

Research Article

Model Predictive Control to Autonomously Approach a Failed Spacecraft

Xun Wang ¹, Zhaokui Wang ², and Yulin Zhang^{1,2}

¹College of Aeronautics and Astronautics, National University of Defense Technology, Changsha 410073, China

²School of Aerospace Engineering, Tsinghua University, Beijing 100084, China

Correspondence should be addressed to Zhaokui Wang; wangzk@tsinghua.edu.cn

Received 1 June 2018; Revised 27 September 2018; Accepted 16 October 2018; Published 6 December 2018

Academic Editor: Seid H. Pourtakdoust

Copyright © 2018 Xun Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this study, a model predictive control (MPC) method is developed for a servicer spacecraft autonomously approaching a tumbling failed spacecraft at an ultraclose range. Flight safety and collision avoidance are basic requirements during the approach. Two types of a failed spacecraft with complex configurations are considered, and a double-ellipsoid composite envelope strategy is designed to model their keep-out zones. Given the keep-out zone of the servicer, two expanded ellipsoids are subsequently introduced to determine the collision and sufficient conditions for collision avoidance are derived by using the form of concave constraint. The tumbling motion of the target is considered, and a CW-based translational dynamics and derived attitude dynamics of the target are formulated to predict the motion of the docking point and keep-out zone. The MPC is formulated to drive the servicer tracking the docking point with collision avoidance and handle constraints including control input saturation and relative velocity bound. Convexification of the collision avoidance constraint and sequential convex programming are adopted for the implementation of MPC. Scenarios on the servicer with different initial positions approaching the target with different angular velocities are simulated, and the simulation results indicate that the proposed MPC method is effective.

1. Introduction

On-orbit services such as the repair of a failed spacecraft, spacecraft refueling, and spacecraft reorbiting are extremely important. A number of missions were conducted, such as the orbital express (OE) demonstration mission [1], the engineering test satellite VII (ETS-VII) project [2], and the spacecraft for the universal modification of orbits (SUMO) project [3]. Recently, on-orbit services for geosynchronous orbit (GEO) targets have attracted significant attention because of their specific importance. The Phoenix Program [4] and the Robotic Servicing of Geosynchronous Satellites Program (RSGS) [5] have thus been proposed in the last decade. An important stage, namely, “approaching targets at an ultraclose range,” is significant for the implementation of on-orbit services. However, it is challenging to ensure an effective and safe approach when the target exhibits a complex configuration and one that especially features uncontrolled tumbling motion.

The study focuses on the control problem for a servicer spacecraft approaching a failed spacecraft with tumbling motion at an ultraclose range. Two types of the on-orbit failed spacecraft with complex configurations are considered. The first is a spacecraft with two large solar panels symmetrically mounted on its body, i.e., most GEO failed spacecraft [6, 7]. The second is a spacecraft with solar panels deployed on only one side where deployment of panels on the other side fails, i.e., SinoSat-II and TV-Sat-I in GEO [8]. The objective of the approach involves tracking a docking point (DP) that is fixed to the body frame of the target and is a few meters from its surface such as the docking mechanism on an apogee motor [5]. The implementation of the approach is conducive to the next step in on-orbit operation, i.e., catching the target with a space manipulator [9]. Flight safety is the major consideration in the approach. The keep-out zone that covers the outer surface of the target is always used for collision avoidance [10, 11]. Most related studies adopted a sphere envelope to model target’s keep-out zone [12, 13]

although this is unsuitable for the approach problem discussed in the study. Consequently, a double-ellipsoid composite envelope strategy is designed to model the keep-out zone of the above two types of the failed spacecraft, and sufficient conditions for the collision are derived.

Numerous guidance and control methods are used for spacecraft approaching at an ultraclose range [14–16]. However, most of the aforementioned methods do not consider collision avoidance with the target. Guidance methods based on the optimal control principle are widely used. Given the complex state or control constraints, it is difficult to solve analytic solutions to optimal control. In reference [17], the Gauss pseudospectral method was employed to solve the optimal rendezvous trajectory with certain constraints. In references [18–20], Pontryagin’s minimum principle was combined with numerical methods to formulate and solve the optimal rendezvous trajectory by considering collision avoidance. Another widely investigated method is the artificial potential function (APF) method. Dong et al. [21] introduced a potential function and two constrained zones to plan a safe rendezvous path. An adaptive control law based on a time-varying sliding manifold was subsequently used to track the desired path. Ge et al. [22] addressed the problem of docking with a tumbling target by using the APF method along with a sliding control. Tumbling motion was classified into three cases, and different safe boundaries were discussed in detail.

Model predictive control is another method that should be considered. Given its working principle, it is extremely suitable for dealing with complex constraints such as multi-variable state constraints and control input constraints. Several studies focused on investigating how to conduct close-range rendezvous and docking using the MPC method [23–25]. In reference [24], strategies for handling collision avoidance constraints, control input constraints, velocity constraints, and line-of-sight constraints were discussed in detail. Several studies also investigated the application of MPC to docking with a tumbling target [13, 26–28]. Under the condition, the docking point and the collision avoidance constraints are time variant. Collision avoidance is always modeled as a nonlinear and nonconvex constraint, and thus, it is difficult to ensure computational convergence and efficiency while solving the underlying optimization problem in MPC. Typically, the linearization of the collision avoidance constraint at the desired docking point is adopted [27], which is conducive to solving the optimization problem. However, the derived new collision avoidance zone is subsequently converted into a half-space, which is overly conservative, and the object in safe zones may be misjudged as corresponding to prohibited zones. Other effective methods for handling the collision avoidance constraints such as sequential convex programming and mixed integer programming have been investigated in spacecraft swarm mission and spacecraft rendezvous [29–31]. Based on the aforementioned studies, an MPC is developed for a servicer approaching a tumbling failed spacecraft. A novel collision discrimination method is proposed with the aim of handling the collision avoidance constraint with the modeled double-ellipsoid envelope of the target. Furthermore, a convexification method for

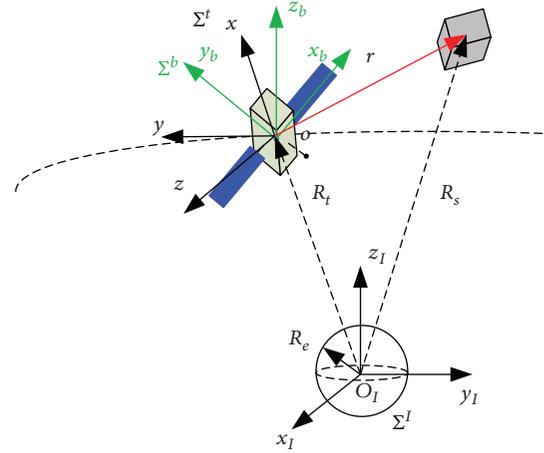


FIGURE 1: Geometrical relationship and coordinate systems.

the constraints and a sequential convex programming are adopted to implement the MPC.

The remainder of the study is organized as follows: Section 2 presents the mathematical formulation of relative translational dynamics and the attitude dynamics of the target. The control object and constraints on modeling for the approach are stated. Section 3 presents the design of the MPC controller and its solution in detail, and Section 4 provides the results of a numerical simulation to verify the performance of the proposed algorithms. The concluding remarks are discussed in Section 5.

2. Mathematical Formulation

In this section, the relative translational dynamics of the servicer and target are set along with the rotational dynamics of the target. Both the aforementioned dynamics form the basis of the state prediction and construction of time-variant constraints used in the design of the MPC.

The relative geometrical relationship and coordinate systems are shown in Figure 1.

As shown in Figure 1, $\mathbf{r} \in \mathbb{R}^3$ denotes the vector directed from the center of mass (c.m.) of the target to the c.m. of the servicer. Additionally, R_t and R_s denote the distances of the target and servicer, respectively, relative to the center of the Earth, and R_e denotes the radius of the Earth.

2.1. Inertial Coordinate System (Σ^I). The origin of the inertial coordinate system $o_I - x_I y_I z_I$ is centered on Earth, the z_I -axis is along the rotational axis, and the x_I -axis points toward the vernal equinox. The y_I -axis completes the right-handed orthogonal.

2.2. Target Local Vertical-Local Horizontal (LVLH) Coordinate System (Σ^t). The origin of the LVLH coordinate system $o - x_t y_t z_t$ lies on the c.m. of the target. The x_t -axis is in the opposite direction to Earth’s center, the y_t -axis is along the direction of flight, and the z_t -axis completes the right-handed orthogonal.

2.3. Target Body-Fixed Coordinate System (Σ^b). The origin of the target body-fixed coordinate system $o - x_b y_b z_b$ is in the c.m. of the target. Without loss of generality, it is assumed that x_b , y_b , and z_b are aligned with its principal axes.

2.3.1. Relative Translational Dynamics. It is assumed that the target is on a circular orbit and the distance between the servicer and target is extremely low. Given a short-period approach, the disturbances caused by solar pressure, atmospheric drag, and the effects of nonspherical gravity perturbation are neglected in both orbital and attitude motions. The Clohessy–Wiltshire (CW) equation is subsequently used to describe the relative translational dynamics [13]:

$$\begin{aligned} \ddot{x} - 2n\dot{y} - 3n^2x &= u_x, \\ \ddot{y} + 2n\dot{x} &= u_y, \\ \ddot{z} + n^2\dot{z} &= u_z, \end{aligned} \quad (1)$$

where $[u_x, u_y, u_z]$ denote the components of the control acceleration and are resolved into Σ^t , $n = \sqrt{\mu/R_t^3}$ denotes the orbital rate of the target with respect to Σ^I , and μ denotes the gravitational constant.

Eq. (1) is rewritten in the form of a state space model as follows:

$$\dot{X} = AX + BU, \quad (2)$$

where $X \in \mathbb{R}^6$ denotes the state vector, $U \in \mathbb{R}^3$ denotes the control vector, and

$$\begin{aligned} X &= \begin{pmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix}, \\ A &= \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3n^2 & 0 & 0 & 0 & 2n & 0 \\ 0 & 0 & 0 & -2n & 0 & 0 \\ 0 & 0 & -n^2 & 0 & 0 & 0 \end{pmatrix}, \\ B &= \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \\ U &= \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix}. \end{aligned} \quad (3)$$

With respect to the implementation of the MPC, the discrete-time model of the relative translational dynamics is derived with a sampling period T_s as follows:

$$X(k+1) = A_d X(k) + B_d U(k), \quad (4)$$

where $X(k)$ and $U(k)$ denote the state and control vectors, respectively, at sampling instant $k \in \mathbb{Z}^+$. The matrices $A_d \in \mathbb{R}^{6 \times 6}$ and $B_d \in \mathbb{R}^{6 \times 3}$ are defined as follows [28]:

$$\begin{aligned} A_d &= e^{AT_s}, \\ B_d &= \left(\int_0^{T_s} e^{A\tau} d\tau \right) B. \end{aligned} \quad (5)$$

2.3.2. Attitude Dynamics of the Target. The attitude dynamics of the target are modeled to describe the relationship between Σ^t and Σ^b . The attitude of the target is parameterized by using the rotation quaternion $\mathbf{q} = [q_0, \underline{q}^T]^T$, where the first component denotes the scalar part and the other is a three-dimensional vector. The rotation matrix associated with \mathbf{q} , denoted by $R_{tb}(\mathbf{q})$, is given as follows [32]:

$$R_{tb}(\mathbf{q}) = \left(q_0^2 - \underline{q}^T \underline{q} \right) \mathbf{I}_3 + 2\underline{q} \underline{q}^T - 2q_0 [\underline{q} \times], \quad (6)$$

where $R_{tb}(\mathbf{q})$ transforms a vector from Σ^t into Σ^b , \mathbf{I}_3 denotes a three-dimensional identity matrix, and $[\ast \times]$ denotes a skew-symmetric matrix that represents the cross product operator.

The angular velocity \mathbf{w} of Σ^b relative to Σ^t is defined as follows:

$$\mathbf{w} = \mathbf{w}_b - \mathbf{w}_t, \quad (7)$$

where \mathbf{w}_b and \mathbf{w}_t denote the angular velocities of Σ^b and Σ^t , respectively, relative to Σ^I .

The first time derivative of Eq. (7) in the Σ^I is given as follows:

$$\left. \frac{d\mathbf{w}}{dt} \right|_I = \left. \frac{d\mathbf{w}_b}{dt} \right|_I - \left. \frac{d\mathbf{w}_t}{dt} \right|_I. \quad (8)$$

Based on Coriolis's theorem, the following expression is obtained:

$$\left. \frac{d\mathbf{w}}{dt} \right|_I = \left. \frac{d\mathbf{w}}{dt} \right|_t + \mathbf{w}_t \times \mathbf{w}. \quad (9)$$

Given that $\left. d\mathbf{w}_b/dt \right|_I = \left. d\mathbf{w}_b/dt \right|_b$ and $\left. d\mathbf{w}_t/dt \right|_I = \left. d\mathbf{w}_t/dt \right|_t$, a combination of Eqs. (8) and (9) yields the following expression:

$$\left. \frac{d\mathbf{w}_b}{dt} \right|_b = \left. \frac{d\mathbf{w}}{dt} \right|_t + \left. \frac{d\mathbf{w}_t}{dt} \right|_t + \mathbf{w}_t \times \mathbf{w}. \quad (10)$$

The attitude dynamics of the target are given by the following expression:

$$I_t \frac{d\mathbf{w}_b}{dt} \Big|_b + \mathbf{w}_b \times (I_t \mathbf{w}_b) = \mathbf{N}_t, \quad (11)$$

where I_t and \mathbf{N}_t denote the inertial tensors of the target and external torques, respectively. It is assumed that the target is on a circular orbit and environmental perturbations are neglected, and $d\mathbf{w}_t/dt|_t = 0$ and $\mathbf{N}_t = 0$ are subsequently obtained, respectively.

Eq. (10) is substituted into Eq. (11) to yield the following expression:

$$I_t (\dot{\mathbf{w}} + \mathbf{w}_t \times \mathbf{w}) + (\mathbf{w}_t + \mathbf{w}) \times (I_t \cdot (\mathbf{w}_t + \mathbf{w})) = 0, \quad (12)$$

where \mathbf{w}_t , resolved into Σ^t , is given as $\mathbf{w}_t = [0, 0, n]^T$.

Subsequently, the change in target's attitude is given as follows [32]:

$$\dot{\mathbf{q}} = \frac{1}{2} \begin{bmatrix} q_0 \mathbf{I}_3 + [\underline{q} \times] \\ -[\underline{q} \times] \end{bmatrix} \mathbf{w}. \quad (13)$$

According to Eq. (6), the rotation matrix $R_{tb}(k+1)$ at the sampling instant $k+1$ is a function of rotation quaternion $\mathbf{q}(k+1)$. According to Eqs. (12) and (13), $\mathbf{q}(k+1)$ can be obtained through solving the numerical integration with the given $\mathbf{w}(k)$ and $\mathbf{q}(k)$. Thereby, $R_{tb}(k+1)$ can be represented as a nonlinear function expression:

$$R_{tb}(k+1) = f(\mathbf{w}(k), \mathbf{q}(k)). \quad (14)$$

The fourth-order Runge-Kutta algorithm is utilized to solve the numerical integration in Eqs. (12) and (13). It is assumed that there is no attitude maneuvering of the failed target and that there is no external torque applied to the target from the servicer. The offline calculation can be adopted to predict target's attitude, which will be used in MPC to predict the relative translational states and servicer's control input.

2.3.3. Control Objective. The study focuses on two types of the on-orbit failed spacecraft. The first includes two large solar panels that are symmetrically mounted on the body as shown in Figure 2. The second includes solar panels on only one side where deployment fails on the other side as shown in Figure 3.

The x_b -axis is along the center line of the solar panels, the y_b -axis is along the symmetric axis of target's body and is orthogonal to the x_b -axis, and the z_b -axis completes the dextral triad. In the study, the target is assumed as tumbling freely. It is also assumed that target's DP is located along the y_b -axis and the position vector of the DP is $\mathbf{r}_{DP}^b = [0, -l, 0]^T$ resolved in Σ^b . The tracking error is defined as $\delta \mathbf{r}(t) = R_{tb}(t) \mathbf{r}(t) - \mathbf{r}_{DP}^b$, where $\mathbf{r}(t)$ denotes servicer's trajectories resolved in Σ^t . Therefore, the objective of the approach

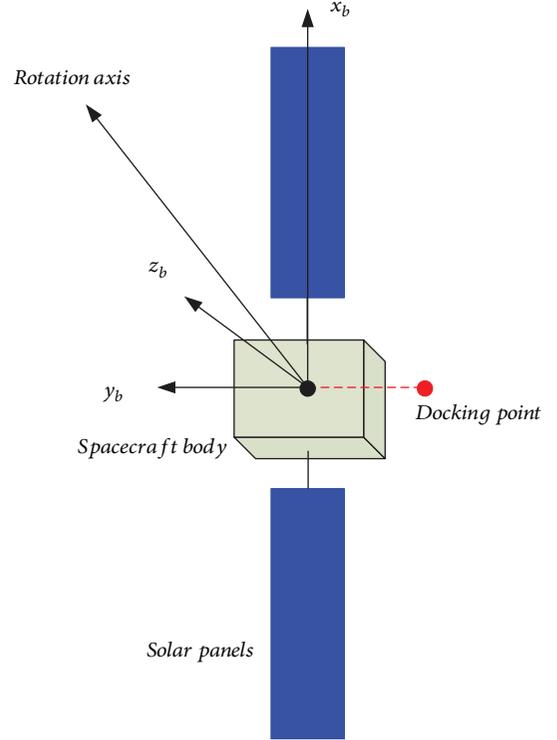


FIGURE 2: Failed spacecraft with two mounted solar panels.

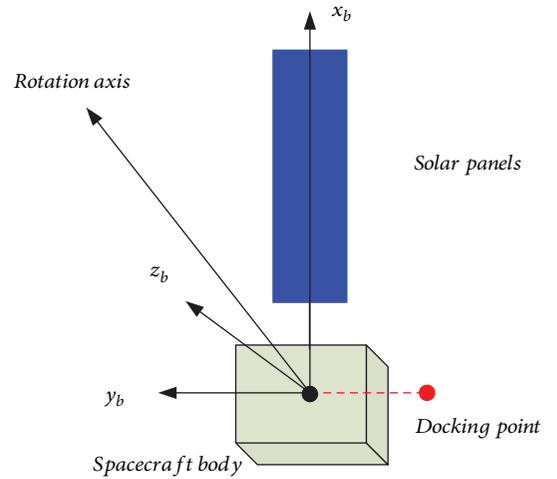


FIGURE 3: Failed spacecraft with solar panels deployed on only one side.

procedure involves driving the servicer while tracking with the target DP, and this is expressed as follows:

$$\lim_{t \rightarrow \infty} \|\delta \mathbf{r}(t)\| \leq \varepsilon, \quad (15)$$

where ε is a constant denoting the upper bound of tracking error.

2.3.4. Constraint Modeling. With respect to the problem of the ultraclose approach, collision avoidance is the basic requirement. Given the aforementioned two types of the

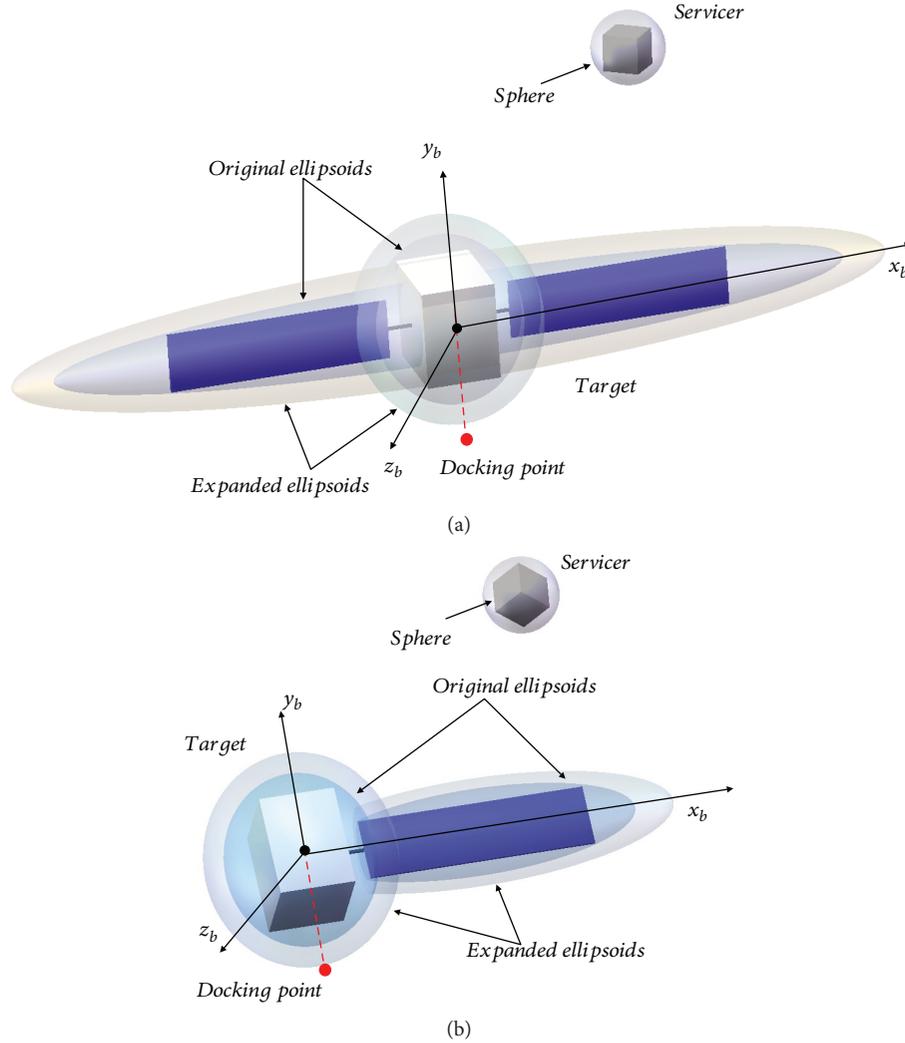


FIGURE 4: The double-ellipsoid composite envelope strategy for (a) the first type of a failed spacecraft and (b) the second type of a failed spacecraft.

failed spacecraft, a double-ellipsoid composite envelope strategy is introduced to model target's keep-out zone in which the spacecraft body and the solar panels are modeled as two different ellipsoids based on their sizes. Furthermore, a sphere envelope with radius r_e is simplified as servicer's keep-out zone.

Intuitively, collision avoidance is attained when the envelopes of the target and the servicer do not intersect. In order to determine collision avoidance, expanded ellipsoids considering the sizes of envelopes of the target and the servicer are introduced, as shown in Figure 4. The original ellipsoids denote target's double-ellipsoid envelope. It is assumed that centers of the original ellipsoids for the first type of the failed spacecraft are located at its mass center. With respect to the original ellipsoid enveloping solar panels of the second type of the failed spacecraft, its center is located at the center of the solar panels.

Furthermore, the scheme for discriminating collision avoidance between the original ellipsoid and the sphere is shown in Figure 5.

Without loss of generality, the original ellipsoid is as follows:

$$\frac{(x-d)^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1, \quad (16)$$

where a , b , and c are constants designed to envelop the target body or the solar panels based on their sizes and $a > b > c$ is satisfied. If the center of ellipsoid is located at target's mass center, $d=0$ is satisfied. The parametric equation of the original ellipsoid is given as follows [17]:

$$\begin{aligned} x &= d + a \sin \theta \cos \varphi, \\ y &= b \sin \theta \sin \varphi, \\ z &= c \cos \theta, \end{aligned} \quad (17)$$

where $\theta \in [0, \pi]$ and $\varphi \in [0, 2\pi)$.

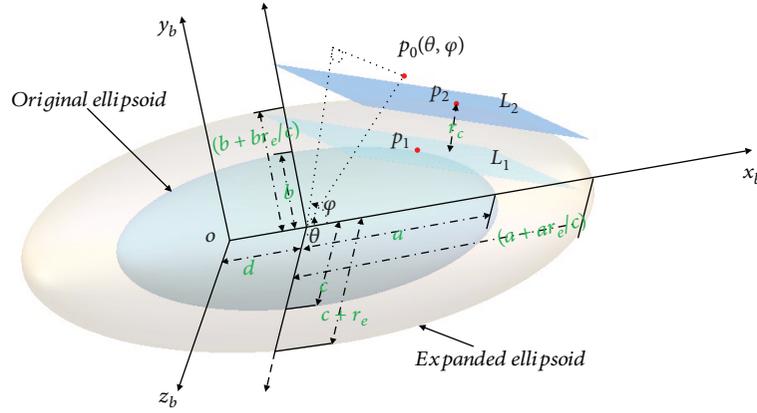


FIGURE 5: Scheme for discriminating collision avoidance between the original ellipsoid and the sphere.

The expanded ellipsoid envelope is as follows:

$$\frac{(x-d)^2}{(a+ar_e/c)^2} + \frac{y^2}{(b+br_e/c)^2} + \frac{z^2}{(c+r_e)^2} = 1. \quad (18)$$

It should be noted that the value of the minimum half-axis of the expanded ellipsoid is the summation of the radius of the sphere and the value of the minimum half-axis of the original ellipsoid. The other two half-axes of expanded ellipsoid are augmented based on the product of the radius and the ratio between the corresponding half-axis and minimum half-axis.

The parametric equation of the expanded ellipsoid is given as follows:

$$\begin{aligned} x &= d + \left(a + \frac{ar_e}{c}\right) \sin \theta \cos \varphi, \\ y &= \left(b + \frac{br_e}{c}\right) \sin \theta \sin \varphi, \\ z &= (c + r_e) \cos \theta. \end{aligned} \quad (19)$$

Proposition 1. Given the original ellipsoid in Eq. (16) and the sphere with radius r_e , the sufficient condition for collision avoidance between the original ellipsoid and the sphere is defined as follows:

$$\frac{(x-d)^2}{(a+ar_e/c)^2} + \frac{y^2}{(b+br_e/c)^2} + \frac{z^2}{(c+r_e)^2} - 1 > 0, \quad (20)$$

where $[x, y, z]$ denote the coordinates of the sphere center. Evidently, Eq. (20) indicates that the center of the sphere is outside the expanded ellipsoid shown in Eq. (18).

Proof. Given a point p_0 using spherical coordinates (θ, φ) describing its direction, there are two other corresponding points located at the original ellipsoidal surface and the

expanded ellipsoidal surface (points p_1 and p_2 , as shown in Figure 5), and their coordinates are given as follows:

$$\begin{aligned} p_1 : x_1 &= d + a \sin \theta \cos \varphi, \\ y_1 &= b \sin \theta \sin \varphi, \\ z_1 &= c \cos \theta, \\ p_2 : x_2 &= d + \left(a + \frac{ar_e}{c}\right) \sin \theta \cos \varphi, \\ y_2 &= \left(b + \frac{br_e}{c}\right) \sin \theta \sin \varphi, \\ z_2 &= (c + r_e) \cos \theta. \end{aligned} \quad (21)$$

Subsequently, the tangent planes of the two points (planes L_1 and L_2 , as shown in Figure 5) are given as follows:

$$\begin{aligned} L_1 : \frac{\sin \theta \cos \varphi}{a} x + \frac{\sin \theta \sin \varphi}{b} y + \frac{\cos \theta}{c} z - \frac{\sin \theta \cos \varphi}{a} d - 1 &= 0, \\ L_2 : \frac{\sin \theta \cos \varphi}{a} x + \frac{\sin \theta \sin \varphi}{b} y + \frac{\cos \theta}{c} z - \frac{\sin \theta \cos \varphi}{a} d - 1 - \frac{r_e}{c} &= 0. \end{aligned} \quad (22)$$

According to Eq. (22), it is noted that the two tangent planes are parallel. Subsequently, the minimum distance from the point p_2 to the tangent plane L_1 is equal to the distance between tangent planes L_2 and L_1 , and this is given as follows:

$$r_c = \frac{r_e}{\sqrt{c^2/a^2 \sin^2 \theta \cos^2 \varphi + c^2/b^2 \sin^2 \theta \sin^2 \varphi + \cos^2 \theta}}. \quad (23)$$

Given that $a > b > c$ is defined, $r_c \geq r_e$ is subsequently obtained and it denotes that the minimum distance between the original ellipsoid and the expanded ellipsoid exceeds or is equal to the radius of the sphere. Thus, when the center of the sphere lies outside the expanded ellipsoid,

the collision avoidance of the sphere and the original ellipsoid is satisfied. The proof is completed.

Remark 1. Given the double-ellipsoid envelope modeling target's keep-out zone, the collision avoidance condition for the target and servicer is the collision avoidance between the sphere and the two original ellipsoids. The modeled ellipsoid envelope is fixed in target's body frame. Therefore, the mathematical description of collision avoidance between the servicer and the target can be conducted in target's body frame. Although this is a sufficient but not a necessary condition to determine collision avoidance, it is derived by using the form of concave constraint, which is conducive to the implementation of MPC as subsequently discussed.

In practice, all thrusters are designed with limited control capability. The infinite norm of vector is applied to describe the thrust saturation constraint, and this is given as follows [24]:

$$\|U\|_{\infty} \leq u_{\max}, \quad (24)$$

where u_{\max} denotes the maximum thrust output.

The relative velocity constraint is also considered to ensure that the servicer exhibits the capability to adjust the thrust output and avoid collision in case of emergency, and this is extremely important in actual on-orbit approaches. The velocity constraint is expressed as follows [27]:

$$\begin{cases} |\dot{x}| \leq \dot{x}_{\max}, \\ |\dot{y}| \leq \dot{y}_{\max}, \\ |\dot{z}| \leq \dot{z}_{\max}, \end{cases} \quad (25)$$

where \dot{x}_{\max} , \dot{y}_{\max} , and \dot{z}_{\max} denote the components of the maximum approach velocity in each direction.

3. Model Predictive Control Formulation

By using the dynamics and constraints discussed in Section 2, the model predictive controller is designed to approach the tumbling failed spacecraft based on the basic principle of finite predictive control. Given the non-convex constraint of the collision avoidance condition, the convexification method is discussed and the sequential convex programming is subsequently used to implement the MPC.

3.1. Prediction of the State Variables. The predicted relative translational state sequence generated by Eq. (4) with state $X(k)$ and control input $U(k)$ is expressed as follows:

$$X_s(k) = A_s X(k) + B_s U_s(k). \quad (26)$$

Subsequently,

$$\begin{aligned} X_s(k) &= \begin{pmatrix} X(k+1|k) \\ X(k+2|k) \\ X(k+3|k) \\ \vdots \\ X(k+N_p|k) \end{pmatrix}, \\ U_s(k) &= \begin{pmatrix} U(k|k) \\ U(k+1|k) \\ U(k+2|k) \\ \vdots \\ U(k+N_c-1|k) \end{pmatrix}, \\ A_s &= \begin{pmatrix} A_d \\ A_d^2 \\ A_d^3 \\ \vdots \\ A_d^{N_p} \end{pmatrix}, \\ B_s &= \begin{pmatrix} B_d & 0 & 0 & \cdots & 0 \\ A_d B_d & B_d & 0 & \cdots & 0 \\ A_d^2 B_d & A_d B_d & B_d & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ A_d^{N_p-1} B_d & A_d^{N_p-2} B_d & A_d^{N_p-3} B_d & \cdots & A_d^{N_p-N_c} B_d \end{pmatrix}. \end{aligned} \quad (27)$$

where $X(k+i|k)$ denotes the predicted state variable at $k+i$ with the given information $X(k)$, $U(k+i|k)$ denotes the predicted control input at $k+i$, $U(k|k) = U(k)$, N_p denotes the prediction horizon, and N_c denotes the control horizon [12].

As described in Section 2.2, the predicated rotation matrix $R_{tb}(k+i)$ at $k+i$ can be solved offline with the given initial $\mathbf{w}(0)$ and $\mathbf{q}(0)$. Thus, while implementing the MPC, the rotation matrix $R_{tb}(k+i)$ is considered as known.

3.2. Optimization Index. The objective of the MPC involves minimizing the tracking error between the predicted states and the desired trajectory. In order to optimize the thruster fuel, the control effort is included in the objective function. Thus, based on Eq. (15), the objective function is defined as follows:

$$\begin{aligned} \Gamma(k) &= \sum_{i=1}^{N_p} \left(R_{tb}(k+i) \mathbf{r}(k+i|k) - \mathbf{r}_{\text{DP}}^b \right)^T Q_i \left(R_{tb}(k+i) \mathbf{r}(k+i|k) - \mathbf{r}_{\text{DP}}^b \right) \\ &\quad + \sum_{i=0}^{N_c-1} U^T(k+i|k) R_i U(k+i|k), \end{aligned} \quad (28)$$

where $Q_i \in \mathbb{R}^{3 \times 3}$ denotes a positive-definite state-weighting matrix, $R_i \in \mathbb{R}^{3 \times 3}$ denotes a positive-definite control-weighting matrix, and $\mathbf{r}(k+i|k)$ is available from the predicted state sequence $X(k+i|k)$.

$$\mathbf{r}(k+i|k) = C_d X(k+i|k),$$

$$C_d = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}. \quad (29)$$

In order to simplify the expression in Eq. (28), matrix $C_s = \mathbf{I}_{N_p} \otimes C_d$ is introduced, where \otimes denotes the Kronecker product of two matrices [27], and \mathbf{I}_{N_p} denotes a N_p -dimensional identity matrix. If we define $\mathbf{r}_d = (\mathbf{r}_{DP}^b, \mathbf{r}_{DP}^b, \dots, \mathbf{r}_{DP}^b)^T$, $R_{tb}(k) = \text{diag}(R_{tb}(k+1), R_{tb}(k+2), \dots, R_{tb}(k+N_p))$, $Q = \text{diag}(Q_1, Q_2, \dots, Q_{N_p})$, and $R = \text{diag}(R_1, R_2, \dots, R_{N_c})$, the objective function in Eq. (28) is expressed as follows:

$$\Gamma(k) = U_s(k)^T W(k) U_s(k) + P(k) U_s(k) + M(k)^T Q M(k), \quad (30)$$

where $W(k) = (\tilde{R}_{tb}(k) C_s B_s)^T Q (\tilde{R}_{tb}(k) C_s B_s) + R$, $P(k) = 2M(k)^T Q \tilde{R}_{tb}(k) C_s B_s$, and $M(k) = R_{tb}(k) C_s A_s X_s(k) - \mathbf{r}_d$ are satisfied.

3.3. Inequality Constraints. In this section, the constraints discussed in Section 2 are reformulated for the implementation of the MPC. We reconsider the collision avoidance constraint shown in Eq. (20), and the collision avoidance between the servicer and the target is given as follows:

$$g_m(x_b, y_b, z_b) = \frac{(x_b - d_m)^2}{(a_m + a_m r_e / c_m)^2} + \frac{y_b^2}{(b_m + b_m r_e / c_m)^2} + \frac{z_b^2}{(c_m + r_e)^2} - 1 > 0, \quad m = 1, 2, \quad (31)$$

where $[x_b, y_b, z_b]$ denote the components of the vector \mathbf{r} resolved in \sum^b and subscript m denotes the two different ellipsoids that model the keep-out zone of the target. Function g_m is defined as the collision threshold, and the collision avoidance is attained when $g_m > 0$. Based on Eq. (31), the collision avoidance constraint for the MPC is expressed as follows:

$$g_m^i(U_s(k)) = \left\{ R_{tb}(k+i) H(i) C_s [A_s X(k) + B_s U_s(k)] - [d_m, 0, 0]^T \right\}^T \times L_m \times \left\{ R_{tb}(k+i) H(i) C_s [A_s X(k) + B_s U_s(k)] - [d_m, 0, 0]^T \right\} - 1 > 0, \quad i = 1, 2, \dots, N_p, \quad m = 1, 2, \quad (32)$$

where $L_m = \text{diag}((a_m + a_m r_e / c_m)^2, (b_m + b_m r_e / c_m)^2, (c_m + r_e)^2)$ and $H(i)$ is expressed as follows:

$$H(i) = [T_1, T_2, \dots, T_j, \dots, T_{N_p}] \begin{cases} T_j = \mathbf{I}_3 & \text{if } j = i, \\ T_j = \mathbf{E}_3 & \text{if } j \neq i, \end{cases} \quad (33)$$

where \mathbf{I}_3 denotes a three-dimensional identity matrix and \mathbf{E}_3 denotes a three-dimensional zero matrix.

After a few manipulations (see Appendix A), Eq. (32) is rewritten as follows:

$$g_m^i(U_s(k)) = U_s(k)^T \tilde{F}_m(i) U_s(k) + \tilde{G}_m(i) U_s(k) + G_m(i)^T L_m G_m(i) - 1 > 0, \quad i = 1, 2, \dots, N_p, \quad m = 1, 2, \quad (34)$$

where $\tilde{F}_m(i) = F(i)^T L_m F(i)$, $\tilde{G}_m(i) = 2G_m(i)^T L_m F(i)$, $F(i) = R_{tb}(k+i) H(i) C_s B_s$, and $G_m(i) = R_{tb}(k+i) H(i) C_s A_s X(k) - [d_m, 0, 0]^T$ are satisfied.

Based on Eq. (24), the control input constraint is reexpressed as follows:

$$(\mathbf{I}_{3N_c}; -\mathbf{I}_{3N_c}) U_s \leq U_{\max}, \quad (35)$$

where \mathbf{I}_{3N_c} denotes a $3N_c$ -dimensional identity matrix and $U_{\max} \in \mathbb{R}^{6N_c}$ is defined as $U_{\max} = (u_{\max}, \dots, u_{\max})^T$.

The velocity constraint expressed in Eq. (25) is reexpressed as follows (see Appendix B):

$$(\mathbf{I}_{3N_p}; -\mathbf{I}_{3N_p}) D_s (A_s X(k) + B_s U_s) \leq V_{\max}, \quad (36)$$

where the matrices D_s and $V_{\max} \in \mathbb{R}^{6N_p}$ are defined as follows:

$$D_d = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

$$D_s = \mathbf{I}_{N_p} \otimes D_d, \quad (37)$$

$$v_{\max} = (\dot{x}_{\max}, \dot{y}_{\max}, \dot{z}_{\max}, \dot{x}_{\max}, \dot{y}_{\max}, \dot{z}_{\max})^T,$$

$$V_{\max} = (v_{\max}^T, \dots, v_{\max}^T)^T.$$

3.4. MPC by Using Sequential Convex Programming. The design of the MPC for the approach problem is summarized as follows:

Problem 1.

$$\begin{aligned}
\min_{U_s(k)} \quad & \Gamma(k) = U_s(k)^T W(k) U_s(k) + P(k) U_s(k) \\
& + M(k)^T Q M(k) \\
\text{subject to} \quad & (\mathbf{I}_{3N_c}; -\mathbf{I}_{3N_c}) U_s(k) \leq U_{\max} \\
& (\mathbf{I}_{3N_p}; -\mathbf{I}_{3N_p}) D_s (A_s X(k) + B_s U_s(k)) \leq V_{\max} \\
& g_m^i(U_s(k)) = U_s(k)^T \tilde{F}_m(i) U_s(k) + \tilde{G}_m(i) U_s(k) \\
& + G_m(i)^T L_m G_m(i) - 1 > 0, \\
& i = 1, 2, \dots, N_p, \quad m = 1, 2 > 0.
\end{aligned} \tag{38}$$

Given that the collision avoidance constraint is a nonconvex constraint, it is difficult to satisfy the convergence and the optimality of the solution to Problem 1. In order to solve Problem 1, it is subsequently converted into a convex optimization problem and sequential convex programming is utilized to implement MPC. Given the collision avoidance constraint in Eq. (38), the linearization technique is adopted to convert it into linear terms as follows:

$$\begin{aligned}
g_m^i(U_s(k)) \approx g_m^i(U_s^{[n]}(k)) + \nabla g_m^i(U_s^{[n]}(k))^T \\
\cdot (U_s(k) - U_s^{[n]}(k)) > 0, \quad i = 1, 2, \dots, N_p, \quad m = 1, 2,
\end{aligned} \tag{39}$$

where $U_s^{[n]}(k)$ denotes the n -iterated solution of $U_s(k)$, and $\nabla g_m^i(U_s^{[n]}(k)) = 2\tilde{F}_m(i) U_s^{[n]}(k) + \tilde{G}_m(i)^T$ denotes the gradient of g_m^i at $U_s^{[n]}(k)$.

Proposition 2. *Eq. (39) is utilized as the collision avoidance constraint as opposed to Eq. (34), and Problem 1 is subsequently converted to Problem 2 as follows:*

Problem 2.

$$\begin{aligned}
\min_{U_s(k)} \quad & \Gamma(k) = U_s(k)^T W(k) U_s(k) + P(k) U_s(k) + M(k)^T Q M(k) \\
\text{subject to} \quad & (\mathbf{I}_{3N_c}; -\mathbf{I}_{3N_c}) U_s(k) \leq U_{\max} \\
& (\mathbf{I}_{3N_p}; -\mathbf{I}_{3N_p}) G_s (A_s X(k) + B_s U_s(k)) \leq V_{\max} \\
& g_m^i(U_s^{[n]}(k)) + \nabla g_m^i(U_s^{[n]}(k))^T (U_s(k) - U_s^{[n]}(k)) > 0, \\
& i = 1, 2, \dots, N_p, \quad m = 1, 2.
\end{aligned} \tag{40}$$

Subsequently, any $U_s(k)$ that is feasible for Problem 2 is also feasible for Problem 1.

Proof. Let $U_s(k)$ denote an arbitrary feasible point for Problem 2, i.e.,

$$g_m^i(U_s^{[n]}(k)) + \nabla g_m^i(U_s^{[n]}(k))^T (U_s(k) - U_s^{[n]}(k)) > 0. \tag{41}$$

Represent $g_m^i(U_s(k))$ by the second-order Taylor series expansion at $U_s^{[n]}(k)$:

$$\begin{aligned}
g_m^i(U_s(k)) = g_m^i(U_s^{[n]}(k)) + \nabla g_m^i(U_s^{[n]}(k))^T (U_s(k) - U_s^{[n]}(k)) \\
+ \frac{1}{2} (U_s(k) - U_s^{[n]}(k))^T \nabla^2 g_m^i(U_s^{[n]}(k)) \\
\cdot (U_s(k) - U_s^{[n]}(k)),
\end{aligned} \tag{42}$$

where $\nabla^2 g_m^i$ denotes the Hessian matrix of g_m^i . It should be noted that $\nabla^2 g_m^i = 2\tilde{F}_m(i)$ denotes a constant positive-definite matrix where $\tilde{F}_m(i) = F(i)^T L_m F(i)$ is satisfied. We substitute Eq. (41) into Eq. (42), and this results in $g_m^i(U_s(k)) > 0$, and thus, any $U_s(k)$ that satisfies the condition in Eq. (39) also satisfies the condition in Eq. (34). The proof is completed.

It should be noted that Problem 2 is a quadratic programming problem [24]. In order to optimize the control input $U_s(k)$, sequential quadratic programming is utilized and a trust region between $U_s^{[n]}(k)$ and $U_s(k)$ is introduced to ensure the convergence.

$$\|U_s(k) - U_s^{[n]}(k)\|_{\infty} \leq \rho^n D_0, \tag{43}$$

where $D_0 \in \mathbb{R}^{3N_c}$ denotes the radius of the trust region and $\rho \in (0, 1)$ denotes a parameter that determines the rate of convergence. After a few manipulations (see Appendix C), Problem 2 that considers the trust region is subsequently converted into Problem 3.

Problem 3.

$$\begin{aligned}
\min_{U_s(k)} \quad & \Gamma(k) = U_s(k)^T W(k) U_s(k) + P(k) U_s(k) + M(k)^T Q M(k) \\
\text{subject to} \quad & (\mathbf{I}_{3N_c}; -\mathbf{I}_{3N_c}) U_s(k) \leq U_{\max} \\
& (\mathbf{I}_{3N_p}; -\mathbf{I}_{3N_p}) G_s B_s U_s(k) \\
& \leq V_{\max} - (\mathbf{I}_{3N_p}; -\mathbf{I}_{3N_p}) G_s A_s X(k) - \nabla g_m^i(U_s^{[n]}(k))^T U_s(k) \\
& < -\nabla g_m^i(U_s^{[n]}(k))^T U_s^{[n]}(k) + g_m^i(U_s^{[n]}(k)), \\
& i = 1, 2, \dots, N_p, \quad m = 1, 2 \\
& (\mathbf{I}_{3N_c}; -\mathbf{I}_{3N_c}) U_s(k) \leq (\rho^n D_0 + U_s^{[n]}(k); \rho^n D_0 - U_s^{[n]}(k)).
\end{aligned} \tag{44}$$

Remark 2. Based on Proposition 2, if there exists a solution feasible for Problem 3, the solution is also feasible for Problem 2. Furthermore, an optimized solution is attained based on sequential quadratic programming for Problem 3. The successive solution process based on Problem 3 requires a first solution $U_s^{[1]}(k)$ to commence. In the study, $U_s^{[1]}(k)$ is set to a zero vector and a maximum iteration number n is designed to end the sequential quadratic programming.

Thus, the constrained optimization problem in the context of the MPC is converted into a sequential quadratic programming problem. The standard quadratic programming algorithm can be employed to solve this problem easily. With the given state vector $X(k)$, rotation quaternion $\mathbf{q}(k)$, and relative angular velocity $\mathbf{w}(k)$ at k , the MPC should be solved at each sampling instant k by using sequential quadratic programming to obtain a sequence of the control input where only the first control input is applied.

4. Simulation Results and Discussion

In order to test the performance of the proposed MPC method, numerical simulations are described in this section. It is assumed that the target is on GEO, and the simulation parameters relevant to the target are listed in Table 1.

Considering the orbit parameters of the target described in Table 1, h is the orbit height, e is the eccentricity, Ω is the right ascension of ascending node, w is the argument of perigee, and f is the true anomaly. Given the two failed spacecraft, both their body's keep-out zones are modeled as an ellipsoid envelope with $a_1 = 2$ m, $b_1 = 1.8$ m, $c_1 = 1.5$ m, and $d_1 = 0$ m. The parameters of the ellipsoid envelope for the first failed spacecraft's solar panels are set to $a_2 = 6$ m, $b_2 = 1.2$ m, $c_1 = 1.2$ m, and $d_2 = 0$ m. The parameters of the ellipsoid envelope for the second failed spacecraft's solar panels are set to $a_2 = 2$ m, $b_2 = 1.2$ m, $c_1 = 1.2$ m, and $d_2 = 4$ m. The radius of the sphere envelope for the servicer spacecraft is $r_e = 0.5$ m.

In our simulations, the sampling period T_s is set to $T_s = 0.1$ s. The weighting matrices are selected as $Q_i = \mathbf{I}_3$ and $R_i = \mathbf{I}_3$. The maximum control input u_{\max} is set to $u_{\max} = 0.1$ m/s², and the maximum relative velocity v_{\max} is set to $v_{\max} = (0.5, 0.5, 0.5, 0.5, 0.5, 0.5)^T$ m/s. The value of the prediction horizon N_p is set to $N_p = 20$, and the value of control horizon N_c is set to $N_c = 10$. The value of maximum iteration number n is set to $n = 5$, and the trust region related parameters are set to $\rho = 0.9$ and $D_0 = [0.04, 0.04, \dots, 0.04, \dots, 0.04]^T$.

4.1. Approaching the First Type of a Failed Spacecraft. First, the effectiveness of the proposed MPC is analyzed through the approach simulation. The initial relative translational states $X(0)$ are set to $[25, -25.3, 0]^T$ m and $[0, -0.0027, 0]^T$ m/s. The initial angular velocities of the failed spacecraft are set to $\mathbf{w}_b^b = [2.2, 2.5, 2.3]^T$ deg/s. The simulation time is set to 100 s. The simulation results are shown in Figures 6 and 7.

TABLE 1: Parameters of the failed spacecraft.

Parameter	Value
Orbit parameter	$h = 35786$ km, $e = 0$, $\Omega = 0^\circ$, $i = 0^\circ$, $w = 0^\circ$, $f = 0^\circ$
Inertial moment	$I_t = \text{diag}(500, 600, 700)$ kg · m ²
Docking port	$\mathbf{r}_{\text{DP}}^b = [0, -3, 0]^T$ m
Initial rotation quaternion	$\mathbf{q}(0) = [1, 0, 0, 0]^T$

As shown in Figures 6(a)–6(c), it is evident that the desired trajectory to track the DP is time variant in target's LVLH frame, the approach process is implemented, and the trajectory tracking error in each direction is at 10^{-4} m magnitude. As depicted in Figure 6(d), the relative velocity in each direction is lower than 0.5 m/s, and the relative velocity constraint is satisfied. As shown in Figure 6(e), the control input in each direction is lower than 0.1 m/s² and the control input constraint is satisfied. As depicted in Figure 6(f), while approaching the target, the collision threshold always exceeds zero, thereby indicating that the center of the servicer lies outside the expanded two ellipsoids and that the collision constraint between the target and servicer is attained. Furthermore, the three-dimensional trajectories of servicer's center relative to the expanded two ellipsoids of the target and servicer's keep-out zone relative to target's keep-out zone are shown in Figure 7. Evidently, the collision constraint is satisfied.

Subsequently, the performances of MPC on target with different initial angular velocities are analyzed. In addition to the initial angular velocities $\mathbf{w}_b^b = [2.2, 2.5, 2.3]^T$ deg/s (case A) as discussed in the previous simulation, initial angular velocities $\mathbf{w}_b^b = [1.2, 1.0, 0.8]^T$ deg/s (case B) and $\mathbf{w}_b^b = [0, 0, 0]^T$ deg/s (case C) are analyzed for comparison purposes and all other simulation parameters are set as the same. Furthermore, the convergence time T and total control input ΔV are calculated at different initial angular velocities. It should be noted that the convergence time T is recorded when the distance between the servicer and the DP is less than 0.1 m. The analysis results are shown in Figure 8.

The servicer can arrive at the DP while satisfying above control input, relative velocity, and collision avoidance constraints in the aforementioned three cases. As shown in Figure 8, the angular velocity of the target significantly influences servicer's transfer trajectory. When the target exhibits low angular velocity, the total control input of the approaching is lower. Furthermore, the influence of the control horizon and predictive horizon on the MPC is also analyzed. With respect to each control horizon $N_c = \{5, 10, 20\}$, the convergence time and total control input are calculated at different control horizons $N_p = \{20, 30, 40, 50, 60\}$. The initial angular velocities of the target are also set to $\mathbf{w}_b^b = [2.2, 2.5, 2.3]^T$ deg/s. The simulation results are shown in Figure 9.

As shown in Figure 9, with increases in the predictive horizon, the convergence time for approaching increases and the total control input decreases. It is noted that the

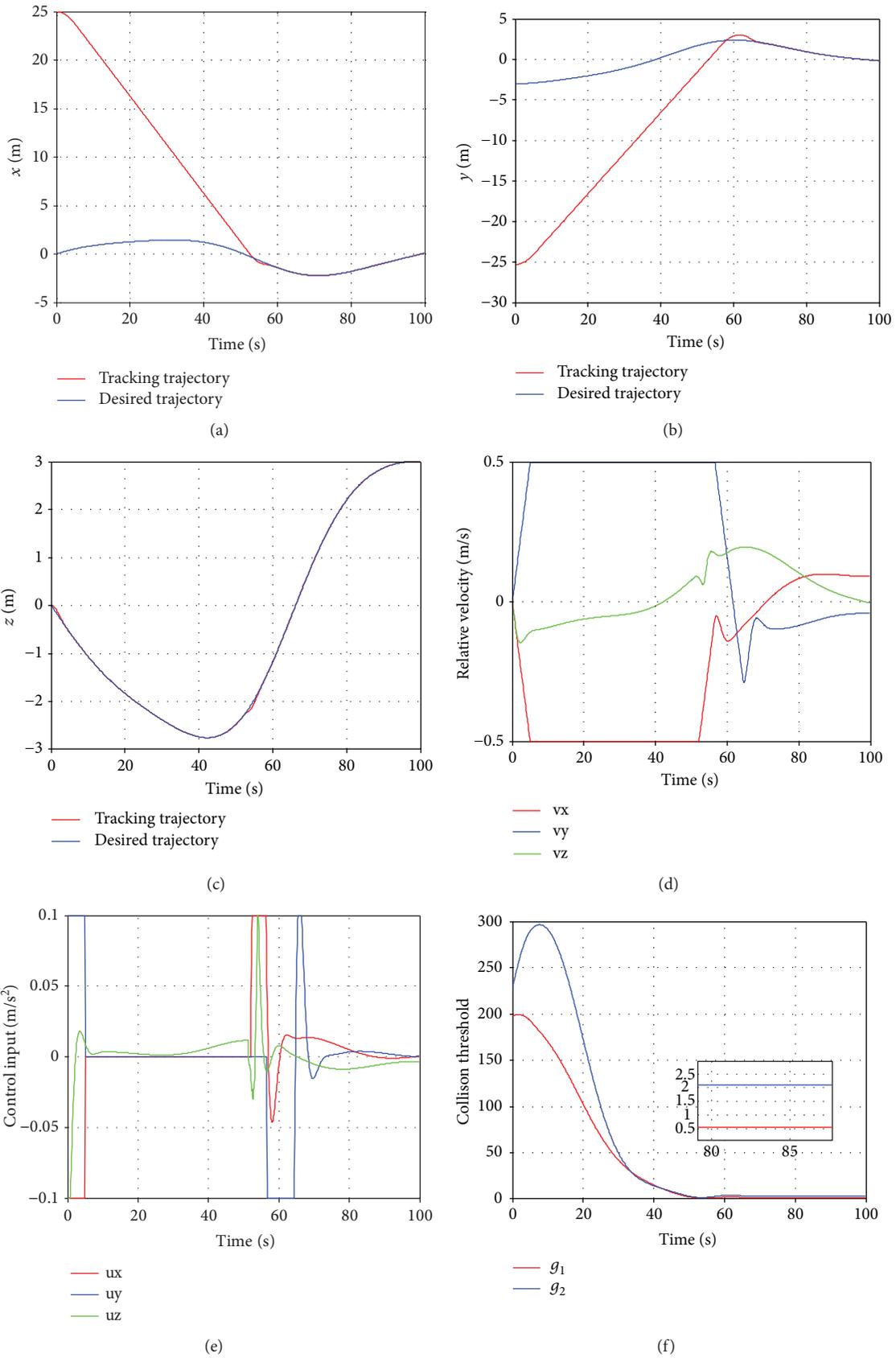


FIGURE 6: Temporal response of the relative position, relative velocity, control input, and collision threshold. (a) Temporal history along the x direction. (b) Temporal history along the y direction. (c) Temporal history along the z direction. (d) Temporal history of relative velocity. (e) Temporal history of the control input. (f) Temporal history of the collision threshold.

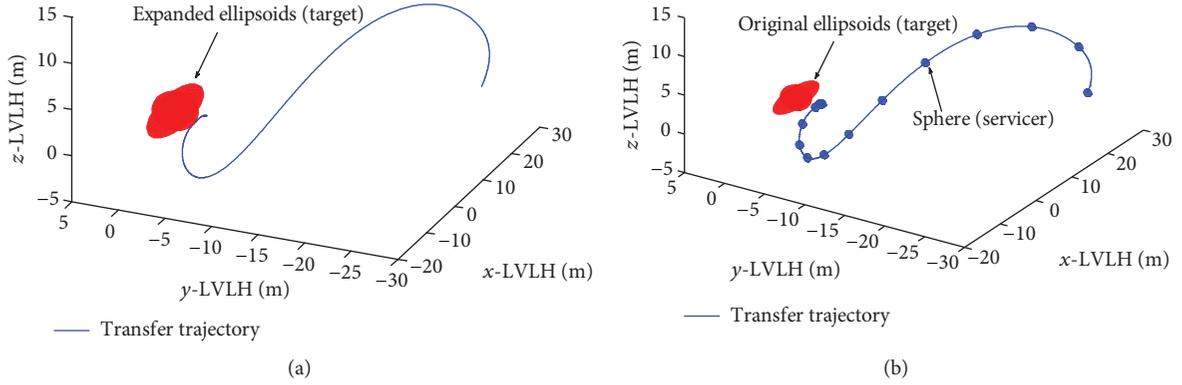


FIGURE 7: Three-dimensional plot of the approaching trajectories resolved in the target body-fixed frame. (a) Servicer's center relative to target's expanded ellipsoids. (b) Servicer's sphere relative to target's original ellipsoids.

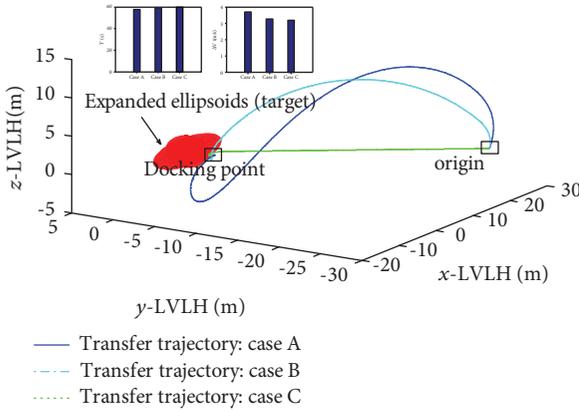


FIGURE 8: Three-dimensional plot of the transfer trajectories at target's different initial angular velocities resolved in the target body-fixed frame.

objective function in Eq. (28) is consist of the tracking error and the control effort. The increase of predictive horizon N_p denotes more predicted states, leading to less state error and less control input to modify the state error. Accordingly, the required total control input is less and the convergence time will be longer. With increases in the control horizon N_c , the convergence time for approaching decreases, while the total control input and the accuracy of tracking the DP increase. This is because more predicted control inputs have been used to modify the state error in every sampling instant, leading to a shorter convergence time and higher tracking accuracy. Furthermore, the increases in the predictive horizon and control horizon, denoting more predicted states and control inputs, lead to a higher computation load when solving the underlying optimization problem in MPC. Therefore, while implementing the on-orbit missions, proper MPC parameters should be selected based on mission requirements.

4.2. Approaching the Second Type of a Failed Spacecraft. With respect to the simulation that involves approaching the second type of the failed spacecraft, the parameters of the

MPC are set as identical to those in the previous simulation. The parameters of the second failed spacecraft's keep-out zone are discussed at the beginning of Section 4. Furthermore, in order to confirm that the proposed method is generally applicable, the designed MPC dealing with the servicer at different initial positions, such as $\mathbf{r}_0 = [25, -25.3, 0]^T$ m (Sat 1), $\mathbf{r}_0 = [25, 25.3, 0]^T$ m (Sat 2), $\mathbf{r}_0 = [-25, 25.3, 0]^T$ m (Sat 3), and $\mathbf{r}_0 = [-25, -25.3, 0]^T$ m (Sat 4), is analyzed. The initial angular velocities of the target are also set to $\mathbf{w}_b^b = [2.2, 2.5, 2.3]^T$ deg/s. Simulation results on the transfer trajectory are shown in Figure 10.

As shown in Figure 10, the proposed MPC exhibits the ability to drive the servicers at different positions to the DP of the target. The convergence times for each servicer are 57.9 s, 59 s, 69.2 s, and 69.2 s. The total control inputs for each servicer are 3.69 m/s, 4.23 m/s, 5.46 m/s, and 4.83 m/s. The temporal response of the collision threshold of each servicer is shown in Figure 11.

As shown in Figure 11, the collision avoidance constraint is evidently satisfied during the approaching process. As shown in Figure 12, required constraints, such as control input and relative velocity, are also satisfied. It should be noted that when compared with linearizing the collision avoidance constraint at the DP in which only half of the plane is the collision avoidance zone, the proposed method for handling collision avoidance constraint exhibits a better performance in which the space outside the expanded ellipsoids is the collision avoidance zone.

In previous simulation analysis, the controller uncertainties are neglected and the measurement of relative states and the control input of the system are assumed to be completely accurate. In order to evaluate the performance of the proposed MPC for real approaching missions, navigation and control noise are considered. Considering unknown but bounded navigation noise, the measurement-relative translational states $\hat{\mathbf{X}}(k)$ can be represented as

$$\begin{cases} \hat{\mathbf{X}}(k) = \mathbf{X}(k) + \mathbf{e}(k), \\ \hat{\mathbf{X}}(k+1) = \mathbf{X}(k+1) + \mathbf{e}(k+1), \end{cases} \quad (45)$$

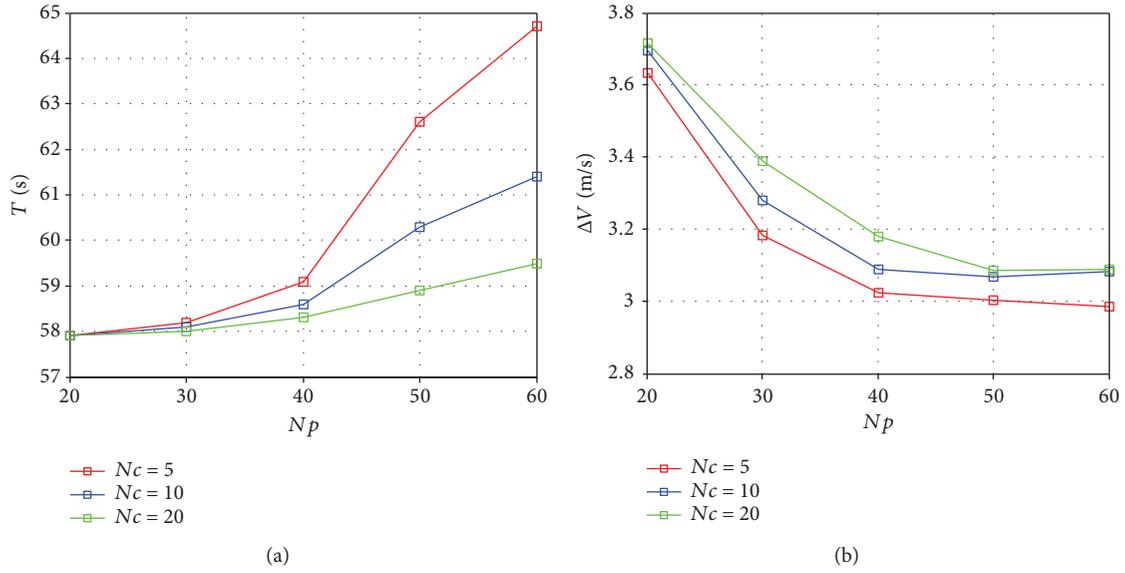
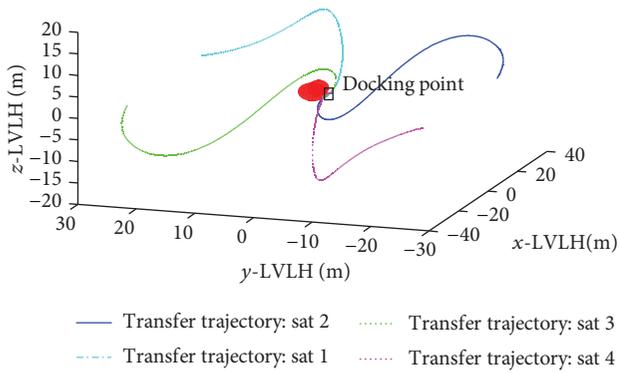

 FIGURE 9: Influences of the predictive horizon N_p and control horizon N_c on (a) convergence time and (b) total control input.


FIGURE 10: Three-dimensional plot of the transfer trajectories with the servicer at different initial positions.

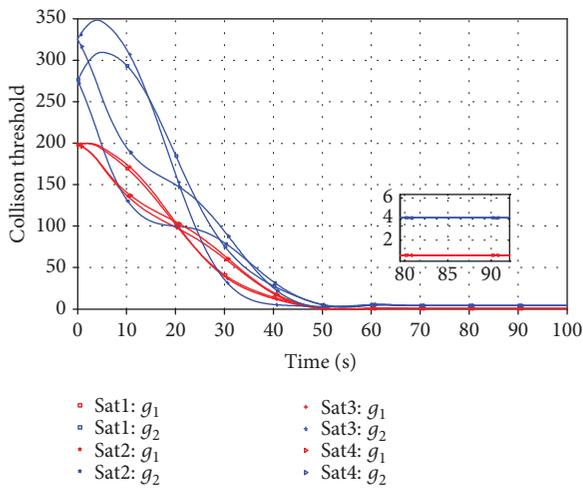


FIGURE 11: Temporal response of the collision threshold of each servicer.

where $\mathbf{e}(k) = [\mathbf{e}_p(k); \mathbf{e}_v(k)]$ and $\mathbf{e}_p(k)$ and $\mathbf{e}_v(k)$ are the measurement errors of position and velocity, respectively. Considering that the control output noise was proportional to the control output [33], the control input $\hat{U}(k)$ applied to the servicer can be represented as

$$\hat{U}(k) = (\mathbf{I}_3 + \Lambda(k))U(k), \quad (46)$$

where $\Lambda(k) = \text{diag}(\Lambda_x(k), \Lambda_y(k), \Lambda_z(k))$ is a three-dimensional diagonal matrix.

Combining Eqs. (4), (45), and (46), the prediction of relative states should be represented as

$$\hat{X}(k+1) = A_d \hat{X}(k) + B_d (I + \Lambda(k))U(k) + \mathbf{e}(k+1) - A_d \mathbf{e}(k). \quad (47)$$

However, navigation and control noise cannot be measured. Then, Eq. (4) with measurement $\hat{X}(k)$ and control input $U(k)$ was used to predict relative states. The parameters of the MPC are set as identical to those in the previous simulation, and the initial position of the servicer is $\mathbf{r}_0 = [25, -25.3, 0]^T$ m (Sat 1). The initial angular velocities of the target are also set to $\mathbf{w}_b^b = [2.2, 2.5, 2.3]^T$ deg/s.

A Monte-Carlo analysis is performed to illustrate the influence of uncertainties on the MPC performance. It is assumed that elements in \mathbf{e}_p , \mathbf{e}_v , and Λ follow Gaussian distribution, and their accuracies (3σ) are defined as E_p , E_v , and E_u , respectively. When developing Monte-Carlo analysis, 1000 simulations in each of two conditions such as $[E_p, E_v, E_u] = [0.005 \text{ m}, 10^{-4} \text{ m/s}, 0.01]$ and $[E_p, E_v, E_u] = [0.02 \text{ m}, 10^{-3} \text{ m/s}, 0.02]$ are conducted. Reconsidering the tracking error defined in Eq. (15), the average tracking

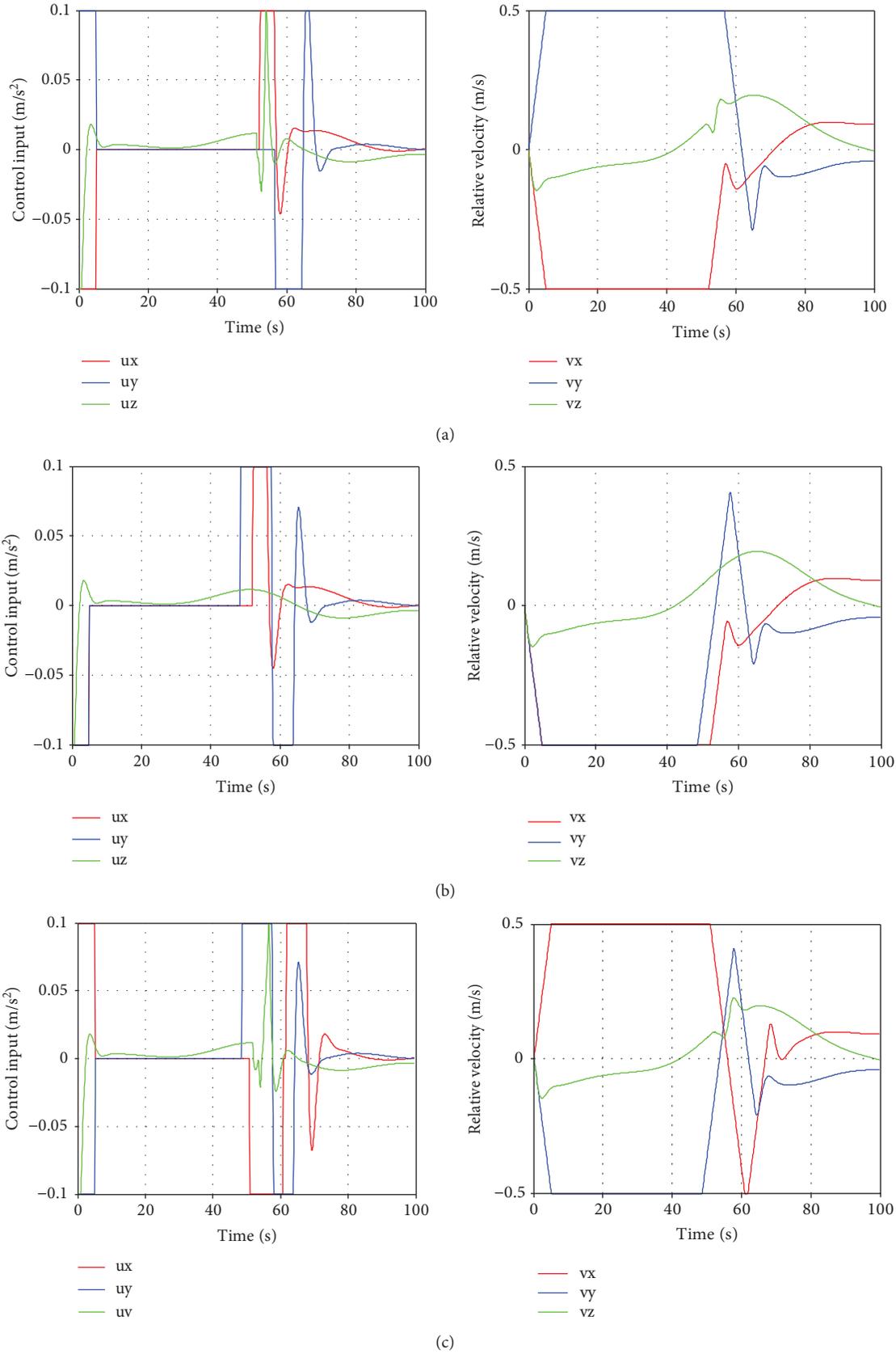


FIGURE 12: Continued.

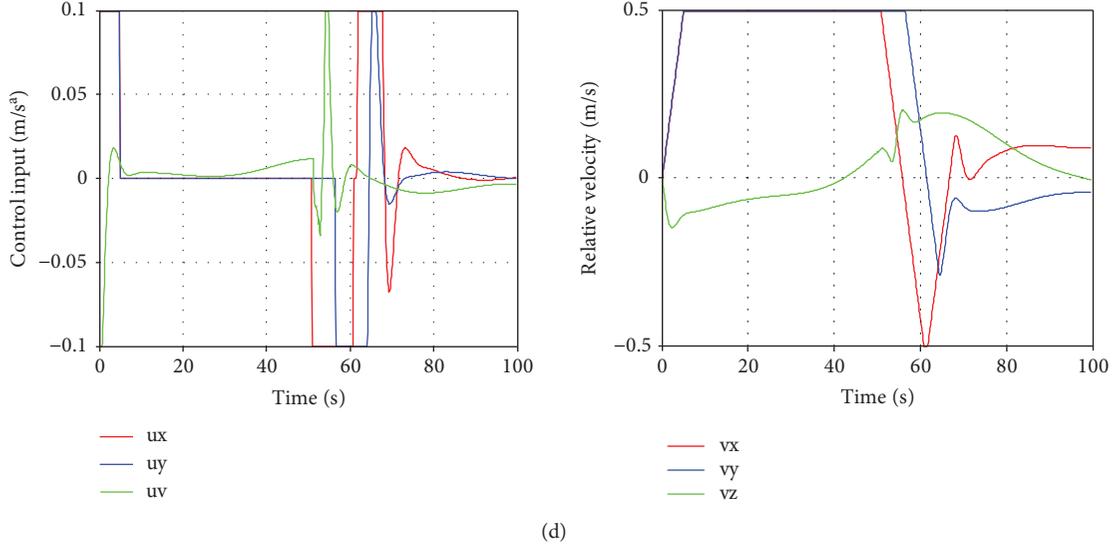


FIGURE 12: Temporal response of the control input (left) and relative velocities (right) for (a) Sat 1, (b) Sat 2, (c) Sat 3, and (d) Sat 4.

error $\delta\tilde{r}$ is considered to evaluate the influence of uncertainties, which is defined as

$$\delta\tilde{r} = \frac{\int_{t_s}^{t_f} \|\delta\mathbf{r}(t)\| dt}{t_f - t_s}, \quad (48)$$

where $t_f = 100$ s is the end time of simulation and $t_s = 80$ s is designed to start recording the tracking error. The Monte-Carlo simulation results regarding the average tracking error $\delta\tilde{r}$ are shown in Figure 13.

When developing Monte-Carlo analysis, the required constraints such as control input, relative velocity, and collision avoidance in every simulation are satisfied. When there are no navigation and control noise, the average tracking error is approximately 0.5 mm. As shown in Figure 13, when the uncertainties are $[E_p, E_v, E_u] = [0.005 \text{ m}, 10^{-4} \text{ m/s}, 0.01]$ and $[E_p, E_v, E_u] = [0.02 \text{ m}, 10^{-3} \text{ m/s}, 0.02]$, the average tracking errors are approximately 1.5 mm and 7 mm, respectively. With the increase of uncertainties, the average tracking error also becomes larger. The effectiveness of the MPC in handling navigation and control noise is also proved. Based on the above analyses, it is concluded that the proposed MPC strategy can drive the servicer to the DP of a tumbling target while satisfying various constraints and providing capability in handling navigation and control noises.

5. Conclusion

A model predictive control (MPC) method for a servicer spacecraft autonomously approaching a tumbling failed spacecraft is presented in the study. The objective of the proposed MPC involves driving the servicer tracking the time-variant motion of the docking port of a target by considering collision avoidance, control input saturation, and velocity constraints. The relative translation is predicted by the CW equation, and target's attitude is predicted by a derived attitude dynamics. Sufficient conditions for collision avoidance

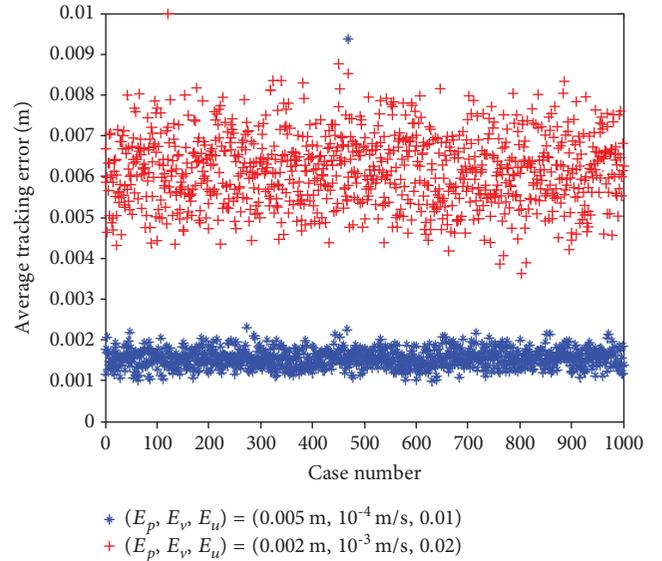


FIGURE 13: Performance comparison in terms of controller uncertainties.

are derived by using the form of concave constraint. The underlying optimization program for the implementation of MPC is converted into a convex optimization problem, and this is solved by using sequential convex programming. The process of the approach is simulated to evaluate the performance of the MPC strategy, and a few main contributions are obtained as follows:

- (1) Given the complex configurations of two types of the failed spacecraft, a double-ellipsoid envelope is designed to model the keep-out zone. Sufficient conditions with simple mathematical expressions for collision avoidance are derived. The proposed modeling method for collision avoidance can be extended to other collision avoidance missions

- (2) The convexification method for the collision avoidance constraints and the sequential convex programming for solving the underlying optimization problem in MPC are effective. The proposed MPC exhibits the capacity to handle the problem of tracking the time-varying docking point while handling various constraints

Appendix

A. The Derivation of Eq. (34)

Eq. (32) can be rewritten as follows:

$$\begin{aligned} g_m^i(U_s(k)) = & \left\{ R_{tb}(k+i)H(i)C_sA_sX(k) - [d_m, 0, 0]^T \right. \\ & \left. + R_{tb}(k+i)H(i)C_sB_sU_s(k) \right\}^T \times L_m \\ & \times \left\{ R_{tb}(k+i)H(i)C_sA_sX(k) - [d_m, 0, 0]^T \right. \\ & \left. + R_{tb}(k+i)H(i)C_sB_sU_s(k) \right\} - 1 > 0, \\ & i = 1, 2, \dots, N_p, \quad m = 1, 2. \end{aligned} \quad (\text{A.1})$$

Then, define $G_m(i)$ as $G_m(i) = R_{tb}(k+i)H(i)C_sA_sX(k) - [d_m, 0, 0]^T$ and $F(i)$ as $F(i) = R_{tb}(k+i)H(i)C_sB_s$, Eq. (A.1) can be rewritten as follows:

$$\begin{aligned} g_m^i(U_s(k)) = & [G_m(i) + F(i)U_s(k)]^T \times L_m \\ & \times [G_m(i) + F(i)U_s(k)] - 1 > 0, \\ & i = 1, 2, \dots, N_p, \quad m = 1, 2. \end{aligned} \quad (\text{A.2})$$

Then, define $\tilde{G}_m(i)$ as $\tilde{G}_m(i) = 2G_m(i)^T L_m F(i)$ and $\tilde{F}_m(i)$ as $\tilde{F}_m(i) = F(i)^T L_m F(i)$, Eq. (A.2) can be expanded as follows:

$$\begin{aligned} g_m^i(U_s(k)) = & U_s(k)^T \tilde{F}_m(i) U_s(k) + \tilde{G}_m(i) U_s(k) \\ & + G_m(i)^T L_m G_m(i) - 1 > 0, \\ & i = 1, 2, \dots, N_p, \quad m = 1, 2. \end{aligned} \quad (\text{A.3})$$

It is noted that Eq. (A.3) is the same as Eq. (34).

B. The Derivation of Eq. (36)

Eq. (25) can be rewritten as follows:

$$(\mathbf{I}_3; -\mathbf{I}_3)D_d X(k) \leq v_{\max}, \quad (\text{B.1})$$

where the matrices v_{\max} and D_d are defined as follows:

$$D_d = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad (\text{B.2})$$

$$v_{\max} = (\dot{x}_{\max} \dot{y}_{\max} \dot{z}_{\max} \dot{x}_{\max} \dot{y}_{\max} \dot{z}_{\max})^T.$$

For the implementation of MPC, the velocity sequence constraints can be represented as follows:

$$\begin{pmatrix} \mathbf{I}_{3N_p}; -\mathbf{I}_{3N_p} \end{pmatrix} \begin{pmatrix} D_d & 0 & 0 & \cdots & 0 \\ 0 & D_d & 0 & \cdots & 0 \\ 0 & 0 & D_d & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \cdots & D_d \end{pmatrix} \begin{pmatrix} X(k+1|k) \\ X(k+2|k) \\ X(k+3|k) \\ \vdots \\ X(k+N_p|k) \end{pmatrix} \leq \begin{pmatrix} v_{\max} \\ v_{\max} \\ v_{\max} \\ \vdots \\ v_{\max} \end{pmatrix}, \quad (\text{B.3})$$

where

$$D_s = \begin{pmatrix} D_d & 0 & 0 & \cdots & 0 \\ 0 & D_d & 0 & \cdots & 0 \\ 0 & 0 & D_d & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \cdots & D_d \end{pmatrix} = D_d \otimes \mathbf{I}_{N_p}, \quad (\text{B.4})$$

$$V_{\max} = (v_{\max}^T, \dots, v_{\max}^T)^T.$$

Substituting Eq. (B.4) into Eq. (B.3), Eq. (B.3) can be rewritten as follows:

$$(\mathbf{I}_{3N_p}; -\mathbf{I}_{3N_p})D_s(A_sX(k) + B_sU_s) \leq V_{\max}. \quad (\text{B.5})$$

It is noted that Eq. (B.5) is the same as Eq. (36).

C. The Derivation of Eq. (44)

Eq. (40) can be rewritten as follows:

Problem 2.

$$\begin{aligned} \min_{U_s(k)} \quad & \Gamma(k) = U_s(k)^T W(k) U_s(k) + P(k) U_s(k) + M(k)^T Q M(k) \\ \text{subject to} \quad & (\mathbf{I}_{3N_c}; -\mathbf{I}_{3N_c}) U_s(k) \leq U_{\max} \\ & (\mathbf{I}_{3N_p}; -\mathbf{I}_{3N_p}) G_s B_s U_s(k) \\ & \leq V_{\max} - (\mathbf{I}_{3N_p}; -\mathbf{I}_{3N_p}) G_s A_s X(k) - \nabla g_m^i(U_s^{[n]}(k))^T U_s(k) \\ & < -\nabla g_m^i(U_s^{[n]}(k))^T U_s^{[n]}(k) + g_m^i(U_s^{[n]}(k)), \\ & i = 1, 2, \dots, N_p, \quad m = 1, 2. \end{aligned} \quad (\text{C.1})$$

Considering the inequality constraint about the trust region, Eq. (43) can be rewritten as follows:

$$\mathbf{I}_{3N_c} \left(U_s(k) - U_s^{[n]}(k) \right) \leq \rho^n D_0 - \mathbf{I}_{3N_c} \left(U_s(k) - U_s^{[n]}(k) \right) \leq \rho^n D_0. \quad (\text{C.2})$$

Then, substituting Eq. (C.2) into Eq. (C.1), Problem 2 that considers the trust region is subsequently converted into Problem 3:

- [23] H. Park, S. Di Cairano, and I. Kolmanovsky, "Model predictive control of spacecraft docking with a non-rotating platform," in *IFAC 18th World Congress*, pp. 8485–8490, Milano, Italy, 2011.
- [24] A. Weiss, M. Baldwin, R. S. Erwin, and I. Kolmanovsky, "Model predictive control for spacecraft rendezvous and docking: strategies for handling constraints and case studies," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 4, pp. 1638–1647, 2015.
- [25] U. Eren, A. Prach, B. B. Koçer, S. V. Raković, E. Kayacan, and B. Açıkmeşe, "Model predictive control in aerospace systems: current state and opportunities," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 7, pp. 1541–1566, 2017.
- [26] H. Park, R. Zappulla, C. Zagarisz, J. V. Llop, and M. Romano, "Nonlinear model predictive control for spacecraft rendezvous and docking with a rotating target," in *Proceedings of the 27th AAS/AIAA Spaceflight Mechanics Meeting*, San Antonio, TX, USA, 2017.
- [27] Q. Li, J. Yuan, B. Zhang, and C. Gao, "Model predictive control for autonomous rendezvous and docking with a tumbling target," *Aerospace Science and Technology*, vol. 69, pp. 700–711, 2017.
- [28] P. Li and Z. H. Zhu, "Model predictive control for spacecraft rendezvous in elliptical orbit," *Acta Astronautica*, vol. 146, pp. 339–348, 2018.
- [29] D. Morgan, S. J. Chung, and F. Y. Hadaegh, "Model predictive control of swarms of spacecraft using sequential convex programming," *Journal of Guidance, Control, and Dynamics*, vol. 37, no. 6, pp. 1725–1740, 2014.
- [30] X. Liu and P. Lu, "Solving nonconvex optimal control problems by convex optimization," *Journal of Guidance, Control, and Dynamics*, vol. 37, no. 3, pp. 750–765, 2014.
- [31] A. Richards, T. Schouwenaars, J. P. How, and E. Feron, "Spacecraft trajectory planning with avoidance constraints using mixed-integer linear programming," *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 4, pp. 755–764, 2002.
- [32] X. Wang, Z. Wang, and Y. Zhang, "Stereovision-based relative states and inertia parameter estimation of noncooperative spacecraft," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, pp. 1–14, 2018.
- [33] H.-C. Lim and H. Bang, "Adaptive control for satellite formation flying under thrust misalignment," *Acta Astronautica*, vol. 65, no. 1-2, pp. 112–122, 2009.



Hindawi

Submit your manuscripts at
www.hindawi.com

