keeping the length of the connecting rigid arm at a fixed length, with no regard for the attitude and angular velocity of each spacecraft.

Direction Constraint

The direction constraint is similar to the length constraint, but spacecraft 1 can no longer rotate about its connection point P_1. In practice, there is now only a single connection point at P_2, and the second spacecraft continues to rotate about it through a spherical hinge. This constraint now locks three-degree-of-freedom and is implemented by setting the direction of $\mathbf{r}_{P_2/P_1}$ fixed in the B_1 frame, where $\mathbf{r}_{P_2/P_1}$ denotes the vector from point P_1 to point P_2.

Rigid Direction Constraint

The rigid direction constraint uses the principles described in the direction constraint but also fixes the second spacecraft's attitude. Therefore, the second spacecraft does not rotate about point P_2. As with the direction constraint, the direction of $\mathbf{r}_{P_2/P_1}$ is fixed in the B_1 frame, with the addition of fixing the rotation of spacecraft 2 about its connection point. This constraint fixes the entire system in place: no translation or rotation is allowed between each spacecraft, which results in a six-degree-of-freedom constraint. The two spacecraft are a single rigid body system with six degrees of freedom (translational and rotational).

MATHEMATICAL OVERVIEW

The holonomic constraint ψ can be defined through a one-dimensional equation that depends on the generalized system coordinates q_i as

$$\psi(q_1, ..., q_n) = 0 \tag{1}$$

These generalized coordinates can be the position, velocity, or attitude components. Each constraint is implemented into the system's equations of motion through a constraint force $\boldsymbol{F}_c$. This force is given by

$$\boldsymbol{F}_c = \lambda \nabla \psi \tag{2}$$

where λ denotes the Lagrange multiplier and ∇ is the spacial derivative operator. The equation above implies that the constraint force is always perpendicular to the constraint surface, which is consistent with physical reality. The constraint force acts on the system to keep the constraint from being violated. Therefore, it will never act on the constraint surface directly, as motion in that surface does not violate the constraint.

Baumgarte Stabilization

Finding the analytic Lagrange multiplier is non-trivial, especially for complex spacecraft systems. Rendezvous, proximity operations, and docking (RPOD) scenarios involve highly coupled dynamics for which solving for the correct Lagrange multiplier is very cumbersome.

A simple approach to avoid this can be to use algebraic constraint elimination: take the second time derivative of the constraint ψ and substitute the result into the equations of motion of the system. Let us consider the planar pendulum system to motivate why this is not an appropriate approach. The pendulum's length r is fixed with length l, which means that the constraint is $\psi = r - l$. Taking the second-order time derivative yields $\ddot{\psi} = \ddot{r} = 0$. When implemented numerically, this constraint is affected by numerical instability and finite precision, which means that $\ddot{\psi} = \epsilon \approx 0$, where ϵ is a very small number. While this may not look like an issue, integrating the second-order time derivative yields $\dot{r} = \epsilon t$ and $r = l + \epsilon t^2$. While the value of ϵ can be very small, with enough time, the constraint is bound to be violated above any acceptable threshold.

The Baumgarte stabilization method[10] is used to circumvent these issues. This method leverages classic control techniques to simulate constrained dynamics. Instead of solving for the analytic expression of the Lagrange multiplier, a proportional-derivative (PD) controller is used instead, as shown below

$$\lambda = k\psi + c\dot{\psi} \tag{3}$$

where k and c are proportional and derivative gains specified by the user, respectively. This approach ensures that the constraint force is larger when the system is farther away from the constraint surface, going down in value as the system approaches that surface. To allow for perturbation rejection and improve performance, an integral feedback term is included, which yields a slightly modified expression for the Lagrange multiplier as follows

$$\lambda = k\psi + k_{\mathrm{I}} \int \psi + c\dot{\psi} \tag{4}$$

where k_I corresponds to the gain for the integral feedback term $\int \psi$. While ψ and $\dot{\psi}$ can be analytically computed depending on the problem setup, the integral feedback term is always computed numerically by setting its derivative equal to ψ. Therefore, using a PID controller adds as many states as the number of constraints on the system, whereas a PD controller does not need any additional states.

For this method to work, some important considerations must be assessed. The values of the gains c and k must be chosen to be large enough such that the constraints are enforced within a fixed tolerance over a time scale that is smaller than the smallest time scale of the original dynamical system. Aggressive maneuvering can lead to very stiff differential equations. Therefore, a sufficiently small time step of integration is needed to run these simulations, as numerical issues may arise. However, the relative motion of two spacecraft tends to be very slow, so one of the objectives is to understand how well this method can be applied to increasingly complex connected spacecraft systems.

CONSTRAINT FORMULATION

The different constraints need to be mathematically defined to implement them into the simulation. It is especially important to be cautious in which frame the constraints are defined and with

respect to which frame the constraint derivative is taken. Since both spacecraft are in orbit, in general, the constraints and their derivatives are defined with respect to the body frames as the relative motion between the two spacecraft is being analyzed, not the motion with respect to the inertial frame.

Length Constraint

The length constraint enforces a fixed length between the two connection points P_1 and P_2. Assuming that the constraint length is l, then the constraint is defined as

$$\psi = \|\mathbf{r}_{P_2/P_1}\| - r \tag{5}$$

where r is the rigid arm's length. Since $\mathbf{r}_{P_1/B_1}$ and $\mathbf{r}_{P_2/B_2}$ are constant, then the $\mathbf{r}_{P_2/P_1}$ vector is usually obtained using these two quantities in addition to the inertial position and velocity of each of the spacecraft's center of mass:

$$\mathbf{r}_{P_2/P_1} = \mathbf{r}_{P_2/N} - \mathbf{r}_{P_1/N} = \mathbf{r}_{P_2/B_2} + \mathbf{r}_{B_2/N} - \mathbf{r}_{P_1/B_1} - \mathbf{r}_{B_1/N} \tag{6}$$

The time derivative of the constraint is given by

$$\dot{\psi} = \hat{\mathbf{r}}_{P_2/P_1} \cdot \dot{\mathbf{r}}_{P_2/P_1} \tag{7}$$

where $\hat{\mathbf{r}}_{P_2/P_1}$ corresponds to the unit vector from point P_1 to point P_2 and is given by

$$\hat{\mathbf{r}}_{P_2/P_1} = \frac{\mathbf{r}_{P_2/P_1}}{\|\mathbf{r}_{P_2/P_1}\|} \tag{8}$$

Now that the constraint and constraint rate are analytically defined, the Lagrange multiplier can be computed using either (3) or (4). The gradient of the constraint for spacecraft 1 is $\nabla_1\psi = -\hat{\mathbf{r}}_{P_2/P_1}$ and for spacecraft 1 it is $\nabla_2\psi = \hat{\mathbf{r}}_{P_2/P_1}$. Therefore, the constraint forces acting on each spacecraft are given by

$$\mathbf{F}_1 = -\lambda\hat{\mathbf{r}}_{P_2/P_1}, \qquad \mathbf{F}_2 = \lambda\hat{\mathbf{r}}_{P_2/P_1} \tag{9}$$

The constraint force acting on spacecraft 1 is equal in magnitude and opposite in direction to the constraint forces acting on spacecraft 2. This is consistent with reality, as when you consider the system of both spacecraft, these are internal forces that should cancel out. It should also be pointed out that the forces act along the rigid arm. Since the goal is to maintain the fixed length of the rigid arm, the constraint forces cannot act in any other direction. If that were the case, the constraint forces would rotate the rigid arm, not acting on the length itself.

It is important to note that because the length constraint only enforces one degree of freedom, each spacecraft is free to rotate about its connection point. Therefore, in addition to the constraint forces applied to each spacecraft at their corresponding connection point, each spacecraft also experiences a torque that arises from the offset between the connection point and the center of mass of each spacecraft as follows

$$\mathbf{L}_{B_1} = \mathbf{r}_{P_1/B_1} \times \mathbf{F}_1, \qquad \mathbf{L}_{B_2} = \mathbf{r}_{P_2/B_2} \times \mathbf{F}_2 \tag{10}$$

Direction Constraint

The direction constraint enforces a fixed direction and length of the connecting arm from spacecraft 1. This means that there are now three one-dimensional constraints defined by

$$\psi^i = \mathbf{r}^i_{P_2/P_1} - \mathbf{r}^i, \quad i = 1, 2, 3 \tag{11}$$

where $\mathbf{r}$ denotes the vector that describes the connecting rigid arm, and the i superscript represents the i-th component of the corresponding vector. Because we have three constraints, the translational motion between both spacecraft is locked. The time derivative of each constraint is

$$\dot{\psi}^i = \left(\frac{^{\mathcal{B}_1}\mathrm{d}}{\mathrm{d}t} \mathbf{r}_{P_2/P_1} \right)^i, \quad i = 1, 2, 3 \tag{12}$$

The time derivative above is taken with respect to the $\mathcal{B}_1$ frame because the rigid arm should have a fixed direction with respect to spacecraft 1. Since both spacecraft are in orbit, the rigid arm is going to move with respect to the inertial frame. That motion seen by the inertial frame does not necessarily imply a constraint violation. The constraint can be properly analyzed only by taking the time rate of change in the $\mathcal{B}_1$ frame.

The spatial gradient is taken for each scalar constraint to find the constraint forces, which yields three distinct forces. Adding them all up yields a single general constraint for each spacecraft, which is given by

$$\mathbf{F}_1 = \begin{bmatrix} -\lambda^1 \\ -\lambda^2 \\ -\lambda^3 \end{bmatrix}, \qquad \mathbf{F}_2 = \begin{bmatrix} \lambda^1 \\ \lambda^2 \\ \lambda^3 \end{bmatrix} \tag{13}$$

where λ^i corresponds to the Lagrange multiplier from constraint ψ^i. If the constraint and its time derivative are written in a particular frame, so is the constraint force. Again, the forces are equal in magnitude and opposite in direction, canceling out when the system is considered a whole. Since the rigid arm is now fixed at point P_1, the constraint force is applied at point P_2 for both spacecraft. Therefore, the torques experienced by each spacecraft are

$$\mathbf{L}_{B_1} = \mathbf{r}_{P_2/B_1} \times \mathbf{F}_1, \qquad \mathbf{L}_{B_2} = \mathbf{r}_{P_2/B_2} \times \mathbf{F}_2 \tag{14}$$

where $\mathbf{r}_{P_2/B_1} = \mathbf{r}_{P_2/P_1} + \mathbf{r}_{P_1/B_1}$.

Rigid Direction Constraint

The rigid direction constraint builds on top of the direction constraint with the addition of locking spacecraft 2 about its connection point P_2. Therefore, we need to define two distinct constraints: ψ_T for the translational constraint and ψ_R for the rotational constraint. The translation constraint is identical to the direction constraint:

$$\psi^i_T = \mathbf{r}^i_{P_2/P_1} - \mathbf{r}^i, \quad i = 1, 2, 3 \tag{15}$$

as is the time derivative:

$$\dot{\psi}^i_T = \left(\frac{^{\mathcal{B}_1}\mathrm{d}}{\mathrm{d}t} \mathbf{r}_{P_2/P_1} \right)^i, \quad i = 1, 2, 3 \tag{16}$$

The constraint forces are

$$\mathbf{F}_1 = \begin{bmatrix} -\lambda_T^1 \\ -\lambda_T^2 \\ -\lambda_T^3 \end{bmatrix}, \qquad \mathbf{F}_2 = \begin{bmatrix} \lambda_T^1 \\ \lambda_T^2 \\ \lambda_T^3 \end{bmatrix} \tag{17}$$

For the rotational constraint, the relative attitude between both spacecraft is locked, which yields the following formulation

$$\psi_R^i = \boldsymbol{\sigma}_{\mathcal{B}_2/\mathcal{B}_1}^i - \boldsymbol{\sigma}^i \tag{18}$$

where $\boldsymbol{\sigma}$ is the offset between the two spacecraft written in modified Rodrigues parameters. While this offset is set to zero throughout this work, it can be defined to any value according to the particular implementation. The gradient needs to be computed to enforce this constraint, now in attitude space. This yields a torque defined as

$$\mathbf{L}_c = \begin{bmatrix} -\lambda_R^1 \\ -\lambda_R^2 \\ -\lambda_R^3 \end{bmatrix} \tag{19}$$

where $\lambda_R^i = -k_R\psi_R^i - c_R\dot{\psi}_R^i$. In practice, the formulation used is slightly different than the one used to enforce the translational constraints. $\dot{\boldsymbol{\psi}}_R$ is not used, and the relative angular velocity is used instead. Since each constraint rate is $\dot{\psi}_R^i = \dot{\boldsymbol{\sigma}}_{\mathcal{B}_2/\mathcal{B}_1}^i$, then we can map it into the relative angular velocity as follows

$$\dot{\boldsymbol{\psi}}_R = \dot{\boldsymbol{\sigma}}_{\mathcal{B}_2/\mathcal{B}_1} = \frac{1}{4}[B\left(\boldsymbol{\sigma}_{\mathcal{B}_2/\mathcal{B}_1}\right)]\boldsymbol{\omega}_{\mathcal{B}_2/\mathcal{B}_1} \tag{20}$$

where $\dot{\boldsymbol{\psi}}_R = \left[\dot{\psi}_R^1, \dot{\psi}_R^2, \dot{\psi}_R^3\right]^T$ and $\boldsymbol{\omega}_{\mathcal{B}_2/\mathcal{B}_1}$ is the relative angular velocity between the two spacecraft. This approach is preferable because the angular velocity translates directly into a physical quantity and is one of the system's state variables.

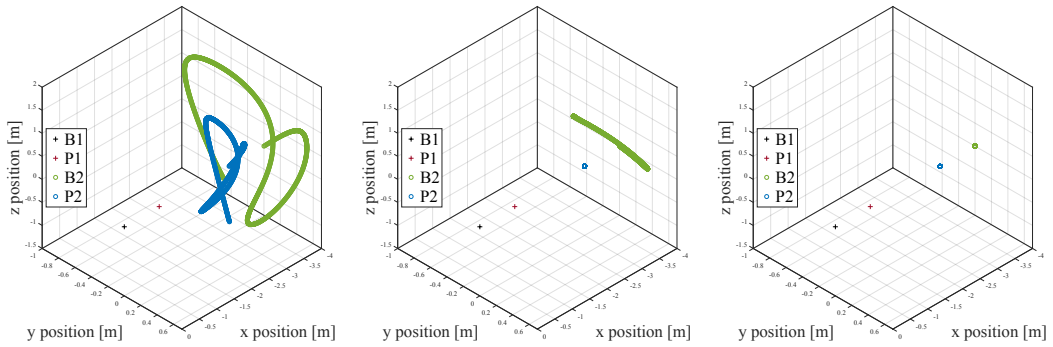
In addition to the constraint forces and torques, the additional torques described in (14) are also present, as the constraint forces are also applied at the point P_2.

NUMERICAL RESULTS

This section shows the corresponding numerical results. These results demonstrate that the stabilization method suits the three constraints described in previous sections. Next, a gain analysis for the PD and PID approaches focuses on the constraint violation performance and the runtime. Next, runtime analysis is performed to show the feasibility of the method's computational time at varying gains. Finally, orbital and attitude maneuver analysis show that this approach works in more challenging scenarios with dynamic behavior.

Constraint Analysis

An initial qualitative assessment of constraint effectiveness can be inferred from observing the relative motion of the two spacecraft. The four points of interest: B_1, B_2, P_1, and P_2 from the configuration shown in Figure 1 are plotted in the $\mathcal{B}_1$ frame for 20 minutes in Figure 2 for each of the different constraint types used.



(a) Length Constraint (b) Direction Constraint (c) Rigid Constraint

Figure 2. Relative motion in the $\mathcal{B}_1$ frame for each constraint type.

For all three plots, points B_1 and P_1 stay fixed as is expected considering their fixed definition in the $\mathcal{B}_1$ frame. In Figure 2(a), the relative motion of the length constraint sees P_2 moving on a spherical surface around P_1, and B_2 following at a fixed distance behind P_2. This is consistent with enforcing a fixed distance between P_1 and P_2 without restrictions on either spacecraft's attitude. Figure 2(b) shows P_2 fixed via the direction constraint as is expected by enforcing the direction of P_2 in the $\mathcal{B}_1$ frame. Finally, Figure 2(c) shows all four points of interest fixed in the $\mathcal{B}_1$ frame when enforcing the rigid constraint, as is expected for the additional attitude requirement on $\mathcal{B}_2$ with respect to $\mathcal{B}_1$.

PD Gain Analysis

When using the proportional-derivative (PD) formulation, the gains are represented by proportional gain k and derivative gain c. These gains depend on the Baumgarte stabilization parameters α and β through the following relationships originally used in Baumgarte's stabilization gains.[10]

$$k = \alpha^2, \quad c = 2\beta \tag{21}$$

For this work, the parameters α and β are equal, making the simulation parameterized by a single variable. The scenario used for all simulations in the PD and PID gain analyses sets spacecraft 1 as a servicer spacecraft positioned radially on the opposite side from the Earth of target spacecraft 2.

The constraint violations for the length constraint scenario using varying gains are shown in Figure 3. The length constraint violation plot shows that the errors decrease with increasing gains. Assuming a reasonable integration time step, this is expected, as higher gains make the simulation more sensitive to constraint errors. However, these benefits start to plateau for values of α greater than 10^2, and the results become noisier. This can be attributed to a few causes. First, the constraint violations are close to 10^{-10}, approaching the available machine precision. Second, numerical integration also has finite precision, and the propagation errors start to outweigh the benefits of making the dynamics stiffer and more sensitive to constraint violations. Nonetheless, the errors stabilize at 10^{-9} meters, which is more than sufficient for orbital simulations.

The direction constraint uses three one-dimensional constraints to calculate the constraint forces and torques. In order to analyze the constraint violations with a single variable, the norm of the difference between $\mathbf{r}_{P_2/P_1}$ and $\mathbf{r}$ is used, similar to equation (11). This metric includes the direction

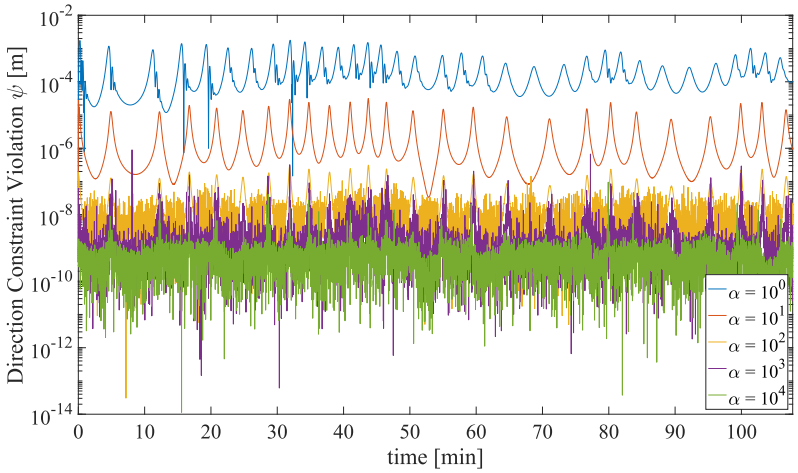


Figure 3. Length constraint violations with the PD formulation.

information, as opposed to the previous metric, which only included the length information. It can be interpreted as an upper bound on the individual constraint violations.

The direction constraint violations for varying gains are shown in Figure 4. Again, the constraint

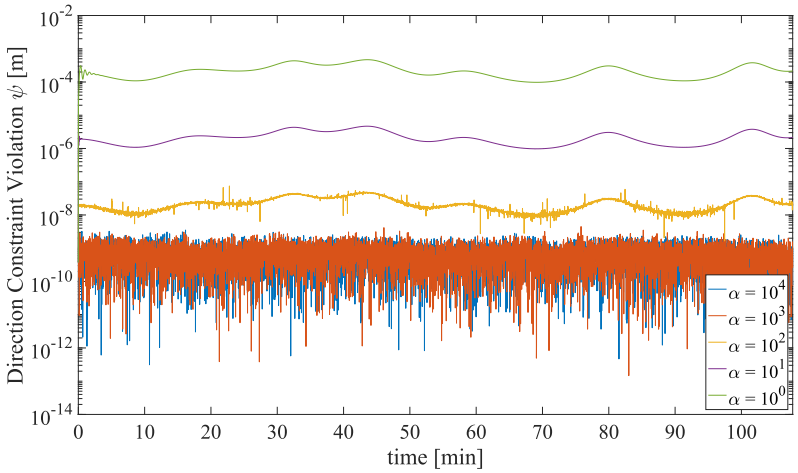


Figure 4. Direction constraint violations with the PD formulation.

violations are lower for higher gains until they plateau for $\alpha \geq 10^3$, becoming noisier with increased values of α. Finally, the violations settle on the order of 10^{-9}, which is more than adequate for spacecraft simulations. Comparing this Figure 4 with Figure 3, there is an oscillatory behavior in the length constraint scenario that is not present in the direction constraint. This is because the length constraint allows both spacecraft to freely rotate about their hinges, similar to an unstable pendulum. The oscillations are typical of the back-and-forth snapping common to these systems.

For the rigid constraint, there is an additional rotational constraint that needs to be met. The PD

gains for the attitude component are calculated using the relationship in (22).

$$k = \left(\frac{\alpha}{10}\right)^2, \quad c = 2\frac{\beta}{10} \tag{22}$$

This relationship between the attitude PD gains and alpha was picked by tuning gain values to minimize attitude constraint violations with $\alpha = 10^2$ held constant. The translation analysis follows the same principles as the direction constraint for the rigid constraint implementation.

The direction constraint violations for varying gains are shown in Figure 5, and attitude constraint violations for varying gains are shown in Figure 6. The direction constraint violation results using

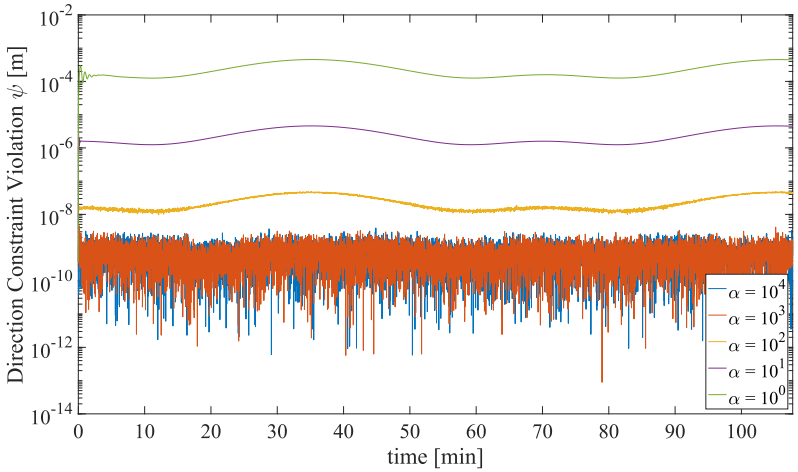


Figure 5. Rigid constraint direction violations with the PD formulation.

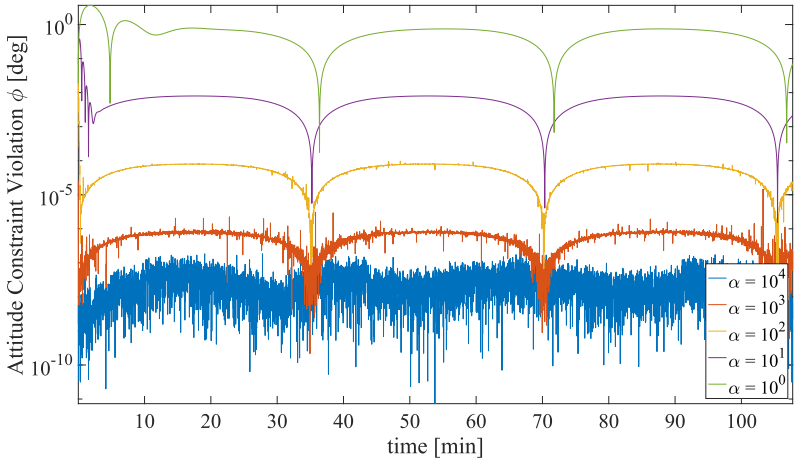


Figure 6. Rigid constraint attitude violations with the PD formulation.

the rigid constraint are similar to those in Figure 4 from the standalone direction constraint but with a less noticeable oscillatory behavior due to the increased rigidity of this scenario. As for the attitude constraint violations, they are almost constant throughout the simulation except when their

value crosses zero at regular intervals (note the logarithmic scale). Following previous trends, the performance increases with higher gains until the results get affected by numerical issues. In this case, the results do not reach a plateau, although it is essential to reiterate that the gains for the attitude constraint are smaller.

While constraint performance is the key metric to understanding whether the proposed method works, it does not tell the whole story. For the approach to be practical, the run time has to be reasonable; otherwise, each simulation takes too long and becomes inappropriate for mission analysis. To that end, the runtimes for each test in the PD gain analysis are shown in Figure 7. The larger

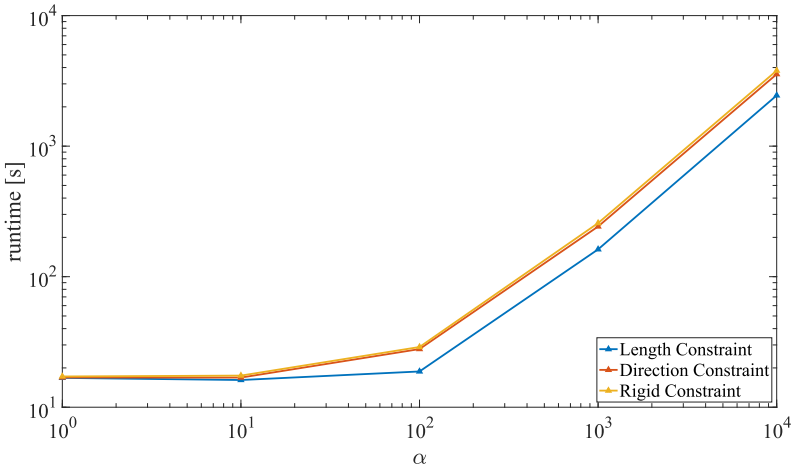


Figure 7. Runtime analysis for all constraint types using the PD formulation.

the gains, the longer the simulation takes to run. As α increases, the PD gains increase, increasing the constraint forces and causing the Runge-Kutta RK45 integrator to take smaller integration time steps. This happens because the RK45 integrator uses variable time steps, so the length of simulation time grows dependent on the stiffness of the equations of motion: the stiffer the system, the smaller the integration steps.

PID Gain Analysis

In the PID gain analysis, the proportional and derivative gains for each type of constraint are formulated similarly from parameters α and β as in the PD gain analysis. The integral gain k_I is set to be one order of magnitude smaller than the corresponding proportional gain for each constraint. This relationship between k and k_I was selected from tuning the PID gains to minimize the constraint violations with $\alpha = 10^2$ held constant. The analysis focuses on the same goals of constraint performance and simulation runtime.

The length constraint violations for this formulation are shown in Figure 8. Following the same trend observed with the PD formulation, the constraint violations, on average, decrease as alpha increases. Also similar to the results using the length constraint with the PD formulation, there is oscillatory motion caused by the unstable pendulum-type motion of each spacecraft on the ends of the fixed length snapping back and forth when perturbed by the constraint force. These resulting oscillations in constraint violations show a more considerable peak-to-trough difference in magnitude than was produced using the PD formulation, as the integral gain contributed to larger constraint

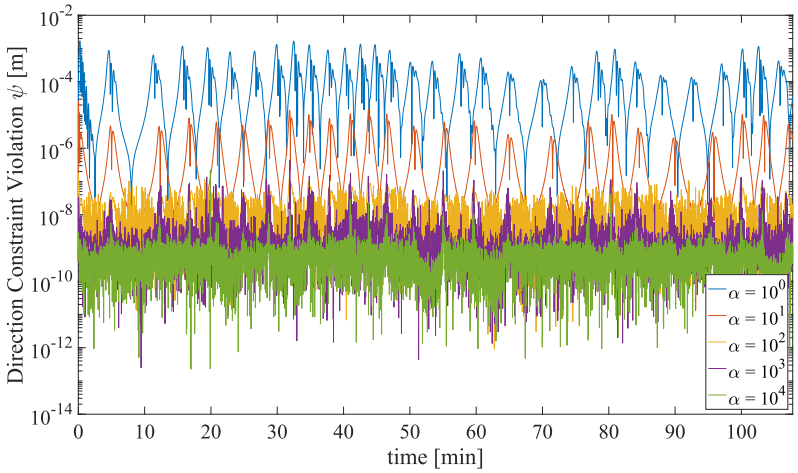


Figure 8. Length constraint violations with the PID formulation.

forces. This also produced a lower average violation using PID compared to PD results at each value of α but plateaued at the same order of magnitude around 10^{-9} meters via the same machine precision and integration limits as was observed in the PD gain analysis.

The direction constraint violations using the direction constraint for varying PID gains are shown in Figure 9. Consistent with previous results, the constraint violations decreased, and high-frequency

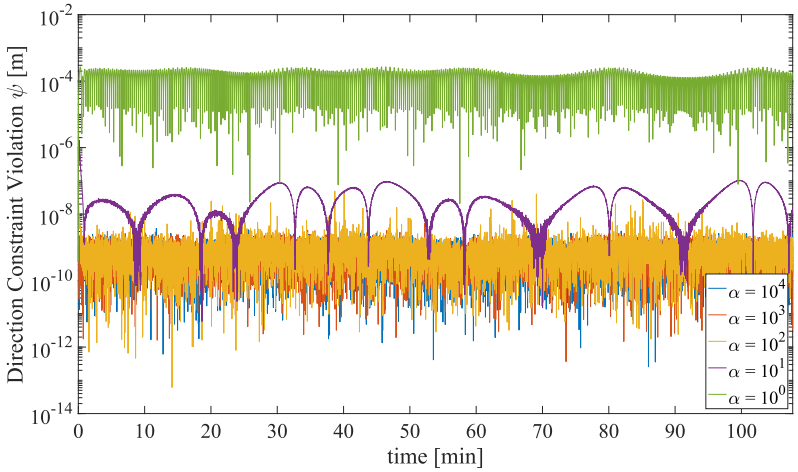


Figure 9. Direction constraint violations with the PID formulation.

noise increased with increasing values of α. However, this time the violations reached the plateau limit at $\alpha \geq 10^2$, an order of magnitude lower value for alpha than when the PD formulation reached the plateau. The oscillatory motion is also significantly smaller than observed with the PD formulation, except $\alpha = 10^1$. While it is unclear why this happens for this specific value, it might be because some fundamental modes are excited at just the right frequency by the constraint forces and torques.

The direction and attitude constraint violations for the rigid constraint implementation for varying

PID gains are shown in Figures 10 and 11, respectively. Using the rigid constraint, results were

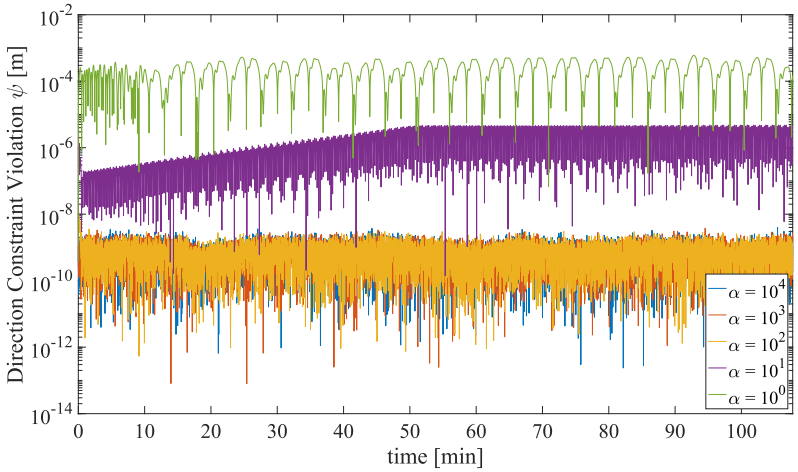


Figure 10. Rigid constraint direction violations with the PID formulation.

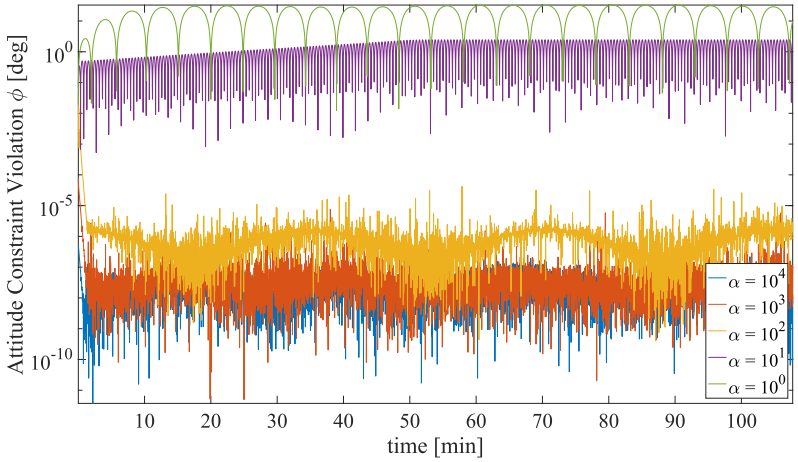


Figure 11. Rigid constraint attitude violations with the PID formulation.

again consistent with decreasing constraint violations and increasing high-frequency noise as alpha increased. As the other PID formulation results exhibited, the plateau is reached sooner than the corresponding PD performance in Figures 5 and 6. An interesting behavior arises for smaller values of α, which showed more significant attitude constraint violations at $\alpha \leq 10^1$ compared to the PD formulation. This can be attributed to the fact that lower gains do not adequately enforce the constraints. Adding another feedback term in the form of an integral term only makes that sensitivity more noticeable.

The runtime results are shown in Figure 12 for the PID formulation. As seen in the PD formulation, the runtime increases with increasing α, except for the lowest α tested. This is because sufficiently small PD gains allow more significant constraint violations, increasing the constraint force and driving the RK45 to smaller time steps. It should be noted that while the runtimes at each

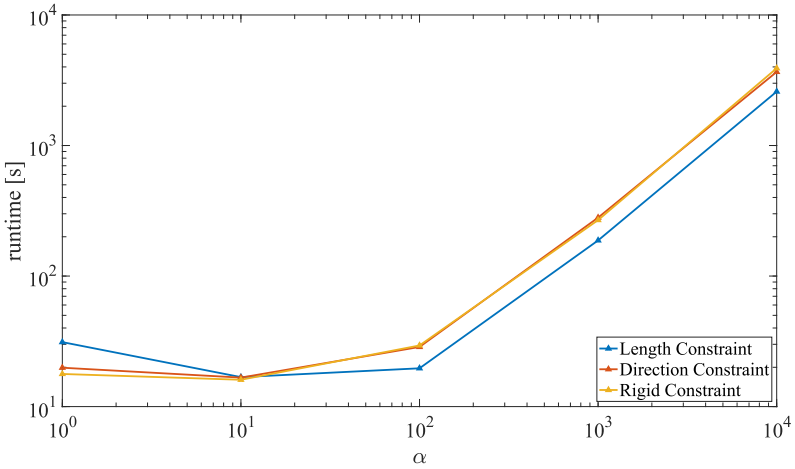


Figure 12. Runtime analysis for all constraint types using the PID formulation.

α using either the PID or PD formulations are almost identical, the constraint violation magnitudes for the PID formulation are significantly smaller at larger values of α for each constraint type. This shows that the addition of integral gain improves the performance of the constraint enforcement without simply contributing a larger constraint forces. If this were the case, the larger constraint forces would have made a stiffer dynamical system resulting in longer runtimes.

Maneuver Analysis

While analysis with varying sets of PD and PID gains shows that the method is effective and can yield adequate performance for the right gains, it has only considered the slowly moving motion of a nadir-pointing orbit. While this is an appropriate scenario, it does not stress-test the proposed approach for more demanding scenarios like when the spacecraft are maneuvering together. Therefore, the next step to assess performance is applying perturbations that stress the constraints. For this set of tests, the rigid constraint is selected using all proportional, derivative, and integral gains. It is the most realistic while also producing the lowest constraint violations in the previous gain analysis.

Orbital Maneuver The orbital maneuver adds a translational perturbation by applying a force on the servicer spacecraft in the radial direction defined by the Hill frame for spacecraft 1. The force, which simulates low-thrust electric propulsion, is turned on at minute 5 and lasts one minute. The resulting performance of the rigid constraint using the PID constraint formulation with this orbital maneuver is shown in Figures 13 and 14.

Before the orbital maneuver is commanded at minute 5, the constraint violations are consistent with the PID gain analysis results. As the perturbing force is applied, a constraint violation spike occurs which is smaller for larger values of α. Another smaller spike in constraint violation occurs when the perturbing force is turned off at minute 6, caused by a lag in the contribution from the integral gain. During the maneuver, the constraint violations hover at a value greater than their pre-maneuver value as the system goes through a dynamic simulation phase. After the maneuver is turned off, the constraint violations settle to a lower value.

It should be noted that the steady-state constraint violations after the maneuver are not as low

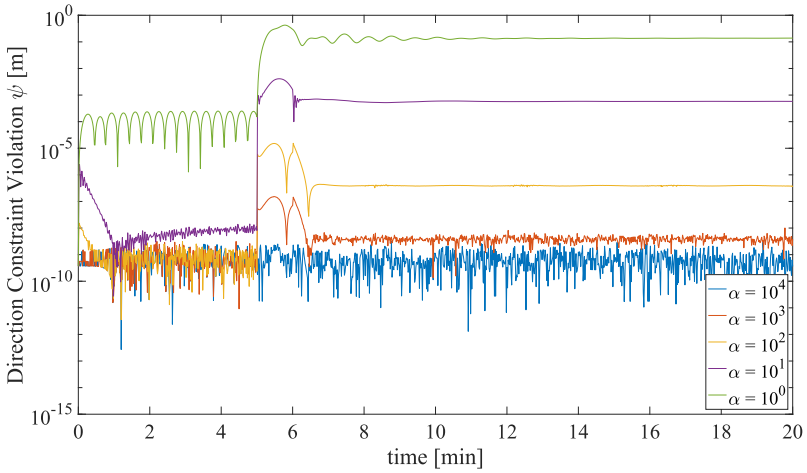


Figure 13. Rigid constraint direction violations with the translational maneuver.

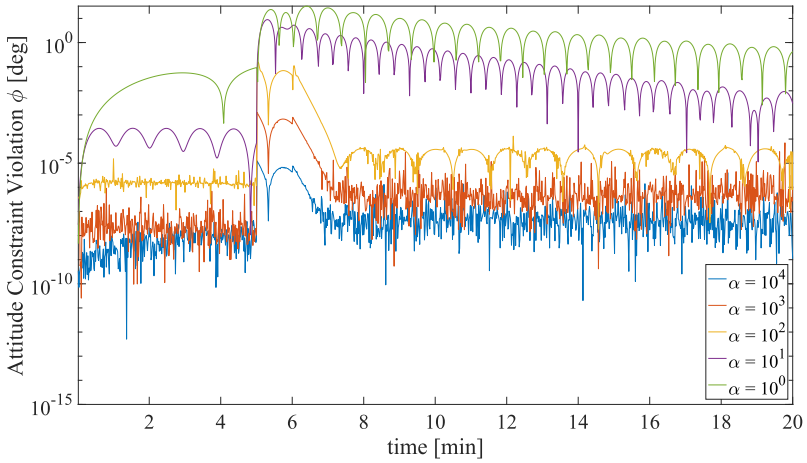


Figure 14. Rigid constraint attitude violations with the translational maneuver.

as pre-maneuver magnitudes. This is caused by the residual motion of the joint spacecraft system stressing the constraints slightly more than the slower-moving pre-maneuver joint system. New angular and translational rates of the attached spacecraft characterize this motion. Whereas the slow orbital motion in the PID gain analysis sees a plateau in direction constraint violations at $\alpha \geq 10^1$, for the maneuver scenario, this plateau is only reached at $\alpha = 10^4$. For the attitude constraint violations, this plateau still isn't reached at $\alpha = 10^4$ after the maneuver finishes.

These results show that the method can handle more realistic dynamic scenarios. While the constraint violations increase when the servicer is thrusting, the constraints are still met within reasonable accuracy, and the system can settle after the maneuver is done.

Attitude Maneuver The second test performs an attitude maneuver, applying a torque to the servicer spacecraft starting at minute 5 until it rotates it by $22.5°$. The resulting performance of the rigid constraint using the PID constraint formulation for the attitude maneuver is shown in Fig-

ures 15 and 16. Before the rotational maneuver is applied at minute 5, the constraint violations

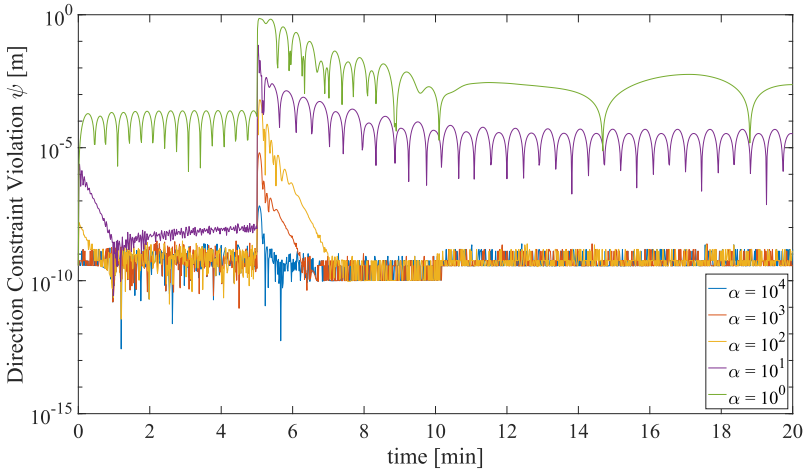


Figure 15. Rigid constraint direction violations with attitude maneuver.

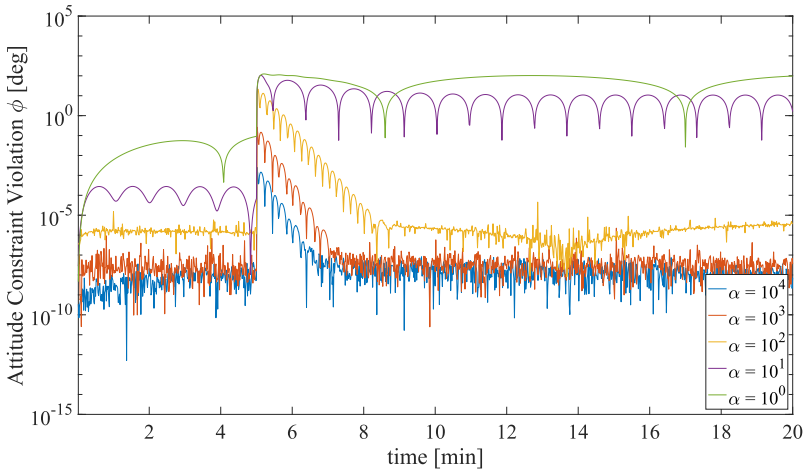


Figure 16. Rigid constraint attitude violations with attitude maneuver.

are consistent with the PID gain analysis results. As with the orbital maneuver, there is a constraint violation spike when the control torque is commanded. However, because this control settles to a new attitude of the joint spacecraft system rather than turning off at a future time, there is no second spike as was observed with the orbital maneuver. In addition, at increasing values of α, the constraint violations exhibit less high-frequency noise after the maneuver is applied because the noise is dampened by the control torque keeping the spacecraft system in a fixed attitude. Moreover, similar to the orbital maneuver, the post-orbital maneuver steady-state constraint violation doesn't settle to the pre-orbital maneuver levels in all cases.

A plot of each spacecraft's attitude expressed in modified Rodrigues parameter components during the rotational maneuver is shown in Figure 17 from the run using $\alpha = 10^2$. The dashed lines show the attitude of spacecraft 2, which has the same settling time as the attitude of spacecraft 1 in

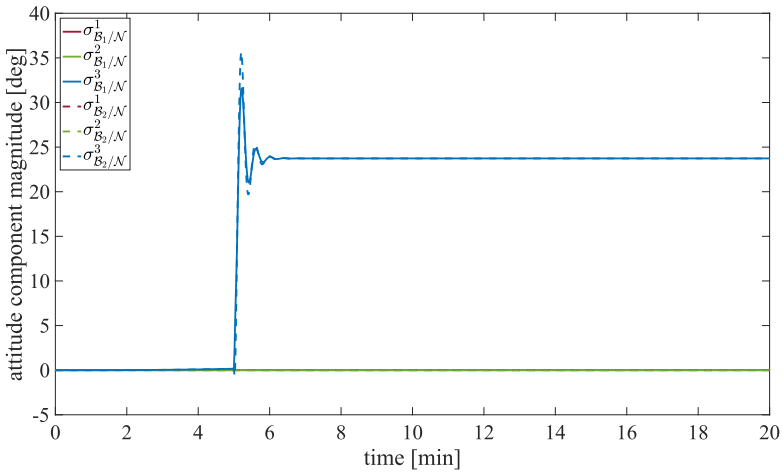


Figure 17. Attitude for both spacecraft with the rotational maneuver.

solid lines. Despite this behavior, spacecraft 2 has a more significant overshoot and oscillations in attitude caused by its lagging behind spacecraft 1. This lag occurs because the maneuver torque is applied exclusively to spacecraft 1 and only passed on to spacecraft 2 through the attitude constraint force. Nonetheless, both spacecraft settle to the intended reference attitude despite the attitude control being only directly applied to spacecraft 1, as they are translationally and rotationally locked together.

CONCLUSION AND FUTURE WORK

With spacecraft docking becoming increasingly significant in recent decades and even more interest on the horizon, properly simulating spacecraft in this scenario is essential to guarantee that mission objectives are met. This work provides a numerical approach to this problem, which aims to facilitate ease of early prototyping and mission analysis.

An analytical mathematical formulation is provided for the Bamgarte stabilization method to enforce the holonomic constraints of the system. The general problem is formulated to be general and applicable to different scenarios involving docked spacecraft. Multiple constraints are explored with increased rigidity to guarantee that the approach is sound for more complex, rigid, and increasingly realistic situations.

Gain selection is an essential element of the Baumgarte stabilization method, as it determines the performance and efficiency of the approach. The PD gain analysis shows that choosing increasingly larger gains improves performance at the cost of runtime until the results plateau. The PID formulation yields an order of magnitude better performance than the PD formulation with the same runtime.

Overall, the Baumgarte stabilization method has proved to be an effective way to simulate constrained spacecraft dynamics. The proper gain selection can accurately describe the behavior of the post-docked dynamics of two spacecraft in various scenarios, from coasting to orbital and attitude maneuvers. Future work includes exploring other control techniques, such as sliding mode control and relaxing the assumption that each spacecraft is rigid. Instead, the objective is to analyze realistic spacecraft with reaction wheels and bending solar panels, among other devices.

ACKNOWLEDGMENT

This research was supported through the Space Force STTR Phase I project in collaboration with Starfish Space.

REFERENCES

[1] A. Ellery, J. Kreisel, and B. Sommer, "The case for robotic on-orbit servicing of spacecraft: Spacecraft reliability is a myth," *Acta Astronautica*, Vol. 63, No. 5-6, 2008, pp. 632–648.

[2] R. L. Newman, "STS-61 mission director's post-mission report," tech. rep., 1995.

[3] N. T. Redd, "Bringing satellites back from THE DEAD: Mission extension vehicles give defunct spacecraft a new lease on life," *IEEE Spectrum*, Vol. 57, 2020, pp. 6–7.

[4] M. A. Shoemaker, M. Vavrina, D. E. Gaylor, R. Mcintosh, M. Volle, and J. Jacobsohn, "OSAM-1 decommissioning orbit design," *AAS/AIAA Astrodynamics Specialis t Conference*, 2020.

[5] B. B. Reed, R. C. Smith, B. J. Naasz, J. F. Pellegrino, and C. E. Bacon, "The restore-L servicing mission," *AIAA space 2016*, p. 5478, 2016.

[6] ESA, "ESA Purchases World-First Debris Removal Mission From Start-Up," January 2020.

[7] OrbitFab, "Rapidly Attachable Fluid Transfer Interface Technical Datasheet," `https://catalog.orbitaltransports.com/content/brands/orbitfab/RAFTIBrochure.pdf`, 2022.

[8] A. Rivolta, P. Lunghi, and M. Lavagna, "GNC & robotics for on orbit servicing with simulated vision in the loop," *Acta Astronautica*, Vol. 162, 2019, pp. 327–335, https://doi.org/10.1016/j.actaastro.2019.06.005.

[9] T. Rybus and K. Seweryn, "Manipulator trajectories during orbital servicing mission: numerical simulations and experiments on microgravity simulator," *Progress in Flight Dynamics, Guidance, Navigation, and Control–Volume 10*, Vol. 10, 2018, pp. 239–264.

[10] J. Baumgarte, "Stabilization of constraints and integrals of motion in dynamical systems," *Computer Methods in Applied Mechanics and Engineering*, Vol. 1, No. 1, 1972, pp. 1–16, https://doi.org/10.1016/0045-7825(72)90018-7.

[11] P. Flores, M. Machado, E. Seabra, and M. T. d. Silva, "A parametric study on the Baumgarte stabilization method for forward dynamics of constrained multibody systems," *Journal of computational and nonlinear dynamics*, Vol. 6, No. 1, 2011.

[12] U. M. Ascher, H. Chin, L. R. Petzold, and S. Reich, "Stabilization of constrained mechanical systems with daes and invariant manifolds," *Journal of Structural Mechanics*, Vol. 23, No. 2, 1995, pp. 135–157.

[13] B. Chen and Y. Geng, "Simulation of rendezvous and docking between service spacecraft and non-cooperative target with MATLAB-Simulink and STK," *Proceeding of the 11th World Congress on Intelligent Control and Automation*, IEEE, 2014, pp. 5727–5732.