Nonlinear Model Predictive Control for Spacecraft Rendezvous and Docking with a Rotating Target

AAS 17-496

NONLINEAR MODEL PREDICTIVE CONTROL FOR SPACECRAFT RENDEZVOUS AND DOCKING WITH A ROTATING TARGET

Hyeongjun Park,[*] Richard Zappulla II,[†] Costantinos Zagaris,[‡] Josep Virgili-Llop,[*] and Marcello Romano[§]

In this paper, we develop a nonlinear model predictive control (NMPC) approach for spacecraft rendezvous and docking (RVD) with a rotating target platform. A strategy to enforce and handle constraints is proposed for collision-free and soft docking while real-time computation is achieved. In the strategy, constraints on thrust, spacecraft positioning within the entry cone from the docking port, and collision avoidance are systemically treated and switched in two-phase spacecraft RVD maneuvering. Dynamically reconfigurable constraints with a switching algorithm are introduced to guide the chaser spacecraft into the docking port in the final docking phase. The performance of the developed NMPC controller is analyzed for test cases using a MATLAB/Simulink-based simulator. The controller is also implemented on an air-bearing test bed to demonstrate the capability to perform real-time computation and to satisfy constraints.

INTRODUCTION

Autonomous spacecraft rendezvous and docking (RVD) with a rotating/tumbling platform is necessary and critical in specific space missions. For instance, it is required to recover an out-of-control satellite and to dock with freely tumbling spacecraft due to fuel depletion.[1] Debris removal missions are also examples for spacecraft RVD with a tumbling object. There is a growing interest in autonomous RVD with a rotating/tumbling target as future space missions require autonomous inspection, recovery, and removal capabilities.

Several approaches to solve the autonomous RVD with a rotating/tumbling object have been proposed in literature. Nolet *et al.*[1,2] proposed an algorithm based on the glideslope method for autonomous docking with a freely tumbling target. This proposed guidance and control method was experimentally validated using both the SPHERES terrestrial and on-orbit test bed. Boyarko *et al.*investigated the minimum-time and minimum-energy optimal control necessary to rendezvous and dock with a tumbling object. The six degree-of-freedom (DoF) docking model of a two-spacecraft rendezvous maneuver was used. Minimum-quadratic-control and minimum-time problems were formulated and addressed using a pseudospectral solver. Ventura *et al.*[3,4] proposed a guidance method based on an inverse dynamics approach. In this method, both rotational and translational trajectories of the chaser spacecraft are parameterized using high-order polynomials. These parameterized trajectories were then utilized to solve a minimum-energy rendezvous and docking

[*]Postdoctoral Research Associate, Mechanical and Aerospace Engineering, Naval Postgraduate School, Monterey, CA 93943.
[†]Ph.D. Candidate, Mechanical and Aerospace Engineering, Naval Postgraduate School, Monterey, CA 93943.
[‡]Ph.D. Student, Mechanical and Aerospace Engineering, Naval Postgraduate School, Monterey, CA 93943.
[§]Associate Professor, Mechanical and Aerospace Engineering, Naval Postgraduate School, Monterey, CA 93943.

manuever between a controlled chaser spacecraft and an uncontrolled target. Lastly, an experimental evaluation of the performance of an inverse dynamics guidance and control strategy for the planar RVD with a rotating target was performed by Wilde *et al.*[5] The experimental results demonstrate the inverse dynamics method utilized provides robust performance within a certain range of target rotational rates.

The requirements for autonomous spacecraft RVD with a rotating target lead to control problems with imposed pointwise-in-time and terminal constraints on both the state and control variables. Model Predictive Control (MPC) approaches have been used to treat the control problems with the constraints effectively. For example, a linear quadratic MPC approach[6] has been proposed for spacecraft RVD to a rotating platform in addition to debris avoidance maneuvers.[7,8] However, the ability to solve the resulting optimization problems of MPC in real time is a crucial requirement. This requirement is challenging to achieve, but often not considered explicitly.[9] The inability to complete the computations in real time can result in loss of stability, delay in the control execution, effective loss of the sampling time, and degraded performance.[9,10]

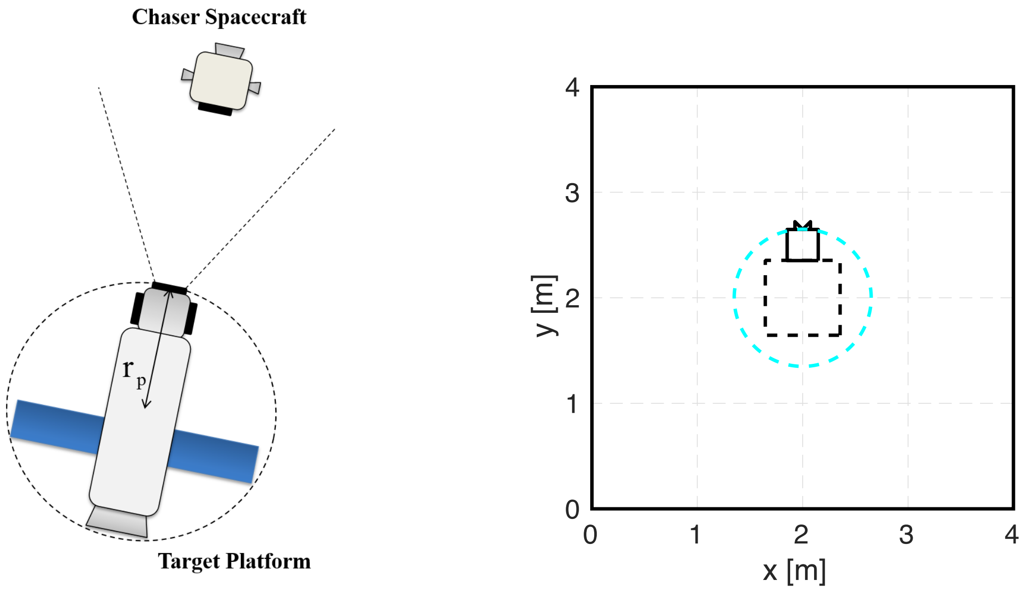
The main objective of this paper is to demonstrate the real-time efficacy of nonlinear MPC (NMPC) coupled with a strategy which handles and enforces the pointwise-in-time state and control constraints to rendezvous and dock with a rotating target. In the proposed strategy, constraints on thrust, spacecraft positioning within a docking cone corridor, and collision avoidance are systemically treated and switched to achieve collision-free and soft-docking in a two-phase RVD maneuvering. The constraints on the terminal safe docking corridor are imposed and predicted in the NMPC framework. The collision avoidance constraints are the key enablers for collision-free and soft docking. Specifically, a nonlinear constraint with a switching algorithm for collision avoidance is introduced to guide the chaser spacecraft into the docking port in the final docking phase. The NMPC problem is solved using the nonlinear programming solver IPOPT (Interior Point OPTimizer) in real time.[11] The performance of the developed NMPC controller is analyzed for several test cases using a MATLAB/Simulink-based simulator.[12,13] Finally, the proposed controller is implemented on an air-bearing test bed to demonstrate the real-time capability of the proposed guidance and control method.

MISSION DESIGN AND DOCKING STRATEGY

Mission Design

We consider an autonomous spacecraft RVD mission between a rotating target platform and a chaser spacecraft. Toward the goal of validating the proposed NMPC strategy on a laboratory experimental test bed, we focus on the terminal planar rendezvous and docking. As a result, the Clohessy-Wiltshire-Hill (CWH) dynamics can be shown to reduce to double integrator dynamics for short duration maneuvers in the immediate vicinity of the target spacecraft. Additionally, it is assumed the target platform rotates at a constant angular velocity $\omega_p \geq 0$ around its center of mass.

In previous experimental investigations, experiments for spacecraft RVD to a rotating target have mainly assumed two similarly-sized spacecraft.[2,5] In our study, we consider two dissimilarly sized spacecraft, as illustrated in Figure 1(a), where the chaser spacecraft is smaller than the rotating target platform. In the experimental setup, a test vehicle (solid black lines) is used as the docking port of an emulated larger rotating target platform (dotted black lines), as illustrated by Figure 1(b). The docking port of the (emulated) target platform is assumed to trace out the circumference of a disk with radius r_p.



(a) Rotating Target with Chaser (b) Emulated Target Platform

Figure 1. Rotating Target Platform Setup

As illustrated in Figure 2, the RVD mission is constructed of two distinct phases: a free-flying phase and a final docking approach phase.[3,4] In each phase, the desired position of the chaser is set as a holding position. In the free-flying phase, the holding position is considered to be along the at the entry to and along the centerline of the docking cone corridor. The transition from the free-flying phase to the final docking approach occurs when the chaser is within a certain distance of the holding position. When the second phase, the final docking approach, begins, the holding position is translated along centerline of the docking cone corridor such that the chaser successfully docks with the rotating target platform. For safe docking and precise attitude pointing to the docking port in the final docking phase, the holding position is guided by a nonlinear collision avoidance constraint.

Rendezvous and Docking Strategy

The docking strategy based on nonlinear constraints is introduced for safe docking between the rotating target and chaser spacecraft. The goal of the proposed strategy is to leverage the handling of nonlinear collision avoidance constraints using the NMPC framework. In addition, to overcome computational challenges in solving the optimization problem associated with NMPC in real time, we aim to design a computationally tractable NMPC strategy for real-time implementation.

In the first phase of free flying, the desired position of the chaser is set as a holding position located at the entry to and along the centerline of the docking cone corridor. A threshold value c_{α_1} for the distance error α_1 between the holding position and the chaser's position is set to decide the switching timing from the free-flying phase to the final docking approach. In addition to the distance error from the holding position, we also consider the attitude error α_2 between the docking port and the chaser spacecraft. Since the target platform rotates and its attitude changes continuously the attitude alignment is critical for safe docking and should be handled carefully. This consideration helps the attitude of the chaser match to the docking cone precisely in the final docking approach

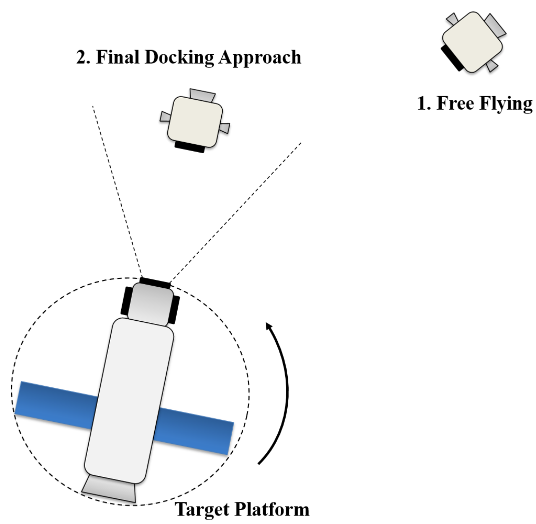


Figure 2. Two-Phase Maneuver for Docking with a Rotating target Platform

phase.

To meet the requirements for the precise attitude alignment and soft docking, two collision avoidance constraints are enforced as shown in Figure 3. One is a fixed keep-out constraint surrounding the rotating target (red circle) to ensure avoidance of collision for the entire maneuvering period. The other collision avoidance constraint (green circle) is dynamically reconfigurable during the chaser's maneuvers, specifically, in the final docking approach. With satisfaction of two conditions, the constraint shrinks to the fixed collision avoidance constraint as its radius decreases. The two conditions are based on the distance error α_1 and the attitude error α_2. If the distance error α_1 between the chaser's position and the holding position is within a certain distance range c_{α_1}, then the attitude error α_2 between the chaser and the rotating docking port is checked. If the attitude error is less than the threshold value c_{α_2}, the dynamic collision avoidance constraint shrinks from the radius of at the previous sampling instant to a new radius by multiplying a constant scale factor γ where $0 < \gamma < 1$. This procedure ensures the precise attitude pointing of the chaser spacecraft and achieves fine control when the chaser gets closer to the docking port of the rotating target platform. Additionally, it plays a role for soft docking as the radius decreases slowly. Hence, we have possible advantage in computation without increasing computational complexity with additional hard constraints for a soft docking condition. To implement the proposed strategy, a simple algorithm for enforcing the dynamically reconfigurable approaching cone constraint and nonlinear collision avoidance constraint is introduced in the next section.

NONLINEAR MODEL PREDICTIVE CONTROLLER DESIGN

In this section, an NMPC controller is designed with the proposed spacecraft RVD strategy. First, a dynamic model is introduced for the chaser spacecraft simulator. Constraints are then described to consider collision-free, soft, and safe docking. The constraint handling algorithm for dynamically reconfigurable constraints is introduced for the phase switching and soft docking.

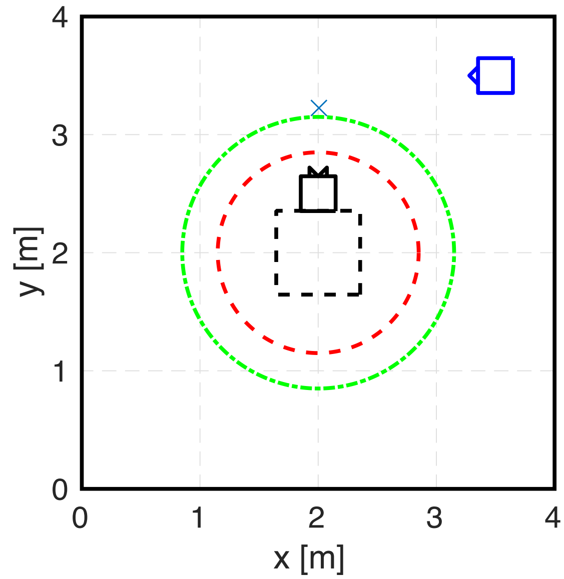


Figure 3. Collision Avoidance Constraints and Holding Position

Model

The CWH equations for spacecraft in circular orbits describes the relative translational motion between the target and chaser spacecraft. The resulting dynamics can be simplified to double integrator dynamics since the chaser spacecraft maneuvers in close proximity of the target spacecraft over a short time span.

The model of the chaser spacecraft is expressed by the following double-integrator type equations of motion for the two translational DoF and one rotational DoF system,

$$\ddot{x} = \frac{F_x}{m}, \quad \ddot{y} = \frac{F_y}{m}, \quad \ddot{\theta} = \frac{\tau}{I_z},$$

where $F_x,\ F_y$ are the control forces in the respective inertial x and y directions, τ is the control torque, m denotes the vehicle mass, and I_z is the vehicle moment of inertia about its vertical axis, respectively.

The state-space representation of the chaser spacecraft model is written as

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

where $\mathbf{x} = [x,\ y,\ \theta, \dot{x},\ \dot{y},\ \dot{\theta}]^T$ denotes the state vector, $\mathbf{u} = [F_x,\ F_y,\ \tau]^T$ denotes the control vector, and A and B are the corresponding state and control matrices, respectively. The discrete-time model is obtained as

$$\mathbf{x}(k+1) = A_d\mathbf{x}(k) + B_d\mathbf{u}(k), \tag{2}$$

where $\mathbf{x}(k) \in \mathbb{R}^6$ and $\mathbf{u}(k) \in \mathbb{R}^3$ are the state and input vectors at the sampling instant $k \in \mathbb{Z}_{0+}$, respectively. The matrices A_d and B_d are the discrete state and control matrices derived from the continuous dynamics in Eq. (1).

Constraints

The following constraints are considered for the NMPC approach to achieve autonomous spacecraft RVD between the chaser spacecraft and the rotating target spacecraft while avoiding collision.

Thruster Constraints We assume that thrusters of the chaser spacecraft simulator generates maximum thrust force u_{max},

$$
\begin{aligned}
|u_1(k)| &\leq u_{max}, \qquad (3)\\
|u_2(k)| &\leq u_{max}. \qquad (4)
\end{aligned}
$$

Approaching Cone Constraints Approaching cone constraints are enforced at the docking port of the target spacecraft to ensure safe docking in the final docking approach phase. In the Free-Flying phase, the entry cone constraints are not enforced but the holding position is placed at the entry of the cone constraints. These constraints are predicted over the horizon of the NMPC framework to achieve good performance in tracking. The approaching cone constraints are linearized by constructing two hyperplanes that define the edges of the entry cone and intersect at the docking point. When these constraints are activated, the chaser FSS is forced to stay within the two hyperplanes until docking is achieved.

$$
\begin{aligned}
-\hat{\mathbf{n}}_{c_1}(k) \cdot \mathbf{p}(k) &\leq -\hat{\mathbf{n}}_{c_1}(k) \cdot \mathbf{p}_{\text{dock}}, \qquad (5)\\
\hat{\mathbf{n}}_{c_2}(k) \cdot \mathbf{p}(k) &\leq \hat{\mathbf{n}}_{c_2}(k) \cdot \mathbf{p}_{\text{dock}}. \qquad (6)
\end{aligned}
$$

Eqs. (5) and (6) enforce hyperplane constraints for the entry cone, where $\hat{\mathbf{n}}_{()}$ defines the normal vector of the hyperplane, $\mathbf{p}_{\text{dock}}$ defines a point on the hyperplane, $\mathbf{p}(k)$ is the position vector at the sampling instant k, i.e., $\mathbf{p}(k) = [x(k),\ y(k)]^T$.

Collision Avoidance Constraints For safe docking, exclusion zone constraints are considered as collision avoidance constraints. A collision avoidance constraint is introduced by defining a keep-out zone in a form of an ellipsoid[14, 15] around the rotating target platform. In this research, circles are used as the keep-out zone constraints to avoid collision. One constraint for collision avoidance is considered as a circle with a constant radius around the rotating target platform. Another collision avoidance constraint is enforced with the holding position to guide the chaser spacecraft into the entry corridor with a dynamic radius.

$$
\begin{aligned}
r_t^2 &\leq (\mathbf{p}(k) - \mathbf{c}_t)^T(\mathbf{p}(k) - \mathbf{c}_t), \qquad (7)\\
r_{\text{hold}}^2(k) &\leq (\mathbf{p}(k) - \mathbf{c}_t)^T(\mathbf{p}(k) - \mathbf{c}_t), \qquad (8)
\end{aligned}
$$

where $\mathbf{c}_t$ is the center position of the rotating target, r_t is the radius of the fixed circle, and r_{hold} is the radius of the dynamic circle. The holding position is located at the aligned line between the docking port and the center of the rotating target platform with a small gap distance from the outer circle constraint in the both phases.

NMPC Design

The NMPC controller is developed to minimize the cost function given by,

$$
\begin{aligned}
J &= (\mathbf{x}(k+N) - \mathbf{x}_t(k+N))^T P(\mathbf{x}(k+N) - \mathbf{x}_t(k+N))\\
&\quad + \sum_{i=0}^{N-1} (\mathbf{x}(k+i) - \mathbf{x}_t(k+i))^T Q(\mathbf{x}(k+i) - \mathbf{x}_t(k+i)) + \mathbf{u}(k+i)^T R\mathbf{u}(k+i), \qquad (9)
\end{aligned}
$$

1. Initialization:
Set $r_{\text{hold}}(k) = r_{\text{hold},0}$ at the sampling instant $k = 0$.
Start *Free-Flying phase*.
2. Measure the distance error α_1 between the chaser's current position $\mathbf{p}(k)$ and $\mathbf{x}_t(k)$.
3.
if $\alpha_1 < c_{\alpha_1}$ **then**
  Measure the attitude error α_2 between the chaser and the docking port.
  if $\alpha_2 < c_{\alpha_2}$ **then**
    if *Free-Flying phase* **then**
      Start *Final Docking Approach phase*.
      $k = k + 1$ and set $r_{\text{hold}}(k) = \gamma r_{\text{hold}}(k-1)$.
      Solve the NMPC problem and go to step 2.
    else
      $k = k + 1$ and set $r_{\text{hold}}(k) = \gamma r_{\text{hold}}(k-1)$.
      Solve the NMPC problem and go to step 2.
    end
  else
    $k = k + 1$ and set $r_{\text{hold}}(k) = r_{\text{hold}}(k-1)$.
    Solve the NMPC problem and go to step 2.
  end
else
  $k = k + 1$ and set $r_{\text{hold}}(k) = r_{\text{hold}}(k-1)$.
  Solve the NMPC problem and go to step 2.
end

Figure 4. An Illustration of the Algorithm to Switch Phases and Enforce a Dynamic Collision Avoidance Constraint

where N is the length of the prediction horizon, and $\mathbf{x}_t$ is the desired position. In the proposed spacecraft RVD strategy, the desired position is the holding position. We assume that the estimation of the desired position is available during maneuvers. The holding position is predictable with the given radius of the dynamic collision avoidance constraint and the constant angular velocity of the rotating target over the prediction horizon. The translational and rotational motions are included in the NMPC formulation. The matrices P, Q, and R in Eq. (9) define the cost function weights on the final state, intermediate states, and control variables, respectively. The matrix P is chosen as the positive-definite solution of the discrete Riccati equation.[16,17] Note that NMPC is able to deal with these constraints directly so that any approximation methods are not required. For each phase of RVD maneuvering, the NMPC problem formulated as follows.

Free-Flying Phase In this phase, Eq. (9) is minimized subject to the equality constraint enforcing the dynamics described by Eq. (2), the control constraints given in Eqs. (3) and (4), and the collision avoidance constraints specified by Eq. (7) with the constant radius r_t and Eq. (8) with the initial radius $r_{\text{hold},0}$. The chaser spacecraft approaches the holding position in this phase without the approaching cone constraints. The phase switching is decided by the proposed algorithm as shown in Figure 4.

Final Docking Approach Phase The outer collision avoidance constraint is dynamically reconfigurable in this phase, *i.e.*, it shrinks continuously if the corresponding conditions in the algorithm

Table 1. Test Case Parameters

Parameter	Value
Initial State	$\mathbf{p}_{c,0} = (3.5, 3.5)$ m
Docking Cone Half-Angle	$\theta_h = 20°$
Docking Cone Corridor Length	$l_c = 0.5$ m
Fixed Keep-Out Zone Radius	$r_t = 0.65$
Initial Radius of Dynamic Keep-Out Zone	$r_{hold,0} = 1.15$ m
Target Angular Velocity, Case A	$0.5°/s$
Target Angular Velocity, Case B	$1.0°/s$
Target Angular Velocity, Case C	$2.0°/s$

are satisfied. Eq. (9) is minimized subject to the equality constraint enforcing the dynamics described by Eq. (2), the control constraints given in Eqs. (3) and (4), the approaching cone constraints predicted over the horizon of the NMPC framework described by Eqs. (5) and (6), and the collision avoidance constraints given in Eq. (7) with the constant radius $r_{\rm t}$ and Eq. (8) with a decreasing radius $r_{\rm hold}(k)$.

SIMULATION RESULTS

The NMPC controller was implemented in simulation, using a MATLAB/Simulink based numerical simulator that represents the experimental test vehicle plant and emulates the on-board sensors and actuators.[12,13] To solve the nonlinear optimization problem associated with NMPC in real time, an efficient solver IPOPT (Interior Point OPTimizer)[11] is utilized

Test Scenarios

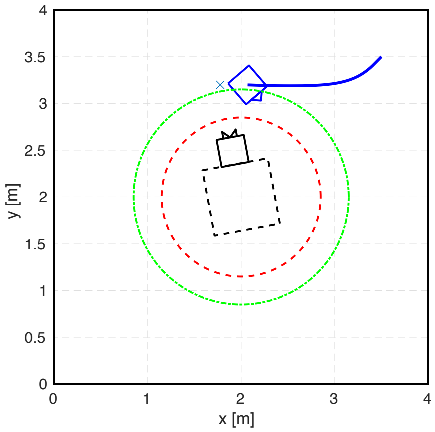
Three different angular rates for the rotating target platform were considered to test the guidance and control algorithm. We assume that the fixed keep-out zone of the rotating target is a circle with the radius of 0.65 m and its center is at $(2,\ 2)$ m. The initial position and attitude of the chaser and the target conditions were the same for all three cases, while the angular velocity of the rotating target changes. The half angle θ_h and length l_c of the entry cone are 20 deg and 0.5 m, respectively. The scale factor γ is chosen to be 0.98 in simulations. Table 1 summarizes the simulation set-up for the cases.

In the simulations and experiments, we have used the same weighting matrices for the NMPC cost functional chosen as,

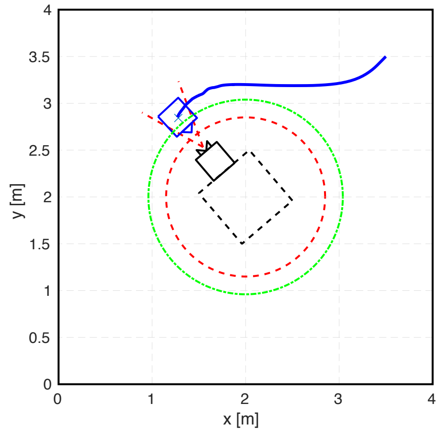
$$Q = \text{diag}(10,\ 10,\ 10,\ 300,\ 300,\ 500),\ \ R = 10\mathbf{I}_3. \tag{10}$$

The prediction horizon is set to $N = 40$, and the sampling period is set to 2 s to consider the controller execution time with the prediction horizon. Thus, the prediction window is 80 s. The maximum thrust force in the x and y directions is selected as $F_{max} = 0.15$ N. As a fuel-related metric, the control effort J_1 is defined using the ℓ_1-norm as,

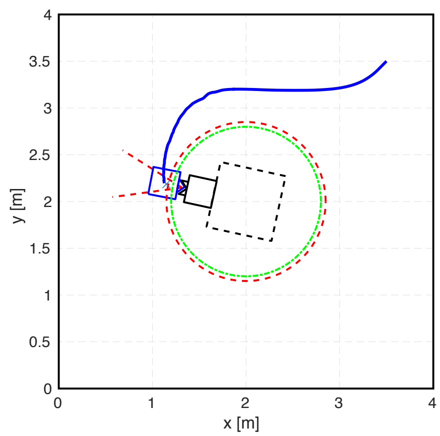
$$J_1 = \int_{t_0}^{t_f} \sum_{i=1}^{8} u_i \, dt, \tag{11}$$



(a) Free-Flying



(b) Final Docking Approaching



(c) Docking

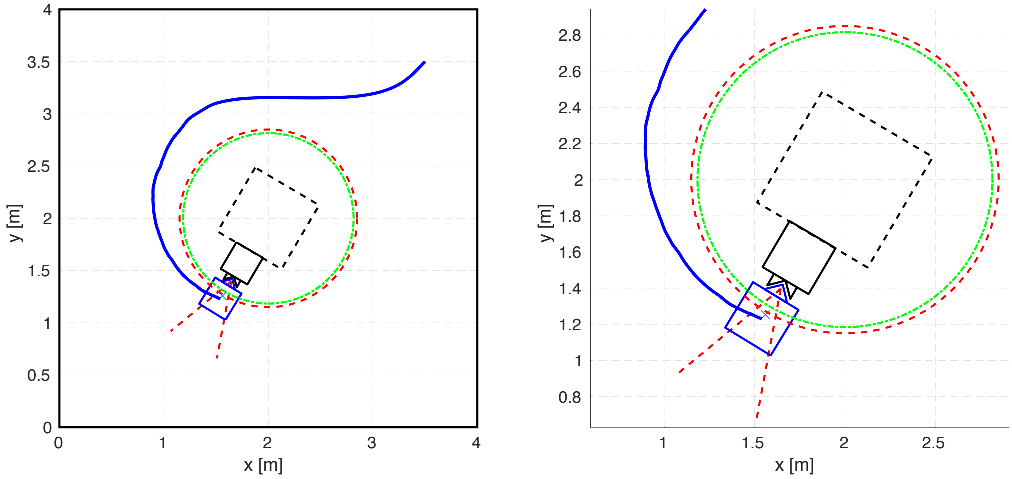
Figure 5. Rendezvous and Docking Maneuver of the Chaser FSS in Case A

Table 2. Simulation Results

Metric	Case A	Case B	Case C
Control effort (Ns)	5.98	7.04	9.40
Time-to-dock (s)	152	155	165
Avg. computation time (s)	0.02	0.02	0.02
Max. computation time (s)	0.08	0.24	0.13

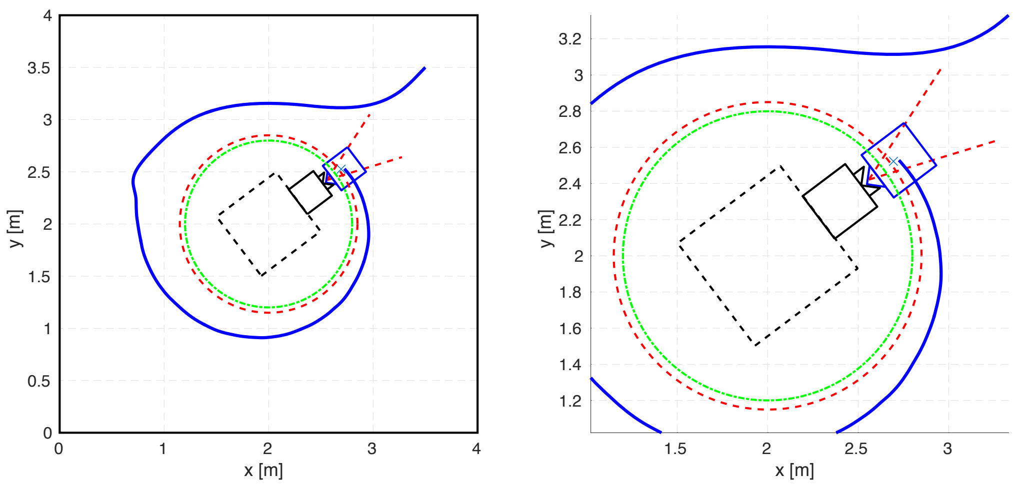
where $u_i \in \{0, 1\}$ is the on-off state of the thruster with a resolution of 0.01 s for each thruster in real implementation, t_0 is the initial time when the FSS guidance is enabled, and t_f is the final time when the docked conditions are met.[18,19]

For three cases, simulation results demonstrate that docking and rendezvous maneuvers are successfully achieved using the NMPC approach with the proposed constraints design and handling



(a) Tarjectory (b) Zoomed-in trajectory

Figure 6. Trajectory of the chaser FSS in Case B



(a) Tarjectory (b) Zoomed-in trajectory

Figure 7. Trajectory of the Chaser FSS in Case C

strategy. Figure 5 shows each phase of the chaser FSS maneuvering in Case A. Figures 6 and 7 illustrate trajectories of Cases B and C, respectively.

Table 2 summarizes the performance results of the test cases in simulations. The dominant factor in the time-to-dock results is the shrinking rate of the radius γ since γ decides the holing position's approaching speed to the docking port. This resulted in the small differences of the docking time among Cases A, B, and C. The average computation time for all cases is 0.02 s and the worst case is 0.24 s in Case B. The simulation were implemented on a computer with Intel(R) CPU @ 2.20 GHz using the MATLAB/Simulink-based simulator. The computation time was measured by CPU time

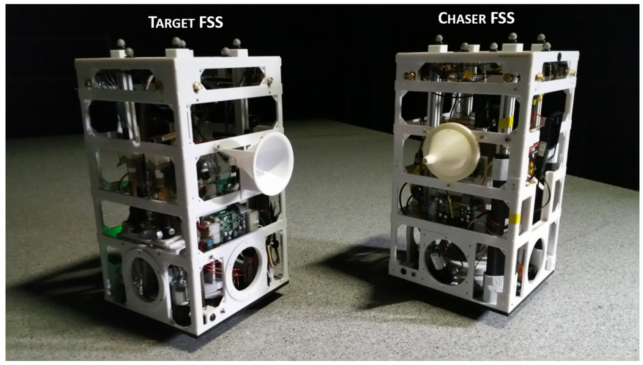


Figure 8. The Chaser and Target FSS Used in the Experimental Testing

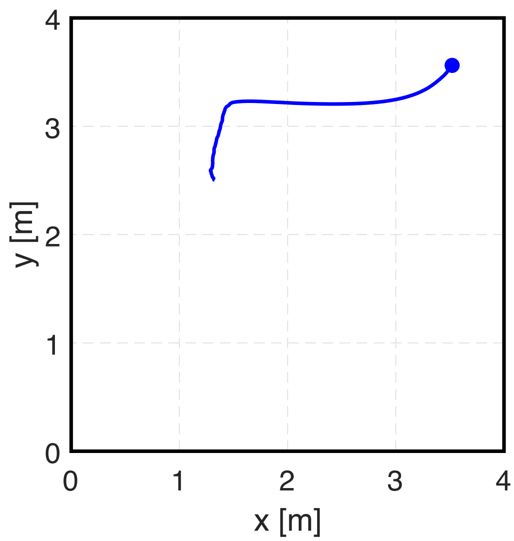


Figure 9. Experimental Trajectory of the Chaser

usage.

EXPERIMENTAL VALIDATION

The experiments were conducted utilizing the NPS-POSEIDYN* test bed[12,13] at the Spacecraft Robotics Laboratory. The experimental setup consisted of two floating spacecraft simulators (FSS), shown in Figure 8, representing the chaser and target spacecraft. Each FSS uses an on-board, PC-104 form-factor computer based on a 32-bit, 1.6 GHz Intel Atom processor with 2 GB of RAM and an 8 GB solid-state drive. A 32-bit `PREEMPT-RT` patched Ubuntu 10.04 operating system provides real-time execution capabilities. The developed NMPC controller is included in the guidance, navigation, and control (GNC) algorithms based on a MATLAB/Simulink environment with the sensor and actuator blocks. The model is then cross-compiled and transferred to the FSS simulator.

The experimental validation was performed utilizing a target platform rotational rate of 0.5 deg/s. The initial attitude of the chaser FSS was selected as 0 deg, and the scale factor for the radius of the

*POSEIDYN stands for Proximity Operation of Spacecraft: Experimental hardware-I-the-loop DYNamic simulator

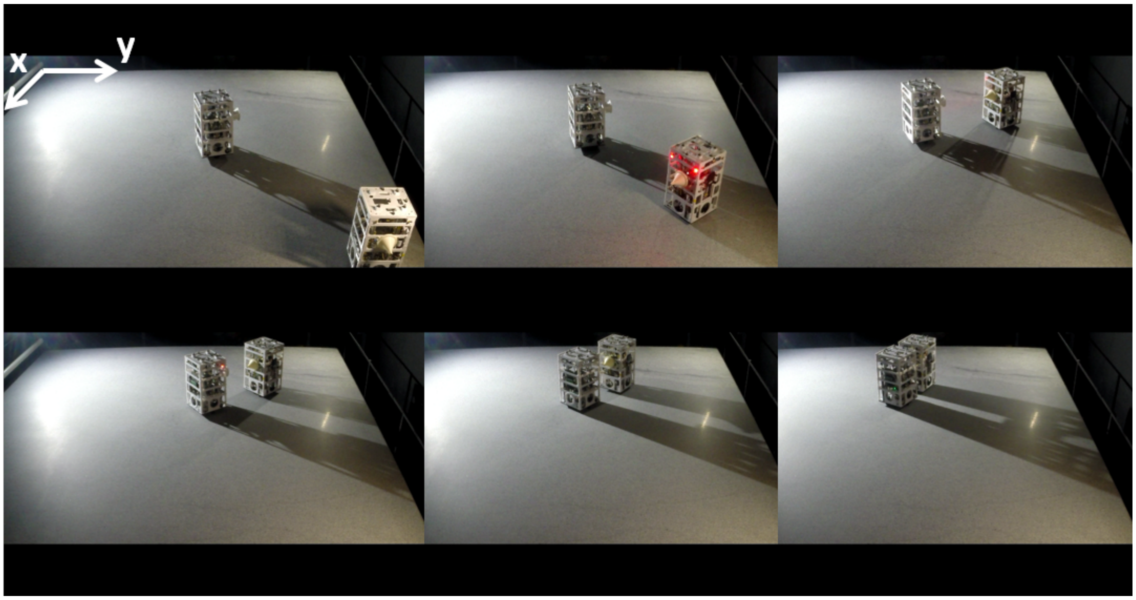


Figure 10. Still Images of the Rendezvous and Docking Maneuver Between the Chaser and Target FSS

Table 3. Experimental Results

Metric	Value
Control effort (N·s)	7.93
Time-to-dock (s)	107
Avg. computation time (s)	0.04
Max. computation time (s)	0.09

keep-out zone constraint is adjusted to $\gamma = 0.9$. Experimental results are presented in Figures 9 and 10. The experimental trajectory shown in Figure 9 is similar to the simulated trajectory in Figure 5 before the chaser FSS began the final docking approach phase. With the reduced shrinking rate γ, the chaser approached the target at a faster rate in the final phase. As shown in Table 3, this resulted in a shorter time-to-dock than the simulation results. This demonstrates the scale factor γ can be used as a tuning parameter for adjusting the soft docking and time-to-dock. Additionally, the control effort was observed to be approximately 28% larger than the control effort in the simulation results for Case A. Furthermore, the computational cost associated with the proposed framework was measured by the elapsed CPU time required to converge to a solution. As tabulated, the proposed framework was found to be capable of generating a solution within 0.1 s. Resultantly, this demonstrates the effectiveness of the proposed guidance framework to generate real-time solutions to the constrained guidance problem.

CONCLUSION

In this paper, a spacecraft rendezvous and docking (RVD) strategy with a rotating target using a nonlinear model predictive control (NMPC) approach has been developed, analyzed, and tested on the MATLAB/Simulink simulator and the air-bearing test bed. Considering the computational complexity involved in solving the nonlinear optimization problem for the NMPC controller, the strategy to effectively enforce and handle constraints has been proposed with dynamically configurable constraints. The constraints on thrust was considered for hardware limitations. The approaching cone constraints on the terminal safe docking corridor have been also imposed and predicted in the NMPC framework. The collision avoidance constraints on chaser spacecraft positioning were designed to achieve collision-free soft docking. A simple algorithm was proposed to switch maneuvering phases of the chaser spacecraft and to enforce a dynamic reconfigurable collision avoidance constraint. Using an efficient solver, IPOPT (Interior Point OPTimizer), the developed NMPC controller with the strategy for handling constraints has been experimentally validated in real time. The experimental results demonstrate the feasibility of using the proposed strategy for spacecraft RVD with a rotating target. They also support further development and implementation of the NMPC approach for spacecraft rendezvous and proximity operations.

REFERENCES

[1] S. Nolet, E. Kong, and D. W. Miller, "Design of an Algorithm for Autonomous Docking with a Freely Tumbling Target," *Defense and Security*, International Society for Optics and Photonics, 2005, pp. 123–134.

[2] S. Nolet, *Development of a Guidance, Navigation and Control Architecture and Validation Process Enabling Autonomous Docking to a Tumbling Satellite*. PhD thesis, Massachusetts Institute of Technology, 2007.

[3] J. Ventura, M. Ciarcià, M. Romano, and U. Walter, "An Inverse Dynamics-Based Trajectory Planner for Autonomous Docking to a Tumbling Target," *AIAA Guidance, Navigation, and Control Conference*, 2016, p. 0876.

[4] J. Ventura, M. Ciarcià, M. Romano, and U. Walter, "Fast and Near-Optimal Guidance for Docking to Uncontrolled Spacecraft," *Journal of Guidance, Control, and Dynamics*, Vol. 38, 2016, pp. 1–17.

[5] M. Wilde, M. Ciarcià, A. Grompone, and M. Romano, "Experimental Characterization of Inverse Dynamics Guidance in Docking with a Rotating Target," *Journal of Guidance, Control, and Dynamics*, 2016, pp. 1–15.

[6] J. Rawlings and D. Mayne, *Model Predictive Control: Theory and Design*. Nob Hill Pub., 2009.

[7] H. Park, S. Di Cairano, and I. Kolmanovsky, "Model Predictive Control for Spacecraft Rendezvous and Docking with a Rotating/Tumbling Platform and for Debris Avoidance," *Proceedings of the 2011 American Control Conference*, 2011, pp. 1922–1927.

[8] 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. 12, 2012, pp. 1398–1427.

[9] R. Findeisen and F. Allgöwer, "Computational Delay in Nonlinear Model Predictive Control," *Proc. Int. Symp. Adv. Contr. of Chem. Proc*, 2004, pp. 427–432.

[10] H. Park, J. Sun, S. Pekarek, P. Stone, D. Opila, R. Meyer, I. Kolmanovsky, and R. DeCarlo, "Real-Time Model Predictive Control for Shipboard Power Management Using the IPA-SQP Approach," *IEEE Transactions on Control Systems Technology*, Vol. 23, No. 6, 2015, pp. 2129–2143.

[11] A. Wächter and L. T. Biegler, "On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming," *Mathematical programming*, Vol. 106, No. 1, 2006, pp. 25–57.

[12] R. Zappulla II, J. Virgili-Llop, H. Park, C. Zagaris, and M. Romano, "Floating Spacecraft Simulator Test Bed for the Experimental Testing of Autonomous Guidance, Navigation, & Control of Spacecraft Proximity Maneuvers and Operations," *AIAA/AAS Astrodynamics Specialist Conference*, 2016, p. 5268, doi:10.2514/6.2016-5268.

[13] R. Zappulla II, J. Virgili-Llop, C. Zagaris, H. Park, and M. Romano, "An Air-Bearing Test Bed for Experimental Evaluation of Auonomous Spacecraft Proximity Manuevers," *AIAA Journal of Spacecraft and Rockets*, doi:10.2514/1.A33769.

[14] C. Petersen, A. Jaunzemis, M. Baldwin, M. Holzinger, and I. Kolmanovsky, "Model Predictive Control and Extended Command Governor for Improving Robustness of Relative Motion Guidance and Control," *Proc. AAS/AIAA Space Flight Mechanics Meeting*, 2014.

[15] C. Jewison, R. S. Erwin, and A. Saenz-Otero, "Model Predictive Control with Ellipsoid Obstacle Constraints for Spacecraft Rendezvous," *IFAC-PapersOnLine*, Vol. 48, No. 9, 2015, pp. 257–262.

[16] J. Maciejowski, *Predictive Control: With Constraints*. Pearson Education, 2002.

[17] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained Model Predictive Control: Stability and Optimality," *Automatica*, Vol. 36, No. 6, 2000, pp. 789–814.

[18] R. Zappulla II, H. Park, J. Virgili-Llop, and M. Romano, "Experiments on Autonomous Spacecraft Rendezvous and Docking Using an Adaptive Artificial Potential Field Approach," *26th AAS/AIAA Space Flight Mechanics Meeting*, Napa, CA, Feb.14–18 2016. `http://hdl.handle.net/10945/50864`.

[19] J. Virgili-Llop, C. Zagaris, H. Park, R. Zappulla II, and M. Romano, "Experimental Evaluation of Model Predictive Control and Inverse Dynamics Control for Spacecraft Proximity and Docking Maneuvers," *6th International Conference on Astrodynamics Tools and Techniques (ICATT)*, Darmstadtium, Germany, 2016. `http://hdl.handle.net/10945/50868`.