An On-Line Global Search Approach to Underactuated Docking Operations Via Model Predictive Control and the Cross-Entropy Method

Anthony Aborizk[(1)], Alexander Soderlund[(2)], Norman Fitz-Coy[(1)]

[(1)] University of Florida
Gainesville, USA
Email: aborizk.anthony@ufl.edu

[(2)] Air Force Research Laboratory
Kirtland Air Force Base, USA

***Abstract* – The proliferation of small satellites in Earth's orbit has intensified the demand for onboard autonomy. While recent advancements have eased the financial burden of launching spacecraft, the weight of these vehicles remains a critical constraint. Furthermore, the occurrence of mission failures due to actuator faults has spurred significant research efforts. To address these challenges – including reducing the burden on ground control operators caused by the mass of on-orbit vehicles, mitigating fiscal constraints related to weight, and providing fault-tolerant solutions – this paper explores a novel approach to the underactuated docking problem. This problem serves as a benchmark within the field of autonomous rendezvous, proximity operations, and docking research. Specifically, the paper proposes a gradient-free nonlinear Model Predictive Control solution called Cross-Entropy Model Predictive Control to tackle issues associated with non-monotonicity observed in prior numerical solutions to this problem. By using a global search algorithm, the optimizer is more likely to bypass shallow minima.**

I. INTRODUCTION

Spacecraft mission design is advancing in complexity beyond the capabilities of the current paradigm. As more spacecraft are deployed and time sensitive maneuvers become increasingly critical to functionality, enabling forms of autonomy in guidance, navigation, and control (GNC) tasks becomes a critical research goal. The field of autonomous rendezvous, proximity operations, and docking (ARPOD) seeks to enable technologies by advancing efforts in areas such as on-orbit satellite servicing, refuelling, constellation management, and planning in both known and unknown dynamical environments. State-of-the-art satellite maneuvering control currently includes methods such as Model Predictive Control (MPC) and Mixed Integer Linear Program (MILP) [1]. However, these algorithms work best in uncluttered, static environments. Achieving success in the desired missions will require algorithms capable of GNC in environments containing fault tolerance, logical modes, time-varying constraints, and complex maneuvers with unknown dynamics [1]. The research herein provides a solution to an ARPOD problem with a focus on fault tolerance in common 6U CubeSats.

The problem of interest is that of underactuated spacecraft docking, which originated as a challenge problem posed by Petersen et. al. [2]. They presented a planar, underactuated 6U CubeSat capable of orienting itself about the angular momentum vector and translating about its unilateral longitudinal axis. In response to this problem, Soderlund and Phillips [3, 4] showed that the linearized dynamics are neither controllable nor stabilizable at the origin–ruling out the use of many classic feedback approaches. Soderlund and Phillips went on to fully characterize the problem as a nonlinear control-affine system with drift and offered a rigorous proof to guarantee local asymptotic stability to the objective state. Two additional solutions to the planar underactuated docking problem have been found by Zaman et. al. [5] using MPC and Paris [6] using Proximal Policy Optimization under the presence of many constraints detailed in [2]. The problem was then expanded to 3 dimensions by Aborizk and Fitz-Coy [7] in which thrust remained constrained along a unilateral body-fixed axis, but attitudinal control was expanded to all three principal axes. A solution was presented to a constrained formulation of the problem using Legendre-Gauss-Radau direct collocation. The research herein expands these efforts by developing a real-time solution to a tighter tolerance with the capability of handling complex constraints.

To accomplish this, an MPC algorithm based on works by Nagabandi et. al. [8] , Zhang et. al. [9], and Aborizk and Fitz-Coy [10] is utilized with the aim of controlling an underactuated satellite to a docking state using a zeroth-order optimizer, the cross-entropy method (CEM), and a quadratic cost. It is a consolidation of MPC techniques that offer stability properties, global optimality, and enable the algorithm to handle complex objectives. This method will henceforth be referred to as CEMPC and is shown to be a viable, real-time solution to the 2D underactuated docking problem.

II. PRELIMINARY

A. Notation

Vectors will be denoted as bolded variables; for example, $\mathbf{x}$. Unit vectors will use the hat subscript, $\hat{\boldsymbol{x}}$. An estimated value will be denoted with a tilde on top, $\tilde{\boldsymbol{x}}$. Matrices are denoted using bolded capital letters, $\mathbf{X}$, and sets will be defined using calligraphic capital letters, $\mathcal{X}$. The optimal value of a quantity is denoted with an asterisk superscript, x^*. The row index of a matrix is denoted as a parenthetical superscript, $\boldsymbol{X}^{y}$. The Euclidean inner product, or dot product, of two vectors will be denoted using $< \boldsymbol{x}, \boldsymbol{y} >$.

B. Model

This study looks at two spacecraft: an uncontrolled cooperative chief in a circular low-Earth orbit and a controlled, but underactuated, 6U CubeSat deputy in Hill's reference frame. If we assume the chief is in a perfectly circular orbit and that perturbations are negligible, the Hill-Clohessy-Whiltshire (HCW) equations allow us to linearly represent the relative dynamics between the two spacecraft with respect to the chief's frame of reference. This frame is defined by a basis $\mathcal{O} := \{\hat{\boldsymbol{x}}_\mathbf{o}, \hat{\boldsymbol{y}}_\mathbf{o}, \hat{\boldsymbol{z}}_\mathbf{o}\}$, and is fixed to the center of mass of the chief body. The deputy's body-fixed frame is expressed as $\mathcal{D} := \hat{\boldsymbol{x}}_\boldsymbol{d}, \hat{\boldsymbol{y}}_\boldsymbol{d}, \hat{\boldsymbol{z}}_\boldsymbol{d}$ which is aligned with, but separate from, the frame $\mathcal{O}$. These frames are depicted in Figure 1 along with an Earth-centered inertial (ECI) frame with basis, $\mathcal{E} := \{\hat{\boldsymbol{x}}_\boldsymbol{e}, \hat{\boldsymbol{y}}_\boldsymbol{e}, \hat{\boldsymbol{z}}_\boldsymbol{e}\}$.

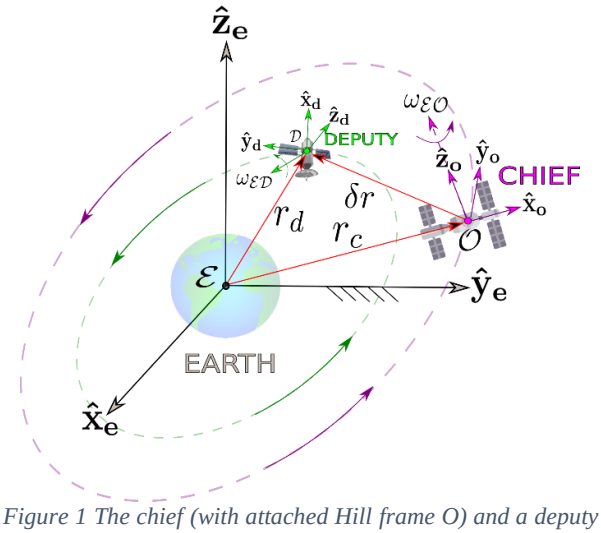


Figure 1 The chief (with attached Hill frame O) and a deputy (with body-fixed frame D) are orbiting about the Earth with inertial frame E. The dashed lines correspond to the closed orbital trajectories of both craft [3].

It is important to note that there are limitations to this approach. Space is nonlinear and the HCW representation runs the risk of error propagation. However, in this case, error is proportional to the distance from the chief. In low Earth orbit, this only becomes an issue when relative distances range on the order of tens of kilometres, which the proposed simulation does not [11]. We assume that the error induced by linearization will be negligible. Additionally, given the complexity of the controller presented herein, only 2 degrees of freedom are analysed to facilitate early-stage design. However, the realism is reduced in two dimensions as z-axis effects and orientation can be significant for docking maneuvers.

The model used for the deputy in this analysis, depicted in Figure 2, is a 6U CubeSat measuring 30 cm x 20 cm x 10 cm along the $\hat{\boldsymbol{x}}_\boldsymbol{d}$-, $\hat{\boldsymbol{y}}_\boldsymbol{d}$-, and $\hat{\boldsymbol{z}}_\boldsymbol{d}$- axes, respectively. The spacecraft can produce thrust unilaterally along the $\hat{\boldsymbol{x}}_\boldsymbol{b}$-axis (positive and negative). Thrusts are assumed to act through the center of mass of the spacecraft. The attitude is controlled by a flywheel rigidly attached to the $\hat{\boldsymbol{z}}_\boldsymbol{d}$-axis of the deputy. A gimbaled sensor is also attached to this axis to prevent rotation about any other axes. This frame notation and the graphics depicted in Figure 1 and 2 are taken from [3] and [2], respectively.

C. Equations of Motion

Let the control-input be defined as $U := \left(F_x, \dot{\psi}\right)^T$ where $\dot{\psi}$ represents the angular acceleration of the flywheel. The variable F_x is the thrust force through the $\hat{\boldsymbol{x}}_\boldsymbol{d}$-axis. The state space is defined as

$$\boldsymbol{x} := (\delta x, \delta y, \theta, \delta \dot{x}, \delta \dot{y}, \omega)^T$$

where $\boldsymbol{x} \in \mathbb{R}^6$, δx, and δy are the relative position values, $\delta \dot{x}$, and $\delta \dot{y}$ are the relative velocities, θ is the orientation of the deputy, and ω is its angular velocity components. The continuous time-invariant Clohessy-Wiltshire equations from [12] are given by

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}(\theta)\boldsymbol{u} \tag{1}$$

where

$$\boldsymbol{A} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3\eta^2 & 0 & 0 & 0 & 2\eta & 0 \\ 0 & 0 & 0 & -2\eta & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\boldsymbol{B}(\theta) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \dfrac{\cos(\theta)}{m_c} & 0 \\ \dfrac{\sin(\theta)}{m_c} & 0 \\ 0 & \dfrac{-D}{I_z} \end{pmatrix}.$$

The mean motion constant is $\eta = \sqrt{\frac{\mu}{{r_c}^3}} =$ 0.001027 rad/s and represents the average angular rate of the chief spacecraft. The variable $\mu = 398600.4418$ km^3/s^2 is the Earth's standard gravitational parameter and r_c is the radius of the chief's circular orbit.

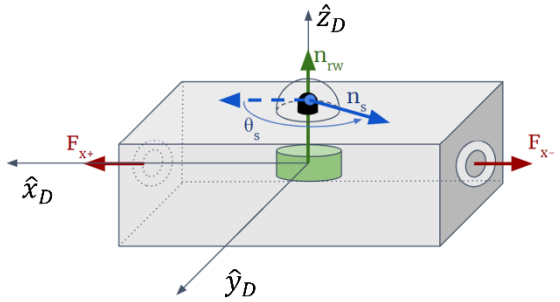


Figure 2 Depiction of the modelled deputy and its body-fixed frame D. Thrust actuation (fx) is aligned along the positive and negative $\widehat{x_d}$ −axes [2].

The flywheel is assumed to be at the center of mass of the spacecraft with rotations about the $\widehat{z_d}$-axis. The moment of inertias of the flywheel and spacecraft are denoted as D and I_z, respectively. Combining the control variables with B produces the rotational equations of motion

$$\ddot{\theta} = \frac{-D\dot{\psi}}{I_z}.$$

In these underactuated dynamics, the rotation and translation are coupled, thus precluding many classical control approaches (such as those utilizing linearization) to be applicable. This coupling is apparent in the control. From (1), since thrust is only applied along the $\widehat{x_d}$-axis, the relationship becomes

$$f_{xy} = \begin{bmatrix} cos(\theta) \\ sin(\theta) \end{bmatrix} f_x$$

where actuation in any direction is dependent on the orientation of the deputy. $\boldsymbol{f}_{xy}$ represents the radial form of the thrust component. To achieve a successful docking configuration, the deputy must reach a set of predefined relative translational and attitudinal states. The objective is to drive the deputy's translational and attitudinal states to the set $\mathcal{A} \coloneqq \boldsymbol{x} \in \mathcal{X}, u \in \mathcal{U} | \boldsymbol{x} = \boldsymbol{0}_6$. $\boldsymbol{x}$ goes to 0 as t goes to infinity.

D. CEMPC

Model Predictive Control (MPC) is an advanced control strategy used in various engineering applications to optimize the performance of dynamic systems while satisfying constraints. Unlike traditional control methods, MPC employs a predictive model of the system to anticipate future behaviour and compute control actions over a finite time horizon. At each time step, MPC solves an optimization problem to determine the optimal control inputs that minimize a predefined cost function, subject to system dynamics and constraints. By considering future predictions and incorporating feedback, MPC enables the control of complex processes with non-linear dynamics, uncertain disturbances, and constraints on both states and inputs. This predictive nature allows MPC to handle time-varying systems and disturbances effectively, making it widely utilized in the space industry. MPC offers a versatile and powerful framework for achieving optimal control performance in dynamic systems.

A gradient-free MPC called the Cross-Entropy Model Predictive Control (CEMPC) algorithm is used to solve the proposed problem. It follows the form

$$\boldsymbol{C}^* = arg \min_{\{\boldsymbol{A}^{(0)},\ldots,\boldsymbol{A}^{(K-1)}\}} \sum_{t'=t}^{t+H-1} r(\hat{\boldsymbol{s}}_{t'}, \boldsymbol{a}_{t'})$$

$$s.t.\ \hat{\boldsymbol{s}}_{t'+1} = \hat{\boldsymbol{s}}_{t'} + f_\theta(\hat{s}_{t'},\ \boldsymbol{a}_{t'})$$

At every time step, t, a sequence of actions is computed up to some horizon time H. To provide a thorough search of the surrounding state space, K sequences are analyzed. $\boldsymbol{C}^{(k)}$ is one of K sequences of actions each with a length equal to H. r is the reward, or cost, function associated with each state-action pair. In the scope of this paper, this function was chosen to be a QR cost due to its stabilizing properties. $\boldsymbol{C}^*$ represents the approximate optimal trajectory over the given horizon.

Control inputs are determined via the CEM, a gradient-free optimizer. Per [13], the CEM produces a global search of the state space during optimization. While this does not always guarantee convergence to a global minimum, it does typically lend itself well to bypassing shallow local optima, a necessary trait for many path planning problems. By iteratively refining a set of candidate solutions based on their performance and updating the distribution from which new solutions are sampled, CEM efficiently explores the search space and converges towards promising solutions.

It works by selecting action sequences corresponding to the J highest-scoring $\boldsymbol{C}^{(k)}$ values and utilizes them to compute a multivariate mean, μ, and covariance, Σ, which represent the means and variances of the pseudo-optimal trajectory distribution over the given horizon. The chosen sequences are referred to as the elites. To find the parameters of the trajectory distribution, the values acquired from random shooting are updated M times per prediction horizon. The updated laws are

$$\boldsymbol{\mu}_{t'}^{m+1} = \alpha * mean(\boldsymbol{A}_{elites}) + +(1-\alpha)\boldsymbol{\mu}_{t'}^{m}$$

$$\boldsymbol{\Sigma}_{t'}^{m+1} = \alpha * variance(\boldsymbol{A}_{elites}) + (1-\alpha)\boldsymbol{\Sigma}_{t'}^{m}$$

where α serves as a hyperparameter governing smoothness, and m ranges from 0 to M. The values of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ at the initial step in the finite horizon trajectory distribution are then utilized for sampling a multivariate Gaussian distribution, thereby generating an action for implementation in simulation. This process is iteratively carried out at each control step until the termination conditions are met. Through incremental optimization, the propagation of errors stemming from the approximate dynamics model for long-term planning is mitigated. It is worth noting the challenge in establishing theoretical stability guarantees, given the inherent black-box nature of this method which stems from its lack of gradient information. Nevertheless, the research herein attempts to stave off these uncertainties with use of a QR cost. Additionally, the notion of stability is bolstered through numerical Monte Carlo demonstrations.

III. Problem Statement

The goal of this research is to dock the underactuated chaser using a CEMPC algorithm. Table I lists the mission parameters used in this analysis. The termination conditions for the docking phase occur when $|x| \leq [0.1, 0.1, 0.035, 0.2, 0.2, 0.017]$. Additionally, a constrained action space is used to simulate control input saturation limits. The goal of the MBRL algorithm is to drive the underactuated spacecraft to within the docking constraints as time goes to infinity without violating any state or action bounds. Situations such as plume impingement or even damage to the chief can occur in configurations where thrusters are pointed at the chief during docking. So, to maintain realism, the docking port is located on the positive $\widehat{\boldsymbol{y_b}}$ face of the spacecraft, as depicted in Figure 3. While this configuration helps to reduce the risk of damage induced by the chaser's thrust, it creates a docking procedure that is difficult to control. Since thrust only exists unilaterally along the $\hat{\boldsymbol{x}}_b$-axis, the spacecraft must perform a V-bar maneuver without the ability to thrust directly towards or away from the chief spacecraft.

To determine the feasibility of this new docking configuration, a test case was designed. The initial conditions were set such that the deputy was positioned -3 m behind the chief, along the $\widehat{\boldsymbol{y_O}}$-axis, with zero relative velocity. The relative angular displacement as measured from the $\widehat{\boldsymbol{x_O}}$-axis was set to $\frac{\pi}{2}$. At this orientation $\widehat{\boldsymbol{y_d}}$ and $\widehat{\boldsymbol{x_O}}$ are antiparallel with the deputy's docking point facing the Earth and the chief's facing the deputy. The objective and constraints considered in this analysis include:

$$\min_{\boldsymbol{x},\boldsymbol{u}} \Sigma_{i=1}^{H-1} \boldsymbol{x}_i^T \boldsymbol{Q} \boldsymbol{x}_i + \boldsymbol{u}_i^T \boldsymbol{R} \boldsymbol{u}_i + \Sigma_{j=1}^{N} \boldsymbol{w}_j \mathbb{I}(\boldsymbol{x}_i) + \boldsymbol{x}_H^T \boldsymbol{P} \boldsymbol{x}_H$$

$$\text{S.T. } \mathrm{f_x} \in [-1,1]N$$

$$\dot{\psi} \in [-181.3, 181.3] \frac{rad}{s}$$

$$y \sin\left(\frac{\alpha}{2}\right) + x \cos\left(\frac{\alpha}{2}\right) \geq 0$$

$$y \sin\left(\frac{\alpha}{2}\right) - x \cos\left(\frac{\alpha}{2}\right) \geq 0$$

$$|v| \leq 10 \frac{m}{s}$$

Table I: List of Mission Parameters

Value	Description
$m = 12$ kg	Mass of the chaser
$\mu = 3.986 \times 10^{14} \frac{m^3}{s^2}$	Earth's gravitational parameter
$\eta = 0.001027 \frac{rad}{s}$	Mean motion of the Chief
$\lvert v_{dock} \rvert \leq [0.2, 0.2] \frac{m}{s}$	maximum docking velocity
$\lvert x_{dock} \rvert \leq [0.1, 0.1] m$	maximum distance to dock
$\lvert \theta_{dock} \rvert \leq 2^\circ$	maximum angular displacement
noise=$(1cm)^2$	magnitude of noise
t = 2 s	timestep

Where $\dot{\theta}$ and $\ddot{\theta}$ are the angular velocity and acceleration of the chaser, respectively, |v| represents the safety bounds on velocity, and the remaining two trigonometric inequality constraints form a "keep-in" zone. This zone takes the shape of a cone emanating from the $-\widehat{\boldsymbol{y_O}}$ side of the target. The idea is to reduce the available space to explore as the chaser gets closer to the docking configuration. The variable $\boldsymbol{w}_j$ represents the weight associated with each constraint, j. Similarly, and $\mathbb{I}(\boldsymbol{x}_i, \boldsymbol{u}_i)$ is an indicator function designed to signal when a control or state has violated a defined boundary. These are considered soft constraints, however, the penalty is obtained immediately when the state falls within the specified set. Additionally, the CEMPC structure allows the controller to observe things ahead of time.

A timestep of 2 seconds was used for propagating the dynamics with control input every 10 seconds. The predicted length was 25 s. The control horizon was designed such that the input would be considered as a piecewise continuous action across the prediction horizon. Finally, a QR cost was used with Q = $I_6[10,10,0.1,0.1,0.1,0.1]^T$ and $R = I_2[1,0.1]^T$.

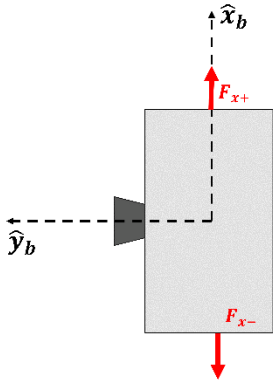


Figure 3 A depiction of the docking port location.

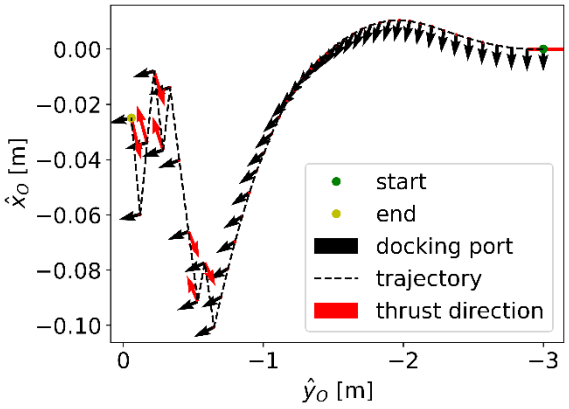
IV. Results

A. *Close Proximity Analysis*

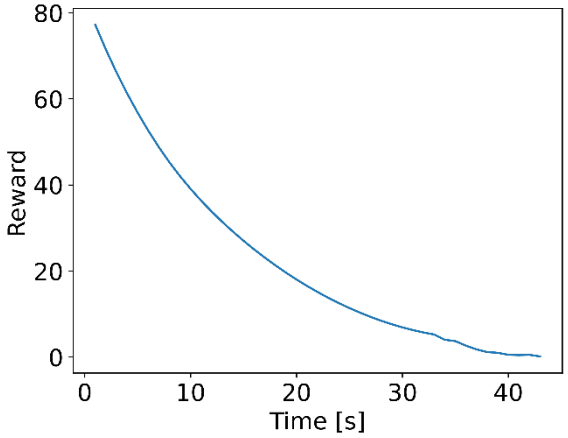
Given that the linearized dynamics are neither controllable nor stabilizable near the origin, and prior attempts at solving this problem numerically have had trouble here, the CEMPC was first tested in this region. The resulting trajectory generated from the CEMPC algorithm is depicted in Figure 4. A prevailing issue experienced in previous efforts to solve the underactuated docking problem numerically centralized around the nMPC from [5] producing non-monotonic cost, especially near the docking region. This is indicative of instability. However, with the use of the CEM this issue was avoided as depicted in Figure 4 b). Additionally, the solver was able to converge to a much tighter tolerance (i.e., to within 0.1 m of the origin instead of 0.5 m) when given the conic state space constraint. While in the presence of nonlinearities such as the one seen in this case study, monotonicity of the cost does not strictly guarantee stability, but it is a neccessary condition for it.

The deputy is initially oriented such that its thrusters are facing towards and away from the chaser. From the remaining figures in Figure 4, the following trajectory is observed: a small positive thrust is exerted and the spacecraft coasts towards its target. Meanwhile, control is being applied to the flywheel to slowly orient the docking port correctly. This coasting effect reduces the relative distance to approximately 1.5 m. By this time the chaser has drifted radially outward as seen in Figure 4 a). It then rapidly relocates itself below the chief (i.e., close to the Earth) before it begins an orbit transfer to achieve the final docking configuration. The entire trajectory was accomplished well within the desired velocity range.

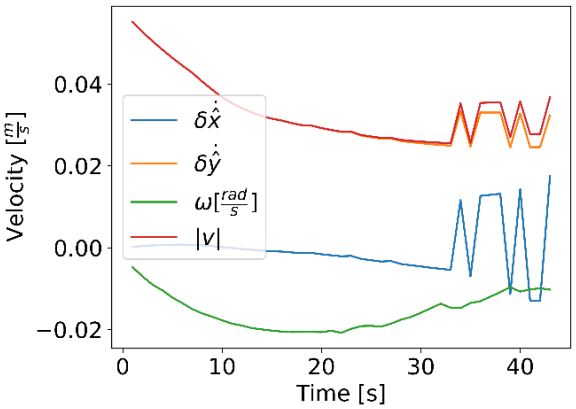
While precise trajectories are achievable with this method, issues arise with robustness. For this problem, the CEMPC is sensitive to initialization of the state space, thus proper alignment of the chaser is needed before beginning the final approach.



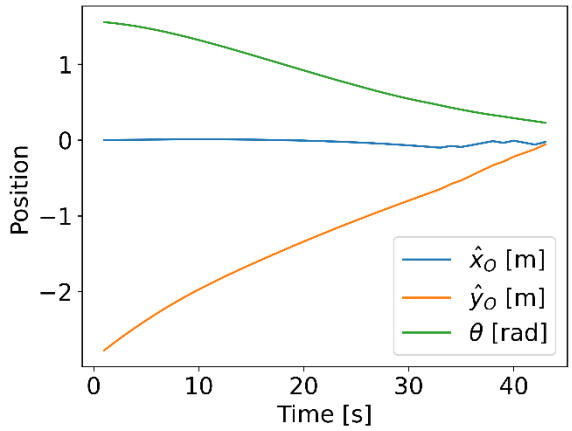
a) Deputy trajectory, orientation and thrust direction



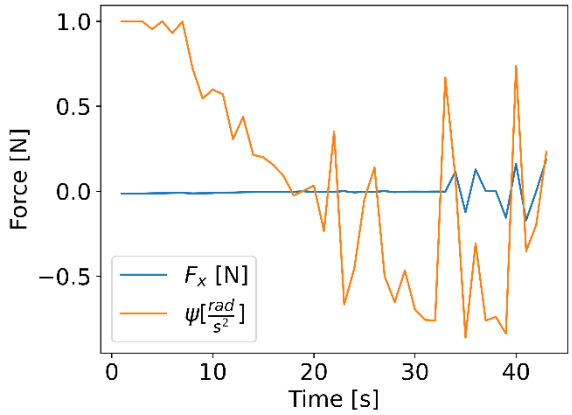
b) The reward function over time



c) Velocity of the deputy in the relative x- and y-axes, as well as the angular velocity of deputy. The speed of the deputy is also provided.



d) Convergence of the position and angular displacement of the deputy to the docking region.



e) The control input throughout the trajectory.

Figure 4 Results generated from the feasibility test.

B. *Repeatability*

To test the reproducibility of the trajectory, the random seed controlling the optimizer initialization was removed and a series of 25 trajectories were generated from the same initial conditions. Figure 7 highlights the solver's ability to dock consistently regardless of optimizer initialization.

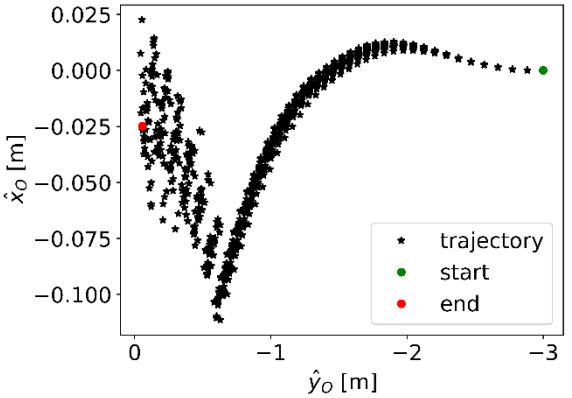
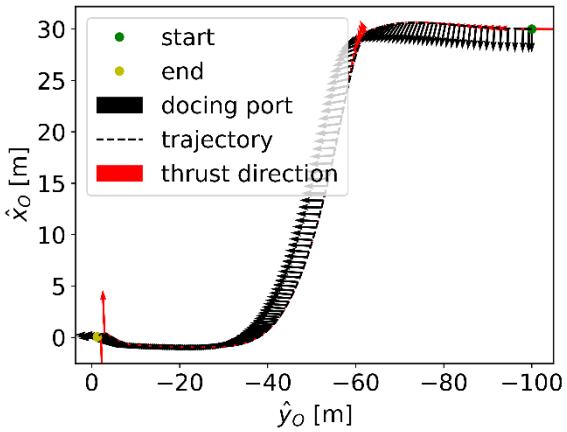


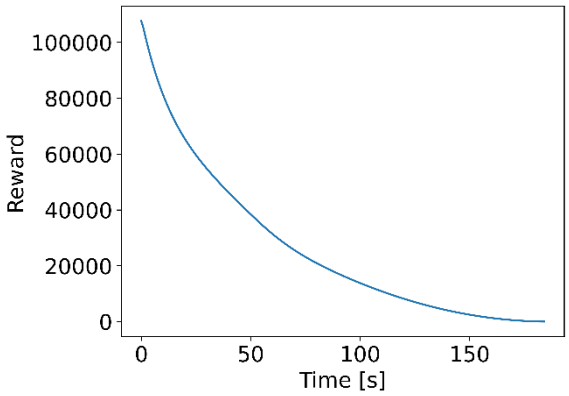
Figure 7 To test the sensitivity of the CEM to changes in initialization, 25 trajectories were generated without seeding.

C. *Increased range*

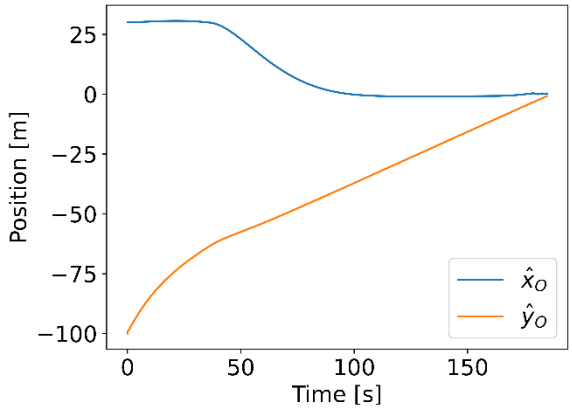
To see if the solver could handle converging from a farther distance, the deputy spacecraft was initialized at $\boldsymbol{x} =$ [30, -100, π/2, 0, 0, 0]. Figure 8 depicts the trajectory information. Without any user input the solver appears in Figure 8 a) to rush the deputy to the V-bar axis before performing tight corrective maneuvers near the target. From Figure 8 d), the desired orientation for docking is achieved in the first 75 seconds of the simulation, after which the deputy relies almost entirely on thrusts along the $\widehat{\boldsymbol{x_O}}$ −axis to control positioning. Again, Figure 8 b) shows a monotonically decreasing cost function.



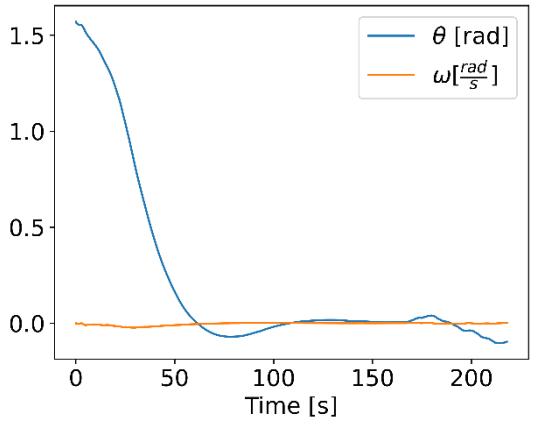
a) Deputy trajectory, orientation and thrust direction



b) The reward function over time



c) Convergence of the position and angular displacement of the deputy to the docking region.



d) The angular displacement and angular velocity of the deputy.

Figure 8 Results generated after increasing the distance between spacecraft.

V. Discussion and Future Work

This paper offers a solution to the underactuated docking problem via the use of a nonlinear Model Predictive Control scheme with a QR cost, soft constraints, and gradient-free optimizer known as the cross-entropy method. This work contributes to existing efforts to solve this benchmark problem by presenting an on-line, numerical solution with the necessary conditions for asymptotic stability; a monotonically decreasing cost function. The CEM produces a global search of the state space during optimization. While this does not always guarantee convergence to a global minimum, it does typically lend itself well to bypassing shallow local optima, a necessary trait for many path planning problems.

Given the known abilities of the CEM as a global search algorithm, the fact that it can produce a monotonically decreasing cost where the previous gradient-based optimizer failed implies that solutions to this problem exist on a non-smooth cost surface. The initialization of gradient-based methods in the future should be handled carefully. Additionally, while robustness to initial conditions was not demonstrated in this work, offline solutions presented in [7] show that they are possible with the use of collocation. Future work will explore the integration of a collocated reward function into the on-line problem to address this topic.

While the precision of the CEM appears to diminish with the increase in relative distance between spacecraft, it is important to note that many "failed" trajectories passed through a much tighter docking region, but were simply not sampled during their time spent there. In other words, the timestep of the solver was too large. The solution, however, would not just involve decreasing the timestep. Realistically, satellites only produce a command output every 2 – 30 seconds in close proximity missions such as these. So, the timestep of the simulator would need to be adjusted to "catch" the deputy docking. Future work will explore the application of CEMPC to the 6 degrees-of-freedom problem outlined in [7] and will explore the addition of a neural network based estimation of the dynamics to allow system identification, as seen in [10].

VI. Acknowledgments

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE-1842473. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

The views expressed are those of the author(s) and do not necessarily reflect the official policy or position of the Department of the Air Force, the Department of Defense, or the U.S. government. Approved for public release; distribution is unlimited. Public Affairs release approval #AFRL-2024-2013

VII. References

[1] M. Pavone, B. Acikmese, I. A. Nesnas and J. Starek, "Spacecraft Autonomy Challenges for Next Generation Space Missions," in *Advances in Control System Technology for Aerospace Applications*, Berlin, Heidelberg, Springer Berlin Heidelberg, 2016, pp. 1-48.

[2] C. D. Petersen, S. Phillips, K. L. Hobbs and L. Kendra, "CHALLENGE PROBLEM: ASSURED SATELLITE PROXIMITY OPERATIONS," in *AAS/AIAA Astrodynamics Specialist Conference*, Big Sky, 2021.

[3] A. Soderlund and S. Phillips, "Hybrid Systems Approach to Autonomous Rendezvous and

Docking of an Underactuated," *Journal of Guidance, Control, and Dynamics,* vol. 0, no. 0, pp. 1-18, 2023.

[4] A. Soderlund, S. Phillips, A. Zaman and C. D. Petersen, "Autonomous Satellite Rendezvous and Proximity Operations via Geometric Control Methods," in *spaceflight mechanics*, virtual, 2021.

[5] A. Zaman, A. Soderlund, C. Petersen and S. Phillips, "Autonomous Satellite Rendezvous and Proximity Operations via Model Predictive Control Methods," in *AAS/AIAA Astrodynamics Specialist Conference*, Big Sky, 2021.

[6] M. Paris, "Safe ARPOD for under-actuated CubeSat via Reinforcement Learning," Politecnico Milani 1863 School of Industrial and Information Engineering, 2021.

[7] A. Aborizk, N. Fitz-Coy and, A. Soderlund, "3d Underactuated Spacecraft Docking Using Legendre Gauss Radau Collocation," in *IEEE Aerospace Conference*, Big Sky, 2024.

[8] A. Nagabandi, G. Kahn, R. Fearing, and S. Levine, "Neural Network Dynamics for Model-Based Deep Reinforcement Learning with Model-Free Fine-Tuning," *CoRR,* vol. 1708.02596, 2017.

[9] D. Zhang, S. Song, and R. Pei, "Safe Guidance for Autonomous Rendezvous and Docking with a Non-cooperative Target," *AIAA guidance, navigation, and control conference*, p. 7592, 2010.

[10] A. Aborizk and N. Fitz-Coy, "Multiphase Autonomous Docking via Model Based and Hierarchical Reinforcement Learning," *Journal of Spacecraft and Rockets,* vol. 0, no. 0, pp. 1-13, 2024.

[11] W. Fehse, Automated Rendezvous and Docking of Spacecraft, 1st ed., Cambridge University Press, 2008, p. 41.

[12] W. Clohessy and R. Wiltshire, "Terminal guidance system for satellite rendezvous," *Journal of the Aerospace Sciences,* vol. 27, no. 9, pp. 653-658, 1960.

[13] M. Kobilarov, "Cross-entropy motion planning," *The International Journal of Robotics Research,* vol. 31, no. 7, pp. 855-871, 2012.