

RESEARCH ARTICLE

Dual arm coordination of redundant space manipulators mounted on a spacecraft

Serdar Kalaycioglu  and Anton de Ruiter

Department of Aerospace Engineering, Toronto Metropolitan University, Toronto, Canada

Corresponding author. Serdar Kalaycioglu; Email: skalay@torontomu.ca

Received: 11 November 2022; **Revised:** 20 February 2023; **Accepted:** 2 April 2023; **First published online:** 29 May 2023

Keywords: NMPC; dual arm coordination

Abstract

The paper addresses a significant challenge in on-orbit robotics servicing and assembly, which is to overcome the saturation setback of force/torque on robot joint and spacecraft actuators during the post-capture stage while controlling a target spacecraft with uncontrolled large angular and linear momentums. The authors propose a novel solution based on two robust and efficient control algorithms: Optimal Control Allocation (OCA) and Non-linear Model Predictive Control (NMPC). Both algorithms aim to minimize joint torques, spacecraft actuator moments, contact forces, and moments of a compound redundant system that includes a common payload (target spacecraft) grasped by dual n -degree space robotics manipulators mounted on a chaser spacecraft. The OCA algorithm minimizes a quadratic cost function using only the current states and the system dynamics, but the NMPC also considers the future state estimates and the control inputs over a specified prediction horizon. It is computationally more involved but provides superior results in reducing joint torques. The literature to date in application of MPC to robotics mainly focuses on linear models but the dual arm coordination is highly non-linear and there is no MPC application on dual arm coordination. The proposed discretized technique offers exact realizations (of a non-linear model) with elegance and simplicity and yet considers the full non-linear model of the dual arm coordinating system. It is computationally very efficient. The computer simulation results show that the proposed algorithms work efficiently, and the minimum torques, contact forces, and moments are realized. The developed algorithm also is very efficient in tracking problems.

1. Introduction

In the last two decades, there has been a significant interest in autonomous robotics spacecrafts equipped with the tools, technologies, and techniques (i) to extend satellite(s)’ lifespan – even if they were not designed to be serviced on orbit and (ii) to assemble large space structures and manufacture on orbit. These on-orbit robotics capabilities will eliminate volume limits imposed by launch vehicles and replace some astronaut extravehicular activity tasks with precision robotics. Thus, on-orbit robotics servicing and assembly are deemed to be a key solution and technology due to existing launch & deployment constraints. These robotics technologies will also enable new architectures and capabilities for a wide range of government and commercial missions including in-space construction of large space structures, communications antennae, and telescopes.

Recently, OSAM-1 (formerly known as Restore-1), a robotics spacecraft, is being jointly developed by NASA and MAXAR Technologies Inc. and soon it will be ready to carry out various on-orbit assembly and servicing tasks such as fabrication, assembly of antennas, and refueling of satellites. ARCHINAUT (OSAM-2) developed by NASA and Made in Space Inc. features multiple robotics arms for on-orbit precision manufacturing and assembly [1–3]. Some other examples include the Gateway Moon Space Station by NASA and international partners, Moon Village and Mars Exploration by the European Space Agency, the OrbitalHub by the German Space Agency, and many other large space structures and telescopes [4].

On-orbit robotics servicing and assembly missions [5] generally include several modes of operations such as (a) orbital approach and rendezvous, (b) robotics arms deployment and pre-contact approach, (c) contact and capture, and (d) post-capture operations.

In the pre-contact stage, space manipulators mounted on the chaser spacecraft approach and gradually follow the motion of the target payload. In the contact stage, the space robots are free floating. Finally, during the post-capture phase, space robots suppress the target's uncontrolled motion and establish a new trajectory [6, 7]. Matsumoto et al. [8] proposed the use of the natural dynamics to complete a passive fly-by approach and an optimal trajectory for close-range rendezvous with a rotating satellite. The study also considered issues such as collision avoidance between the manipulator and the target satellite. Ma et al. [9] designed an optimal trajectory for a spacecraft to approach a tumbling satellite by minimizing time and fuel. A cubic spline-based collision avoidance algorithm for a robotics spacecraft was developed by Fejzic [10]. Xin and Pan [11] developed an optimal control of spacecraft approaching a tumbling target. They minimized the flexible motion induced by large angular maneuvers using a non-linear optimal control technique. Vafa and Dubowsky [12] proposed a special cyclic motion trajectory of a manipulator's joints to change the base spacecraft's orientation. The path planning problem of a free-floating target with a manipulator having angular momentum was addressed by Yamada et al. in ref. [13, 14].

Gasbarri [15] reported that the dynamics modeling and pre-contact motion planning of a space robot are significantly more complicated than those of fixed-base manipulators due to the "dynamic coupling" between the manipulator arms and the base platform. The study investigated pre-contact and post-capture phases and proposed a scheme to control either joint torques or moments applied to the chaser spacecraft.

The issue of stability and control during the initial stage of contact and capture has been extensively studied and published. Cyril [16] developed a dynamics model of the compound system for a post-capture phase and proposed a model-based feedback linearization control scheme. Dimitrov [17] presented two new control laws regarding management of the total angular momentum of a compound system during a post-capture phase. Wang [18] also emphasized on the post-capture phase and presented a novel sliding mode control scheme for a tether-connected robotics system.

Suresh [19] presented a robust control theory based on a dissipative function for a stable compound system during an autonomous capture. Aghili [20] developed a control model which dissipated unwanted linear and angular momentum to bring a tumbling target spacecraft to rest. Xu [21] proposed a control strategy to maneuver the captured spacecraft along with the mounted space robot during deployment of a target spacecraft. Cheah [22] proposed a feedback control law which can deal with uncertain system parameters and still unload the angular momentum of the compound system. Dixon [23] presented an adaptive regulation in the case of uncertain dynamics parameters of space robots. Cheah [24] developed an approximate Jacobian adaptive control scheme for robots. Wang [25] presented a Jacobian adaptive controller for free-floating space robots. Zhang [26] developed a modified adaptive sliding mode control algorithm to reduce the unknown angular momentum of a target spacecraft.

Schneider [27] studied a modified impedance control technique called Object Impedance Control for fixed-base multiple robots manipulating a common object. In this study, the common object followed a reference mechanical impedance. More recently, Moosavian [28] developed the Multiple Impedance Control algorithm for several cooperating robots which manipulate a common payload.

The Model Predictive Control (MPC) framework simultaneously allows optimization of the given performance index in conjunction with the accounting of the system dynamics and constraints as demonstrated in [29–34]. In a study conducted by Gangapersaud et al. [35], a novel detumbling strategy is presented to detumble targets without prior knowledge of their inertial parameters. Detumbling of the target is achieved by controlling the space robot to follow a reference force/torque that is designed to detumble the target while respecting force/torque limits at the end-effector, without the use of the target's inertial parameters. To ensure a stable detumbling strategy, a robust compensator is developed predicated on bounds of the target's unknown inertial parameters.

One of the most practical and theoretical challenges in implementing a robust model of MPC is the ability to tackle the various uncertainties that can affect the performance of the system. Due to

the receding horizon nature of the model, standard MPC can provide a level of robustness that is not available in other methods [36]. Unfortunately, the literature showed that the standard MPC cannot provide adequate performance in complicated robotics systems [37]. To address this issue, many research efforts have been carried out on developing robust MPC that can provide robust stability and constraint satisfaction in the presence of uncertainties [38–42]. Over the last few years, the scope and the real-time capabilities of Non-linear Model Predictive Control (NMPC) have been greatly improved. There has been remarkable increase in the number of tools that enable fast solutions of the various non-linear programs, such as gradient use (Englert et al. [43]), input parameterization (Rathai [44]), and real-time iterations (Quirynen et al. [45]). The application of NMPS for free-floating space manipulator is provided in refs. [46–48].

However, regardless of the applied control theory, it is observed that one of the most crucial problems and technical challenges for spacecraft robotics in the literature [49] is the saturation of force and torque of robot joint and spacecraft actuators during the post-capture stage while controlling a target spacecraft due to its uncontrolled large angular and linear momentums. In the literature [50], the manipulator damps out the target's angular and linear momentums as quickly as possible subject to the constraint that the magnitude of the exerted force and torque remain below their prespecified values in the post-capture phase.

This paper directly focuses on the very issue and presents a novel solution with two robust and efficient control algorithms which minimize the joint torques and contact forces while maintaining the desired trajectory of the payload (i.e., target spacecraft) during the post-capture phase.

The literature to date in the area of application of MPC to robotics mainly focuses on linear models but the dual arm coordination is highly non-linear, and there is no robust MPC application on dual arm coordination. The proposed discretized technique offers exact realizations (of a non-linear model) with elegance and simplicity and yet takes into account the full non-linear model of the dual arm coordinating system. Furthermore, the cost function selected in NMPC is unique focusing on a practical problem, that is minimization of joint and spacecraft actuators moments during a post-capture phase while unloading momentum and bringing a target spacecraft to rest. In addition, the method is further extended to accommodate the pre-contact phase as well by simply changing the cost function to incorporate the minimum joint rates.

As far as the spacecraft hardware and software implementations are concerned, the space community is divided on the matter of centralized versus distributed processor architecture [51]. After evaluating the proposed control algorithms, we have determined that a distributed architecture is the optimal choice for our application. This architecture offers several benefits, including parallel control of the spacecraft and robots, reduced computer speed demands, and enhanced reliability through inherent redundancy. Our proposed architecture consists of three main processors: (i) Spacecraft Attitude Determination and Control System (ADACS) and Robot Control System (RCS), (ii) Command and Data Handling (C&DH), and (iii) Communications Processor.

The control algorithm resides in the ADACS & RCS computer, responsible for controlling the spacecraft's 3-axis momentum attitude, orbital maneuvers, and end-effector position and orientation control of the robots. The algorithm will also collect sensory data from sources such as the star trackers, joint encoders, and the vision system, among others. A star tracker allows for precise control of the spacecraft's attitude, with accuracy of 0.01° or better. The spacecraft is equipped with three orthogonal reaction wheels and a combination of torque rods and magnetometer. The vision system determines the pose of the target spacecraft. The Computer Software Configuration Items (CSCIs) include a range of modules, such as OCA, NMPC, Forward and Inverse Kinematics Functions, the Robot Dynamics Function, Jacobian, Jacobian Rate, Impedance Control Trajectory Generation Function, and more. These CSCIs are located in the ADACS and RCS computer.

Section 2 presents the theoretical formulations, including the kinematics, dynamics model of the compound system which consists of a chaser spacecraft and two n -degree redundant manipulators and a target spacecraft. Two generic optimal control algorithms (utilizing (i) only the current states OCA or (ii) the current and the future predicted states – NMPC) are formulated to obtain optimal control actions for joint torques and the contact force/moments, respectively, to minimize a specified performance index.

Simulation results and discussion are presented in Section 3, and some concluding remarks are provided in Section 4.

2. Theoretical formulations

2.1. Description of the system

The system under consideration consists of a rigid spacecraft (chaser), two *n*-DOF redundant manipulators, and a rigid payload (target). An example of the system of a rigid spacecraft with two *n*-degree robots and a rigid payload is illustrated in Fig. 1(a). The geometry of the robotics arm is presented in Fig. 1(b–c). The parameters of the spacecraft robotics system used in the simulations are provided in Table I in Appendix.

Let the $\tilde{\mathbf{R}}_c$, the position vector and true anomaly β , define the location of the instantaneous center of mass *C* of the spacecraft with respect to the inertial coordinate system, *X*, *Y*, *Z* having its origin at the center of Earth. It is assumed that the inertial coordinate system is set up with *x*-axis pointing to perigee and *z*-axis to orbit normal, and the *y*-axis completes the right-hand triad.

The motion is described by using a set of coordinate axes *x*, *y*, *z* fixed to the central body such that they are roll, pitch, and yaw axes, respectively. These in turn are obtained from a set of orbital coordinates axes *x*₀, *y*₀, *z*₀ through three rotations as described in Section 2.2. Here *z*₀-axis is directed towards the center of Earth, *y*₀-axis is along the orbit normal, and *x*₀-axis completes the right-hand triad. For convenience, a set of local coordinates is also assigned to each joint of the robot manipulators where *z*_{*i*} is parallel to the axis of rotation and *x*_{*i*} is along the link and *y*_{*i*} completes the right-hand triad.

2.2. Kinematics of the system

The rotational transformation from orbital to body axes is represented by a 1-3-2 Euler sequence, with angles ϕ , ψ , and θ , respectively. The components of the body-attached angular velocity along *x*, *y*, *z* axes can be written in terms of $\dot{\phi}$, $\dot{\psi}$, $\dot{\theta}$ as follows [52, 53]

$$\tilde{\boldsymbol{\omega}}_c = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} \dot{\theta} \cos \theta \cos \varphi - \dot{\phi} \sin \theta - (\cos \theta \sin \varphi \cos \varnothing + \sin \theta \sin \varnothing) \omega_o \\ -\dot{\theta} \sin \varphi + \dot{\theta} - \cos \varphi \cos \varnothing \omega_o \\ \dot{\theta} \sin \theta \cos \varphi + \varphi \cos \theta - (\cos \theta \sin \varphi \cos \varnothing - \cos \theta \sin \varnothing) \omega_o \end{bmatrix} \tag{1}$$

where ω_o is the orbital angular velocity The transformation matrix \mathbf{R} which describes the relative rotational motion between the orbital coordinate frame (*x*₀, *y*₀, *z*₀) and the spacecraft attached coordinate frame (*x*, *y*, *z*) can be written as:

$$\mathbf{R} = \underline{\Theta}_y \underline{\Psi}_z \underline{\Phi}_x \mathbf{R} = \begin{bmatrix} \cos \theta \cos \varphi & (\cos \theta \sin \varphi \cos \varnothing + \sin \theta \sin \varnothing) & (\cos \theta \sin \varphi \sin \varnothing - \sin \theta \cos \varnothing) \\ \sin \varphi & \cos \varphi \cos \varnothing & \cos \varphi \sin \varnothing \\ \sin \theta \cos \varphi & (\sin \theta \sin \varphi \cos \varnothing - \cos \theta \sin \varnothing) & (\sin \theta \sin \varphi \sin \varnothing + \cos \theta \cos \varnothing) \end{bmatrix} \tag{2}$$

$$\underline{\Phi}_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \varnothing & \sin \varnothing \\ 0 & -\sin \varnothing & \cos \varnothing \end{bmatrix} \tag{2a}$$

$$\underline{\Psi}_z = \begin{bmatrix} \cos \varphi & \sin \varphi & 0 \\ -\sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2b}$$

$$\underline{\Theta}_y = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \tag{2c}$$

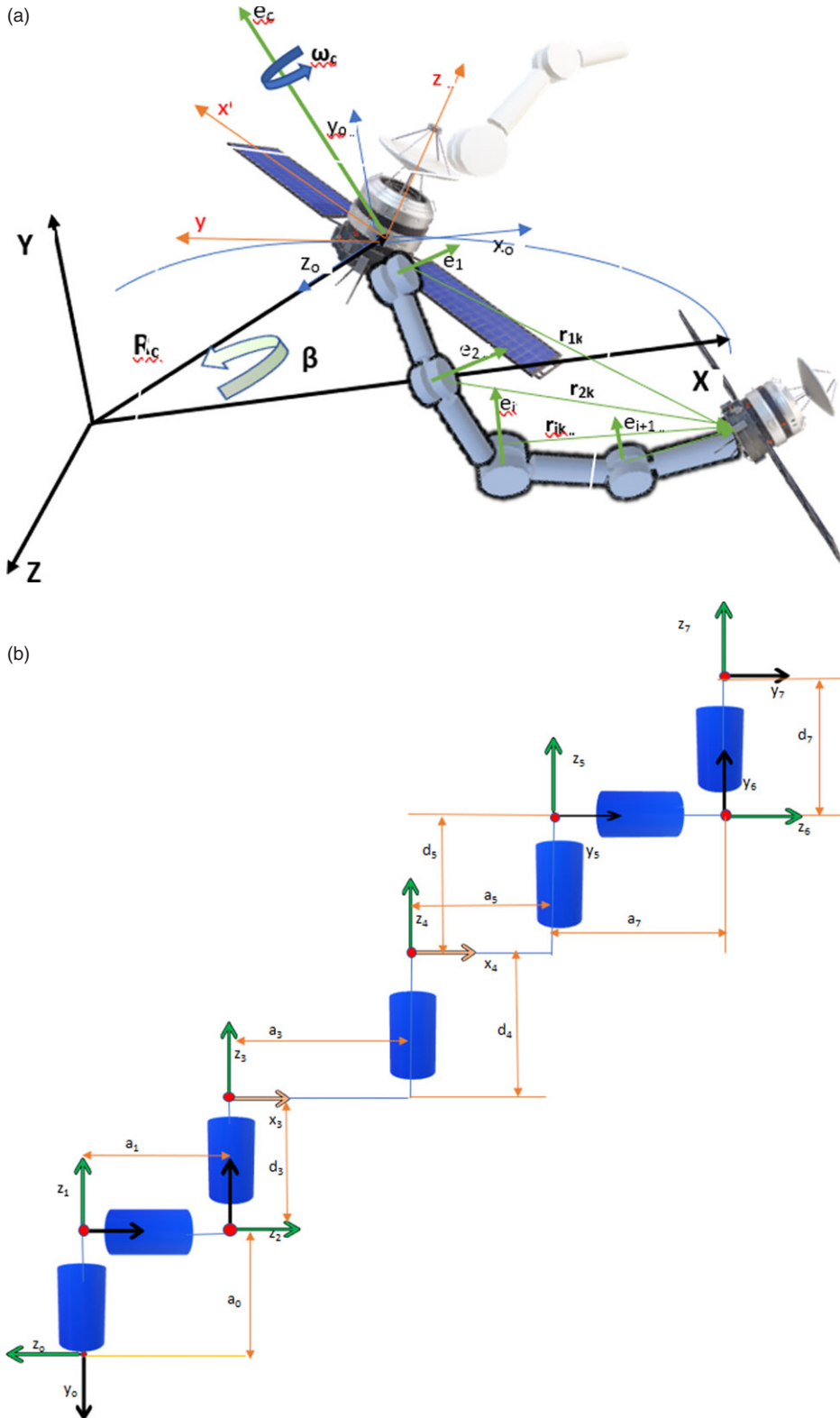


Figure 1. (a) The geometry and description of the system. (b) The robot arm geometry. (c) The 3-D robot arm model. (d) Free body diagram of link i .

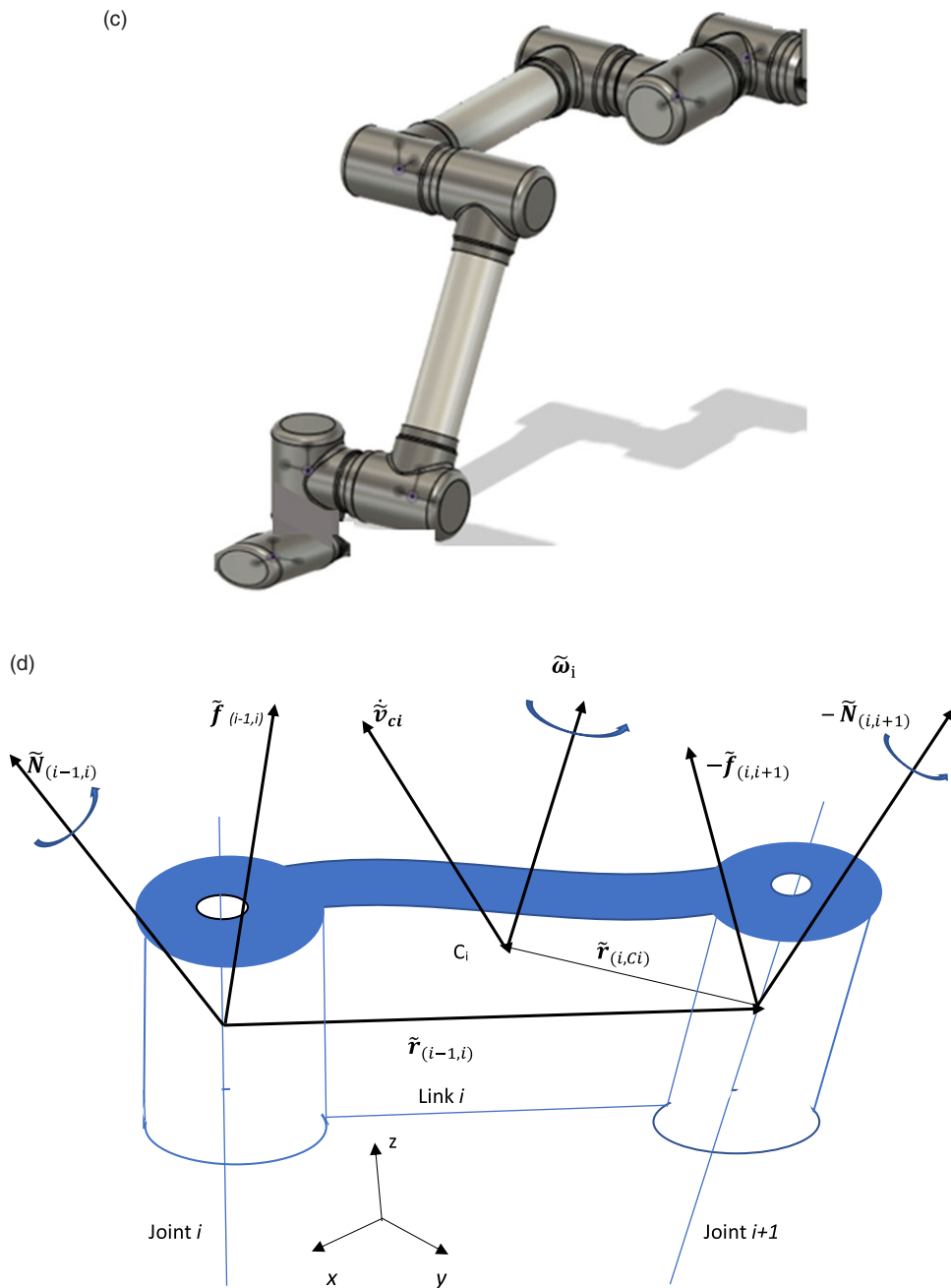


Figure 1. Continued

The transformation matrix $\underline{\mathbf{B}}$ which defines the rotational motion from the inertial coordinates frame X, Y, Z to the orbital coordinate frame (x_o, y_o, z_o) can be calculated as follows:

$$\underline{\mathbf{B}} = \begin{bmatrix} \sin \beta & 0 & \cos \beta \\ 0 & 1 & 0 \\ -\cos \beta & 0 & \sin \beta \end{bmatrix} \text{ where } \beta \text{ is the true anomaly.} \tag{3}$$

These transformation matrices are obtained for both the target and chaser spacecrafts. Similarly, each homogeneous transformation matrix T_i^j that transforms the coordinates of a point from frame j to frame i on a robot manipulator is calculated using Denavit-Hartenberg (D-H) convention [52, 54].

$$T_i^j = A_{i+1}A_{i+2}\dots A_{j-1}A_j \quad i < j \quad A_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\cos \theta_i \sin \alpha_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

where the quantities $d_i, a_i, \theta_i, \alpha_i$, are parameters of link i and joint- i and a_i , is the length of the link, d_i is the offset, θ_i is the joint angle while α_i is the twist.

The Jacobian matrices and their first-time derivatives between the center of the chaser spacecraft and a point k on the left robotics manipulator are as follows:

$$(J_c^k)_L = \begin{bmatrix} \tilde{e}_c & \tilde{e}_1 & \tilde{e}_2 & \tilde{e}_3 & \tilde{e}_4 & \dots & \tilde{e}_7 \\ \tilde{e}_c \times \tilde{r}_{ck} & \tilde{e}_1 \times \tilde{r}_{1k} & \tilde{e}_2 \times \tilde{r}_{2k} & \tilde{e}_3 \times \tilde{r}_{3k} & \tilde{e}_4 \times \tilde{r}_{4k} & \dots & \tilde{e}_7 \times \tilde{r}_{7k} \end{bmatrix} \quad (5)$$

$$\left(\dot{J}_c^k\right)_L = \begin{bmatrix} \tilde{e}_c & \tilde{e}_1 & \dots & \dots & \dots & \dots & \tilde{e}_7 \\ \tilde{e}_c \times (\tilde{\omega}_c \times \tilde{r}_{ck}) & \tilde{e}_1 \times (\dot{\theta}_1 \tilde{e}_1 \times \tilde{r}_{1k}) & \dots & \dots & \dots & \dots & \tilde{e}_7 \times (\dot{\theta}_7 \tilde{e}_7 \times \tilde{r}_{7k}) \end{bmatrix} \quad (6)$$

where \tilde{e}_c is the unit vector along $\tilde{\omega}_c$ spacecraft angular velocity vector, \tilde{e}_i is the unit vector along the rotation axis of the i th joint and $\tilde{r}_{ck}, \tilde{r}_{ik}$ are the displacement vectors from the center mass of the chaser spacecraft and i th joint to the point k , respectively. For example, if the Jacobian matrix between the center of the chaser spacecraft and the end-effector is selected, k will be assigned to $(n + 1)$. Similarly, the Jacobian matrices and time derivatives can be calculated for the right manipulator.

The linear, angular velocities, and accelerations of any point on the left manipulator can be calculated by using the Jacobian matrix [54]:

$$\begin{bmatrix} \tilde{\Omega}_k \\ \tilde{V}_k \end{bmatrix}^L = (J_c^k)_L \begin{bmatrix} \tilde{\omega}_c \\ \dot{\theta}_L \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \dot{R}_c \end{bmatrix} \quad (7)$$

$$\begin{bmatrix} \dot{\tilde{\Omega}}_k \\ \dot{\tilde{V}}_k \end{bmatrix}^L = (J_c^k)_L \begin{bmatrix} \dot{\tilde{\omega}}_c \\ \ddot{\theta}_L \end{bmatrix} + \left(\dot{J}_c^k\right)_L \begin{bmatrix} \tilde{\omega}_c \\ \dot{\theta}_L \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \ddot{R}_c \end{bmatrix} \quad (8)$$

where $\dot{\theta}_L^T = [\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3 \dots \dot{\theta}_n]$ is the left-arm joint rates, \tilde{V}_k and $\tilde{\Omega}_k$ are the (3×1) linear and angular velocity vectors of the k point on the left manipulator, respectively. Similarly, the linear and angular velocity of any point on the right manipulator can also be determined by simply changing the index from L to R .

2.3. Dynamics model of the system

The dynamics equations are obtained by Newton-Euler formulation. Figure 1(d) shows all the forces and moments acting on link i .

The translational equation of motion is developed by adding the inertial force to the static balance of forces on each link so that

$$\tilde{f}_{(i-1,i)} - \tilde{f}_{(i,i+1)} - m_i \dot{v}_{ci} = \tilde{\mathbf{0}} \quad i = 1, \dots, m \quad (9)$$

where $\tilde{f}(i-1, i)$ and $-\tilde{f}(i, i + 1)$ are the coupling forces applied to link i by links $i-1$ and $i + 1$, respectively.

The inertial torque acting on link i is given by the time rate of change of the angular momentum of the link at that instant. Let $\tilde{\omega}_i$ be the angular velocity vector and I_i be the centroidal inertia tensor of

link i , then the angular momentum is given by $\underline{I}_i \dot{\tilde{\omega}}_i$ and the gyroscopic torque is calculated as $\tilde{\omega}_i \times \underline{I}_i \dot{\tilde{\omega}}_i$. Adding these terms to the static balance of moments yields

$$\begin{aligned} & \tilde{N}_{(i-1,i)} - \tilde{N}_{(i,i+1)} - (\tilde{r}_{(i-1,i)} + \tilde{r}_{(i,Ci)}) \\ & \times \tilde{f}(i-1, i) + (-\tilde{r}_{(i,Ci)}) \times (-\tilde{f}(i, i+1)) - \underline{I}_i \dot{\tilde{\omega}}_i - \tilde{\omega}_i \times (\underline{I}_i \dot{\tilde{\omega}}_i) = \tilde{\mathbf{0}} \end{aligned} \tag{10}$$

where $\tilde{N}_{(i-1,i)}$ and $\tilde{N}_{(i,i+1)}$ are the coupling moments and $\tilde{r}_{(i-1,i)}$, $\tilde{r}_{(i,Ci)}$ are the displacement vectors as shown in Fig. 1(d).

The complete set of equations for the two robots and the chaser spacecraft is obtained by evaluating both equations for all the links and the spacecraft, $i = 1, \dots, m$. To obtain closed-form dynamic equations, one can first eliminate the constraint forces and separate them from the joint torques, so as to explicitly involve the joint torques in the dynamic equations as follows [47]:

$$\begin{bmatrix} \underline{G}_v & \underline{G}_{cw} & \underline{G}_{c\theta_L} & \underline{G}_{c\theta_R} \\ \underline{G}_{cw}^T & \underline{G}_w & \underline{G}_{w\theta_L} & \underline{G}_{w\theta_R} \\ \underline{G}_{c\theta_L}^T & \underline{G}_{w\theta_L}^T & \underline{G}_{\theta_L} & \underline{G}_{\theta_{LR}} \\ \underline{G}_{c\theta_R}^T & \underline{G}_{w\theta_R}^T & \underline{G}_{\theta_{LR}}^T & \underline{G}_{\theta_R} \end{bmatrix} \begin{bmatrix} \ddot{\tilde{R}}_c \\ \dot{\tilde{\omega}}_c \\ \ddot{\tilde{\theta}}_L \\ \ddot{\tilde{\theta}}_R \end{bmatrix} + \begin{bmatrix} \tilde{c}_v \\ \tilde{c}_w \\ \tilde{c}_{\theta_L} \\ \tilde{c}_{\theta_R} \end{bmatrix} = \begin{bmatrix} \tilde{F}_c \\ \tilde{\tau}_c \\ \tilde{\tau}_{\theta_L} \\ \tilde{\tau}_{\theta_R} \end{bmatrix} \tag{11}$$

where \underline{G} the mass/inertia matrix is a positive definite matrix and, $\dot{\tilde{R}}_c$, $\dot{\tilde{\omega}}_c$ are the linear and angular acceleration of the chaser spacecraft, $\ddot{\tilde{\theta}}_L$, $\ddot{\tilde{\theta}}_R$ are the joint angular accelerations for the left and right arm, respectively. \tilde{c}_v , \tilde{c}_w , \tilde{c}_{θ_L} , and \tilde{c}_{θ_R} are the non-linear terms while \tilde{F}_c , $\tilde{\tau}_c$ are the external control force and moments for the chaser spacecraft and $\tilde{\tau}_{\theta_L}$, $\tilde{\tau}_{\theta_R}$ are the joint control torques for the left and right arm, respectively.

2.4. Proposed optimal control techniques for dynamics control of coupled spacecraft attitude and robotics manipulators

The system consisting of two redundant manipulators and the two spacecrafts, namely the chaser and the target, is mathematically an underdetermined system due to the redundant number of actuators and sensors available to control the attitude and translational motions of the links and the spacecrafts.

This section presents a novel control solution to this problem by two control algorithms based on (i) Optimal Control Allocation OCA and (ii) Non-linear Model Predictive Control (NMPC). Both algorithms are aimed to minimize joint torques, spacecraft actuator moments, and contact forces moments of a compound redundant system which consists of a common payload (target spacecraft) grasped by dual n -degree space robotics manipulators mounted on a chaser spacecraft while tracking a desired trajectory. The OCA is based on optimal control allocation minimizing a quadratic cost function using the current state values and the system dynamics. On the other hand, the second NMPC algorithm minimizes a quadratic cost function which not only includes the current states but also the future state estimates, and the control inputs over a specified prediction horizon. The NMPC is deemed to be a natural extension of the OCA algorithm presented here and either technique can be employed based on availability of on-board computational resources.

2.4.1. The OCA formulation

The OCA algorithm involves two stages as demonstrated below in a control system block diagram in Fig. 2(a). The first stage of the control block diagram includes impedance control technique which is responsible for generating the desired trajectories for the end-effectors of the left and right robots prior to making a contact with the payload [54].

The second stage of the block diagram based on OCA algorithm is robust and computationally very efficient and provides minimum norm of joint torques and contact forces for an underdetermined

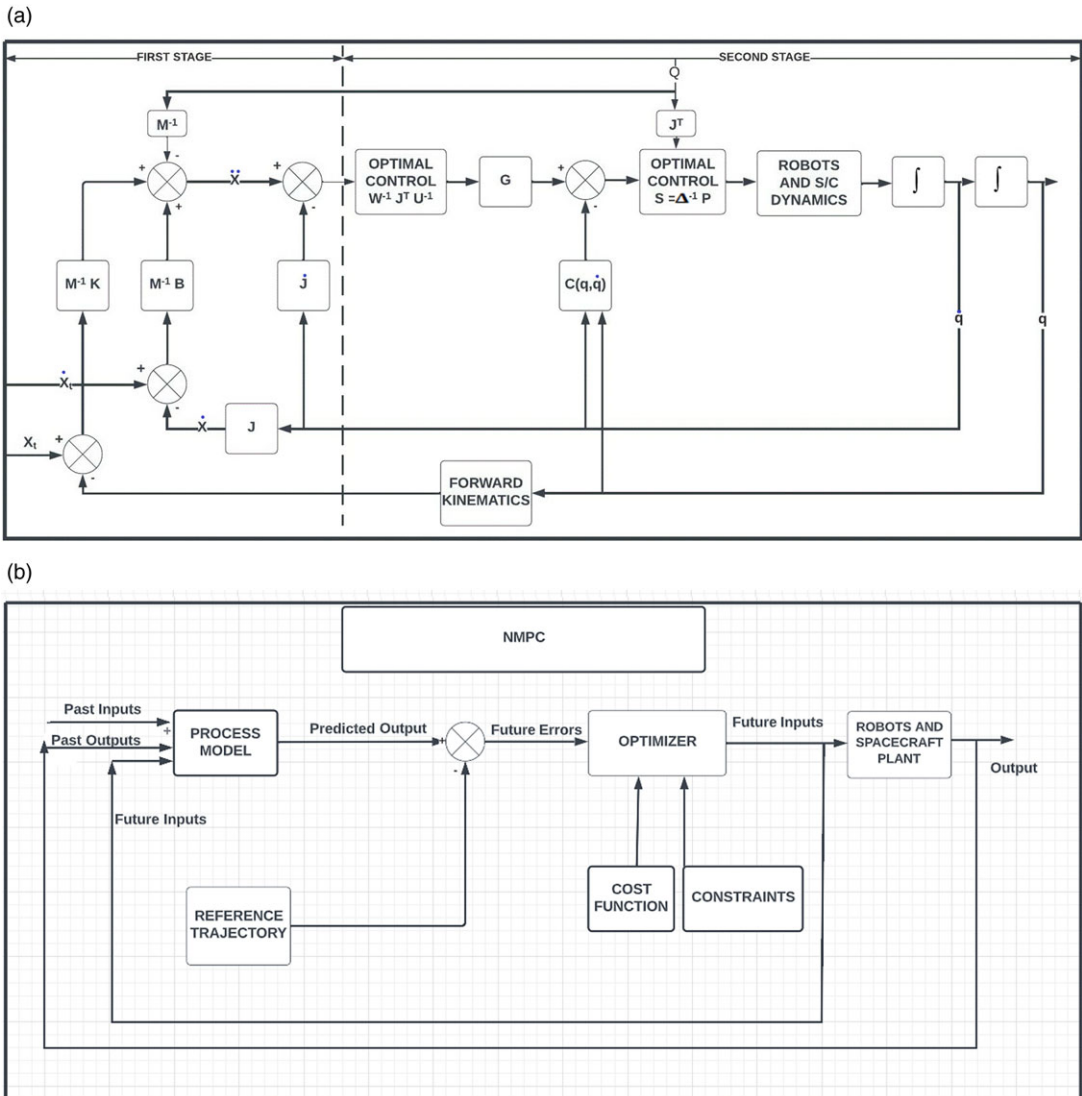


Figure 2. (a) Two-stage control system block diagram with OCA. (b) Non-linear model predictive control block (NMPC) diagram.

compound robotics system. This technique was originally developed for cooperating rovers [51]. The formulation is hereby extended to free-floating spacecraft-based cooperating space arms. In addition, it is further applied to two different modes of operations (i.e., pre-contact and post-capture) of space arms as presented below.

2.4.1.1. OCA pre-contact stage

This is the mode of operation where the two end-effectors approach toward the target spacecraft (i.e., payload). In a real space application, the relative motion of the target is determined via a vision or a laser system located on the robot end-effectors or on the capture spacecraft and thus the end-effectors approach to the vicinity of the payload's grapple fixtures to grab the payload.

One can assume that the relative payload-grapple fixture trajectory, \tilde{X}_{i-rel} , is determined via the chaser spacecraft vision system. After transforming this information into the inertial reference frame, the following relationships can be written between the time derivatives of the pose of the left grapple fixture and the joint rates and angular rates of the chaser spacecraft using the Jacobian matrix and its time derivative.

$$\dot{\tilde{X}}_L = \begin{bmatrix} \tilde{\Omega}_k \\ \tilde{V}_k \end{bmatrix}^L = (J_c^k)_L \begin{bmatrix} \tilde{\omega}_c \\ \dot{\tilde{\theta}}_L \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \dot{\tilde{R}}_c \end{bmatrix} \tag{12a}$$

$$\dot{\tilde{X}}_{Lrel} = \dot{\tilde{X}}_L - \begin{bmatrix} \mathbf{0} \\ \dot{\tilde{R}}_c \end{bmatrix} \tag{12b}$$

$$\ddot{\tilde{X}}_L = \begin{bmatrix} \dot{\tilde{\Omega}}_k \\ \dot{\tilde{V}}_k \end{bmatrix}^L = (J_c^k)_L \begin{bmatrix} \dot{\tilde{\omega}}_c \\ \ddot{\tilde{\theta}}_L \end{bmatrix} + (\dot{J}_c^k)_L \begin{bmatrix} \tilde{\omega}_c \\ \dot{\tilde{\theta}}_L \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \ddot{\tilde{R}}_c \end{bmatrix} \tag{13a}$$

$$\ddot{\tilde{X}}_{Lrel} = \ddot{\tilde{X}}_L - \begin{bmatrix} \mathbf{0} \\ \ddot{\tilde{R}}_c \end{bmatrix} \tag{13b}$$

where k refers to the location of the left grapple fixture, \tilde{X}_L and \tilde{X}_{Lrel} are the absolute and the relative displacement vectors, respectively. Similar relations exist for the right grapple fixture.

During the pre-contact stage, the end-effectors are in a free motion (i.e., holding no common payload) and the system consists of a tree structure. The following cost function, C_o is employed to obtain the minimum norm joint and Euler angular rates:

$$C_o = \frac{1}{2} \tilde{S}_o^T \underline{W} \tilde{S}_o + \sum_i \tilde{\lambda}_i^T \left\{ (J_c^k)_i \dot{\tilde{q}}_i - \dot{\tilde{X}}_{irel} \right\} \tag{14}$$

\underline{W} is a square $(3 + 2n, 3 + 2n)$ positive definite weighting matrix, n is being the total number of joints for each arm, and $\tilde{\lambda}_i^T$ is the (6×1) Lagrangian multiplier and i is replaced by the L and R for the left and right robots, respectively.

The $\dot{\tilde{q}}$ and \tilde{S}_o vectors which consist of the spacecraft angular as well as the left and right arm joint angular rates are provided below:

$$\dot{\tilde{q}}_i = \begin{bmatrix} \tilde{\omega}_c \\ \dot{\tilde{\theta}}_i \end{bmatrix} \text{ and } \tilde{S}_o = \begin{bmatrix} \tilde{\omega}_c \\ \dot{\tilde{\theta}}_L \\ \dot{\tilde{\theta}}_R \end{bmatrix} \tag{15}$$

Minimizing the cost function C_o with respect to $\dot{\tilde{q}}_i$ and $\tilde{\lambda}_i$ can be carried out by differentiating C_o with respect to $\dot{\tilde{q}}_i$ and $\tilde{\lambda}_i$ (where i is replaced by L and R for the left and right robots, respectively), and one can calculate the minimum norm of joint and Euler rates as follows:

$$\frac{\partial C_o}{\partial \dot{\tilde{q}}_i} = \left\{ \underline{W}_i \dot{\tilde{q}}_i + (J_c^k)_i^T \tilde{\lambda}_i \right\} = \tilde{\mathbf{0}} \tag{16}$$

$$\frac{\partial C_o}{\partial \tilde{\lambda}_i} = (J_c^k)_i \dot{\tilde{q}}_i - \dot{\tilde{X}}_i = \tilde{\mathbf{0}} \tag{17}$$

Solving for $\dot{\tilde{q}}_i$ from Eq. (16), one can obtain the following relationship:

$$\dot{\tilde{q}}_i = -\underline{W}_i^{-1} (\underline{J}_c^k)_i^T \tilde{\lambda}_i \tag{18}$$

Substituting this relationship into Eq. (17), and defining a new variable matrix, \underline{U}_i

$$\underline{U}_i = (\underline{J}_c^k)_i \underline{W}_i^{-1} (\underline{J}_c^k)_i^T \tag{19}$$

$$\tilde{\lambda}_i^{\text{opt}} = -\underline{U}_i^{-1} \dot{\tilde{X}}_i \tag{20}$$

In addition, substituting $\tilde{\lambda}_i^{\text{opt}}$ into Eq. (18), one can solve for the minimum norm $\dot{\tilde{q}}_i$ and $\ddot{\tilde{q}}_{i,\text{min}}$ as follows:

$$\dot{\tilde{q}}_{i,\text{min}} = \underline{W}_i^{-1} (\underline{J}_c^k)_i^T \underline{U}_i^{-1} \dot{\tilde{X}}_i \tag{21}$$

$$\ddot{\tilde{q}}_{i,\text{min}} = \underline{W}_i^{-1} (\underline{J}_c^k)_i^T \underline{U}_i^{-1} \left\{ \ddot{\tilde{X}}_i - (\underline{J}_c^k)_i \dot{\tilde{q}}_{i,\text{min}} \right\} \tag{22}$$

Furthermore, referring to the first part of the control block diagram, the impedance control scheme can be employed to generate trajectories for the both end-effectors to track the target payload as described below:

$$\begin{aligned} \ddot{\tilde{X}}_i &= \underline{M}_i^{-1} \underline{B} \left\{ \dot{\tilde{X}}_{di} - \dot{\tilde{X}}_i \right\} + \underline{M}_i^{-1} \underline{K} \left\{ \tilde{X}_{di} - \tilde{X}_i \right\} \\ i &= L, R \end{aligned} \tag{23}$$

where \underline{M}_i , \underline{B} , \underline{K} are (6×6) positive definite matrices and are selected to satisfy the tracking performance requirements while \tilde{X}_{di} and \tilde{X}_i are the desired payload and the end-effector trajectory respectively [54]. The matrix selection process will not be discussed here for the sake of brevity.

Through the application of the inverse dynamics of the system as shown in the control block diagram, the joint torques, and the chaser spacecraft moments then can be calculated accordingly as shown below:

$$\begin{bmatrix} \tilde{F}_c \\ \tilde{\tau}_c \\ \tilde{\tau}_{\theta_L} \\ \tilde{\tau}_{\theta_R} \end{bmatrix} = \begin{bmatrix} \underline{G}_v & \underline{G}_{c\omega} & \underline{G}_{c\theta_L} & \underline{G}_{c\theta_R} \\ \underline{G}_{cv}^T & \underline{G}_w & \underline{G}_{w\theta_L} & \underline{G}_{w\theta_R} \\ \underline{G}_{c\theta_L}^T & \underline{G}_{w\theta_L}^T & \underline{G}_{\theta_L} & \underline{G}_{\theta_{LR}} \\ \underline{G}_{c\theta_R}^T & \underline{G}_{w\theta_R}^T & \underline{G}_{\theta_{LR}}^T & \underline{G}_{\theta_R} \end{bmatrix} \begin{bmatrix} \ddot{\tilde{R}}_c \\ \dot{\tilde{\omega}}_c \\ \ddot{\tilde{\theta}}_L \\ \ddot{\tilde{\theta}}_R \end{bmatrix} + \begin{bmatrix} \tilde{c}_v \\ \tilde{c}_w \\ \tilde{c}_{\theta_L} \\ \tilde{c}_{\theta_R} \end{bmatrix} \tag{24}$$

2.4.1.2 OCA post-capture stage

Saturation of torques of space robot joint actuators and momentum wheels mounted on a spacecraft is a major concern in real applications. In this mode of operation, a cost function C_p is defined to minimize the joint torques $\tilde{\tau}_{\theta_L}$, $\tilde{\tau}_{\theta_R}$ and the chaser spacecraft thrust force and moments \tilde{F}_c , $\tilde{\tau}_c$ and the contact forces and moments \tilde{F}_i and \tilde{N}_i applied by the robot manipulators on the target spacecraft (i.e., payload)

Therefore, a cost function, C_p can be designed as:

$$C_p = \frac{1}{2} \tilde{S}^T \underline{W} \tilde{S} + \tilde{\lambda}^T \tilde{E} \tag{25a}$$

\underline{W} is a square (2n + 18, 2n + 18) positive definite weighting matrix provided in Eq. (25b) while $\tilde{\lambda}$ is the ((2n + 12), 1) Lagrangian multiplier and \tilde{E} is a vector which consists of the constraints equations as defined by Eq. (28).

$$\underline{W} = \begin{pmatrix} \underline{W}_{N_L} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \underline{W}_{F_L} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \underline{W}_{N_R} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \underline{W}_{F_R} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \underline{W}_{F_c} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \underline{W}_{\tau_c} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \underline{W}_{\tau_L} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \underline{W}_{\tau_R} \end{pmatrix} \tag{25b}$$

The \tilde{S} vector consists of the end-effector contact forces and moments as well as the chaser spacecraft control force and moments, the joint torques for the left and right arm, respectively, as given below:

$$\text{Let } \tilde{Q}_i = \begin{bmatrix} \tilde{N}_i \\ \tilde{F}_i \end{bmatrix}, \text{ then } \tilde{S}^T = [\tilde{Q}_L, \tilde{Q}_R, \tilde{F}_c, \tilde{\tau}_c, \tilde{\tau}_{\theta_L}, \tilde{\tau}_{\theta_R}] \tag{26}$$

where \tilde{F}_i and \tilde{N}_i are the contact forces and moments applied by the left and right robot manipulators on the target spacecraft, respectively, while \tilde{F}_c and $\tilde{\tau}_c$ are the chaser spacecraft control thrust forces and moments.

The \tilde{E} vector can be calculated as follows:

$$\tilde{E} = \begin{bmatrix} \tilde{F}_L + \tilde{F}_R - m_t \ddot{x}_t \\ \tilde{N}_L + \tilde{N}_R - \tilde{d}_L \times \tilde{F}_L - \tilde{d}_R \times \tilde{F}_R - [I_t \Omega_t + \Omega_t \times I_t \Omega_t] \\ \begin{bmatrix} \tilde{F}_c \\ \tilde{\tau}_c \\ \tilde{\tau}_{\theta_L} \\ \tilde{\tau}_{\theta_R} \end{bmatrix} - \begin{bmatrix} G_v & G_{c\omega} & G_{c\theta_L} & G_{c\theta_R} \\ G_{cw}^T & G_w & G_{w\theta_L} & G_{w\theta_R} \\ G_{c\theta_L}^T & G_{w\theta_L}^T & G_{\theta_L} & G_{\theta_{LR}} \\ G_{c\theta_R}^T & G_{w\theta_R}^T & G_{\theta_{LR}}^T & G_{\theta_R} \end{bmatrix} \begin{bmatrix} \ddot{R}_c \\ \dot{\omega}_c \\ \ddot{\theta}_L \\ \ddot{\theta}_R \end{bmatrix} - \begin{bmatrix} \tilde{c}_v \\ \tilde{c}_w \\ \tilde{c}_{\theta_L} \\ \tilde{c}_{\theta_R} \end{bmatrix} \end{bmatrix} \tag{27}$$

where \tilde{d}_L, \tilde{d}_R are the displacement vectors from the left and right end-effectors to the center of mass of the target spacecraft, respectively. Also, $\tilde{d}_L \times \tilde{F}_L$ can be re-written as a matrix multiplication as $\underline{D}_L \tilde{F}_L$, where $\underline{d}_L = (x_L, y_L, z_L)T$ and then

$$\underline{D}_L = \begin{bmatrix} 0 & -z_L & y_L \\ z_L & 0 & -x_L \\ -y_L & x_L & 0 \end{bmatrix} \tag{28}$$

and L can be replaced by R for the right robot.

The cost function C_p can be minimized by differentiating C_p with respect to $\tilde{\lambda}_i$ and \tilde{S} to calculate the minimum norm of joint torques and end-effectors force and moments exerted on the target spacecraft.

$$\frac{\partial C_p}{\partial \tilde{S}} = \tilde{\mathbf{0}} \tag{29}$$

and

$$\frac{\partial C_p}{\partial \tilde{\lambda}} = \tilde{\mathbf{0}} \tag{30}$$

Carrying out some algebraic manipulations, one can eliminate $\tilde{\lambda}$ from the equations and obtain the minimum norm of \tilde{S} vector as provided in Eq. (31).

$$\tilde{S} = \underline{\Delta}^{-1} \tilde{P} \tag{31}$$

where $\underline{\Delta}$ is a $((2n + 18) \times (2n + 18))$ square matrix. $\underline{\Delta}$ and $\tilde{\mathbf{P}}$ are given below:

$$\underline{\Delta} = \begin{bmatrix} \underline{W}_{N_L} & \underline{\mathbf{0}} & -\underline{W}_{N_R} & \underline{\mathbf{0}} & -\underline{W}_{\tau_L} (\underline{J}_c^k)^T & \underline{W}_{\tau_L} (\underline{J}_c^k)^T \\ (\underline{D}_L - \underline{D}_R) \underline{W}_{N_L} & \underline{W}_{F_L} & \underline{\mathbf{0}} & -\underline{W}_{F_R} & -(\underline{\mathbf{1}} + \underline{D}_L - \underline{D}_R) (\underline{J}_c^k)^T & \underline{W}_{\tau_R} (\underline{J}_c^k)^T \\ \underline{\mathbf{0}} & \underline{\mathbf{1}} & \underline{\mathbf{0}} & \underline{\mathbf{1}} & \underline{\mathbf{0}} & \underline{\mathbf{0}} \\ \underline{\mathbf{1}} & -\underline{D}_L^T & \underline{\mathbf{1}} & \underline{D}_R & \underline{\mathbf{0}} & \underline{\mathbf{0}} \\ (\underline{J}_c^k)^T_{L1} & (\underline{J}_c^k)^T_{L2} & \underline{\mathbf{0}} & \underline{\mathbf{0}} & \underline{\mathbf{1}} & \underline{\mathbf{0}} \\ \underline{\mathbf{0}} & \underline{\mathbf{0}} & (\underline{J}_c^k)^T_{R1} & (\underline{J}_c^k)^T_{R2} & \underline{\mathbf{0}} & \underline{\mathbf{1}} \end{bmatrix} \quad (32)$$

$$\tilde{\mathbf{P}} = \begin{bmatrix} \tilde{\mathbf{0}} \\ \tilde{\mathbf{0}} \\ \tilde{\mathbf{0}} \\ \tilde{\mathbf{0}} \\ m_t \ddot{\tilde{\mathbf{x}}}_t \\ \left[\underline{I}_t \dot{\tilde{\Omega}}_t + \tilde{\Omega}_t \times \underline{I}_t \tilde{\Omega}_t \right] \\ \begin{bmatrix} \underline{G}_v & \underline{G}_{c\omega} & \underline{G}_{c\theta_L} & \underline{G}_{c\theta_R} \\ \underline{G}_{cw}^T & \underline{G}_w & \underline{G}_{w\theta_L} & \underline{G}_{w\theta_R} \\ \underline{G}_{c\theta_L}^T & \underline{G}_{w\theta_L}^T & \underline{G}_{\theta_L} & \underline{G}_{\theta_{LR}} \\ \underline{G}_{c\theta_R}^T & \underline{G}_{w\theta_R}^T & \underline{G}_{\theta_{LR}}^T & \underline{G}_{\theta_R} \end{bmatrix} \begin{bmatrix} \ddot{\tilde{\mathbf{R}}}_c \\ \dot{\tilde{\omega}}_c \\ \ddot{\tilde{\theta}}_L \\ \ddot{\tilde{\theta}}_R \end{bmatrix} + \begin{bmatrix} \tilde{c}_v \\ \tilde{c}_w \\ \tilde{c}_{\theta_L} \\ \tilde{c}_{\theta_R} \end{bmatrix} \end{bmatrix} \quad (33)$$

where $\dot{\tilde{\mathbf{x}}}_t$ and $\tilde{\Omega}_t$ are the linear and rotational velocity of the center of mass of the payload,

$$\dot{\tilde{\mathbf{X}}}_{dt} = \begin{bmatrix} \tilde{\Omega}_t \\ \dot{\tilde{\mathbf{x}}}_t \end{bmatrix} \quad (34)$$

where \underline{W}_{Ni} and $\underline{W}_{\tau i}$ are the sub-weighting matrices, and they are the components of the \underline{W} square $(2n + 18, 2n + 18)$ weighting matrix while $(\underline{J}_c^k)^T_{L1}$ and $(\underline{J}_c^k)^T_{R1}$ are $(n \times 3)$ partial Jacobian matrices defined as below:

$$(\underline{J}_c^k)^T_i = \left[(\underline{J}_c^k)^T_{i1}, (\underline{J}_c^k)^T_{i2} \right] \quad (35)$$

2.4.2 The NMPC formulation

The NMPC block diagram is provided in Fig. 2(b). The NMPC block diagram simply replaces the second stage of Fig. 2(a). The reference trajectory (shown in Fig. 2(b)) is the output of the first stage, that is, the impedance control trajectory generation in Fig. 2(a). However, the future state estimates are also incorporated in the calculations to estimate the future reference trajectory values over the prediction horizon. In this case, a performance index (i.e., a cost function) of the system is optimized by implementing a robust NMPC that takes into account the predictions of the output signal and the constraints on the states, inputs, and outputs as shown in Fig. 2(b).

The main difference between the NMPC and the OCA is that the former uses a system model to predict and control the future behavior while the latter only considers the current and the past behaviors.

The NMPC algorithm is used to predict and control the system's future behavior by implementing a set of optimization problems at each control interval. These problems are usually related to the constraints and cost functions. The cost function is a scalar that must be minimized at each interval to measure the performance of the system. In addition to the cost functions, the system can also be subjected to physical constraints on the states and plant output to check its performance. The modified states are adjusted according to the constraints applied to the system.

The conventional MPC formulation for the dual arm non-linear system can be written as:

$$\min \int_0^{T_p} [(\tilde{\mathbf{y}}(t) - \tilde{\mathbf{y}}_r(t))^T \mathbf{K} (\tilde{\mathbf{y}}(t) - \tilde{\mathbf{y}}_r(t)) + \tilde{\mathbf{S}}^T(t) \mathbf{W} \tilde{\mathbf{S}}(t)] dt$$

subject to:

$$\begin{aligned} \dot{\tilde{\mathbf{y}}} &= \tilde{\mathbf{g}}(\tilde{\mathbf{y}}) + \mathbf{L}\tilde{\mathbf{S}} \\ \tilde{\mathbf{z}} &= \tilde{\mathbf{g}}_z(\tilde{\mathbf{y}}) + \mathbf{H}\tilde{\mathbf{S}} \\ \tilde{\mathbf{y}}(0) &= \tilde{\mathbf{y}}(t_0) \\ \tilde{\mathbf{S}}_{\min} &< \mathbf{A}\tilde{\mathbf{S}} < \tilde{\mathbf{S}}_{\max} \end{aligned} \tag{36}$$

where T_p is the prediction horizon; \mathbf{K} and \mathbf{W} are $(2(6 + 2n) \times 2(6 + 2n))$ and $((18 + 2n) \times (18 + 2n))$ positive definite square weighting matrices, respectively; $\tilde{\mathbf{g}}(\tilde{\mathbf{y}})$, $\tilde{\mathbf{g}}_z(\tilde{\mathbf{y}})$, \mathbf{L} , \mathbf{H} are part of the non-linear system equations and \mathbf{A} is a square positive definite scaling matrix for saturation limits. $\tilde{\mathbf{y}}(t)^T = [\tilde{\mathbf{R}}_c, \tilde{\boldsymbol{\theta}}_L, \tilde{\boldsymbol{\theta}}_R, \dot{\tilde{\mathbf{R}}}_c, \dot{\tilde{\phi}}, \dot{\tilde{\psi}}, \dot{\tilde{\theta}}, \dot{\tilde{\boldsymbol{\theta}}}_L, \dot{\tilde{\boldsymbol{\theta}}}_R]$ is a $(I \times 2(6 + 2n))$ state vector including the chaser spacecraft position, Euler angles, and the joint angles for the left and the right arms and their rates; $\tilde{\mathbf{y}}_r(t)$ is the reference/desired states, the $\tilde{\mathbf{S}}$ vector consists of the contact forces and moments as well as the chaser spacecraft control force and moments, the joint torques for the left and right arm, respectively, as previously defined in Eq. (26), $\tilde{\mathbf{S}}^T = [\tilde{\mathbf{Q}}_L, \tilde{\mathbf{Q}}_R, \tilde{\mathbf{F}}_c, \tilde{\boldsymbol{\tau}}_c, \tilde{\boldsymbol{\tau}}_{\theta_L}, \tilde{\boldsymbol{\tau}}_{\theta_R}]$ and is constrained between $\tilde{\mathbf{S}}_{\min}$ and $\tilde{\mathbf{S}}_{\max}$.

The non-linear system can now be described with an exact quasi-linear parameter varying realization [43].

$$\begin{aligned} \tilde{\mathbf{y}}(\hat{k} + 1) &= \hat{\mathbf{A}}(\hat{\mathbf{g}}(\hat{k})) \tilde{\mathbf{y}}(\hat{k}) + \hat{\mathbf{B}}(\hat{\mathbf{g}}(\hat{k})) \tilde{\mathbf{S}}(\hat{k}) \\ \tilde{\mathbf{z}}(\hat{k}) &= \hat{\mathbf{C}}(\hat{\mathbf{g}}(\hat{k})) \tilde{\mathbf{y}}(\hat{k}) + \hat{\mathbf{D}}(\hat{\mathbf{g}}(\hat{k})) \tilde{\mathbf{S}}(\hat{k}) \\ \hat{\mathbf{g}}(\hat{k}) &= \mathbf{f}_g(\tilde{\mathbf{y}}(\hat{k})) \tilde{\mathbf{S}}_{\min} < \mathbf{A}\tilde{\mathbf{S}}(\hat{k}) < \tilde{\mathbf{S}}_{\max} \end{aligned} \tag{37}$$

where \hat{k} is the sampling instant and $\tilde{\mathbf{z}}(\hat{k})$ is a vector of the measured outputs at instant \hat{k} .

The NMPC is applied at each sampling instant, and the discrete states $\tilde{\mathbf{y}}(\hat{k})$ and control inputs $\tilde{\mathbf{S}}(\hat{k})$ are obtained minimizing the following performance index, that is, the Cost Function:

$$\begin{aligned} C_n &= \frac{1}{2} \sum_{j=1}^{N_p} \left[(\tilde{\mathbf{y}}(\hat{k} + j) - \tilde{\mathbf{y}}_r(\hat{k} + j))^T \mathbf{K} (\tilde{\mathbf{y}}(\hat{k} + j) - \tilde{\mathbf{y}}_r(\hat{k} + j)) \right. \\ &\quad \left. + \tilde{\mathbf{S}}(\hat{k} + j - 1)^T \mathbf{W} \tilde{\mathbf{S}}(\hat{k} + j - 1) \right] \\ \text{subject to } \tilde{\mathbf{y}}(\hat{k} + j + 1) &= \hat{\mathbf{A}}(\hat{\mathbf{g}}(\hat{k} + j)) \tilde{\mathbf{y}}(\hat{k} + j) + \hat{\mathbf{B}}(\hat{\mathbf{g}}(\hat{k} + j)) \tilde{\mathbf{S}}(\hat{k} + j) \\ \tilde{\mathbf{z}}(\hat{k} + j) &= \hat{\mathbf{C}}(\hat{\mathbf{g}}(\hat{k} + j)) \tilde{\mathbf{y}}(\hat{k} + j) + \hat{\mathbf{D}}(\hat{\mathbf{g}}(\hat{k} + j)) \tilde{\mathbf{S}}(\hat{k} + j) \\ \tilde{\mathbf{S}}_{\min} &\leq \mathbf{A}\tilde{\mathbf{S}}(\hat{k} + j) \leq \tilde{\mathbf{S}}_{\max} \end{aligned} \tag{38}$$

The cost function C_n can be further simplified with the substitution of $\tilde{\mathbf{y}}(\hat{k} + j)$ as follows:

$$\begin{aligned} C_n &= \frac{1}{2} \sum_{j=1}^{N_p} \left[\tilde{\mathbf{S}}(\hat{k} + j)^T \hat{\mathbf{H}} \tilde{\mathbf{S}}(\hat{k} + j) + \mathbf{2}(\tilde{\mathbf{f}})^T \tilde{\mathbf{S}}(\hat{k} + j) \right] \\ \hat{\mathbf{H}} &= (\hat{\mathbf{B}}^T \mathbf{K} \hat{\mathbf{B}} + \mathbf{W}) \\ (\tilde{\mathbf{f}})^T &= (\tilde{\mathbf{y}}(\hat{k} + j))^T \hat{\mathbf{A}}^T \mathbf{K} \hat{\mathbf{B}} - \tilde{\mathbf{y}}_r(\hat{k} + j)^T \mathbf{K} \hat{\mathbf{B}} \end{aligned} \tag{39}$$

The cost function C_n can be minimized by differentiating C_n with respect to $\tilde{\mathbf{S}}(\acute{k} + j)$ to calculate the minimum norm of vector $\tilde{\mathbf{S}}(\acute{k} + j)$ which consists of joint torques and end-effectors force and moments exerted on the target spacecraft.

$$\begin{aligned} \frac{\partial C_n}{\partial \tilde{\mathbf{S}}(\acute{k} + j)} &= \tilde{\mathbf{0}} \\ \tilde{\mathbf{S}}(\acute{k} + j) &= -\hat{\mathbf{H}}^{-1} \tilde{\mathbf{f}} \\ \tilde{\mathbf{S}}(\acute{k} + j) &= -(\hat{\mathbf{B}}^T \mathbf{K} \hat{\mathbf{B}} + \mathbf{W})^{-1} (\tilde{\mathbf{y}}(\acute{k} + j))^T \hat{\mathbf{A}}^T \mathbf{K} \hat{\mathbf{B}} - \tilde{\mathbf{y}}_r(\acute{k} + j)^T \mathbf{K} \hat{\mathbf{B}} \quad (40) \end{aligned}$$

subject to $\tilde{\mathbf{y}}(\acute{k} + j + 1) = \hat{\mathbf{A}}(\hat{\mathbf{g}}(\acute{k} + j)) \tilde{\mathbf{y}}(\acute{k} + j) + \hat{\mathbf{B}}(\hat{\mathbf{g}}(\acute{k} + j)) \tilde{\mathbf{S}}(\acute{k} + j)$

$$\tilde{\mathbf{z}}(\acute{k} + j) = \hat{\mathbf{C}}(\hat{\mathbf{g}}(\acute{k} + j)) \tilde{\mathbf{y}}(\acute{k} + j) + \hat{\mathbf{D}}(\hat{\mathbf{g}}(\acute{k} + j)) \tilde{\mathbf{S}}(\acute{k} + j)$$

$$\tilde{\mathbf{S}}_{\min} \leq \mathbf{\Lambda} \tilde{\mathbf{S}}(\acute{k} + j) \leq \tilde{\mathbf{S}}_{\max}$$

3. Results and discussion

The computer simulation results are provided and discussed in this section. There are two sets of simulation results associated with the two modes of operation, namely pre-contact and post-capture. As noted in the literature, the challenges associated with the joint torque saturation are related to the post-capture mode; therefore, comparative simulation results for the OCA and NMPC algorithms are presented, compared, and discussed during the post-capture stage.

3.1. Pre-contact mode of operation

In this simulation, first a trajectory for the center of mass of a fictitious target spacecraft is prescribed starting from an initial condition $\tilde{\mathbf{x}}_i^T(t_o) = [1.5, 0, 0, 0, 0, 0]$ to the final condition $\tilde{\mathbf{x}}_i^T(t_f) = [2, 0.5, 0, 0, 0, \pi/12]$ where $t_o = 0$ and $t_f = 2.5$ s, the initial and the final time values, respectively. $\tilde{\mathbf{x}}_i^T$ consists of three position and three orientation angles measured in meters and radians, respectively. It is assumed that the target spacecraft is equipped with two grapple fixtures located opposite sides of the target spacecraft, and the end-effectors will track these fixtures during the pre-contact phase and eventually lock into them during the capture phase.

The trajectories $\tilde{\mathbf{X}}_{di}$ for these grapple fixtures are then calculated using the rigid body motion of the target spacecraft center of mass $\tilde{\mathbf{x}}_i^T$ and the known geometry of these locations from the center of mass. Then, Eq. (23) is employed using the first stage of the control block diagram, basically the impedance control as previously presented in Fig. 2(a).

The main objective of this simulation is to obtain the minimum norm of the joint angular rates for the robot manipulators while their end-effectors track the desired grapple fixture trajectories on the target spacecraft. The conventional inverse kinematics functions cannot directly be applied, and thus, the desired joint values and rates cannot uniquely be calculated since the system consists of redundant manipulators, that is, $n > 6$. The desired trajectories for the end-effectors were generated in the orbital plane as shown in Fig. 3(a–f).

The variation of minimum norm joint rates for the left and the right arms with time is plotted in Fig. 4(a–b)

The joint angles are obtained by integrating the joint rates using Eq. (21) and plotted in Fig. 5(a–b).

The joint accelerations for each robot are also calculated using Eq. (22) and are plotted with time in Fig. 6(a–b).

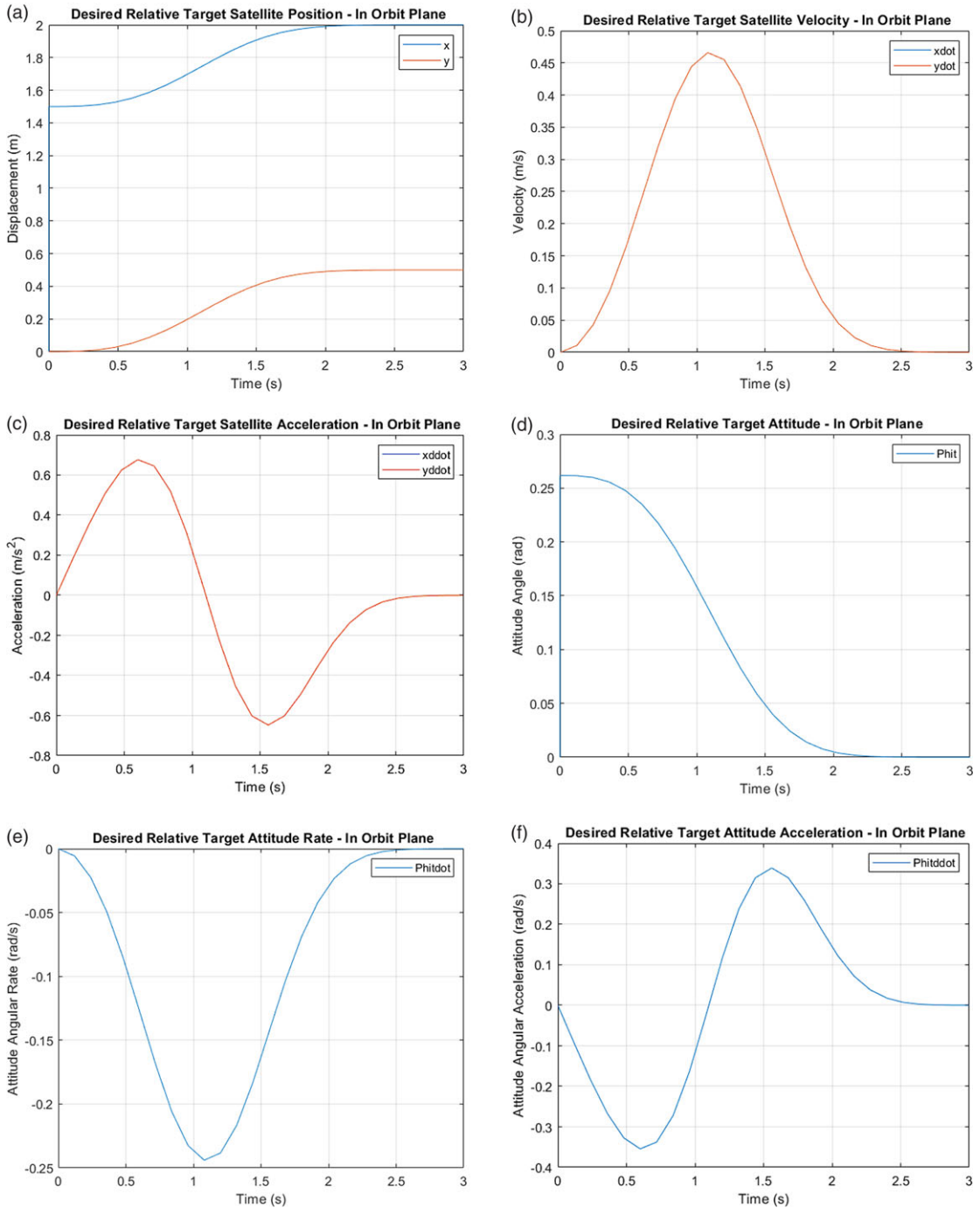


Figure 3. (a–f) Target linear and rotational displacement, velocity, and acceleration with time.

The chaser spacecraft control force $\tilde{\mathbf{F}}_c$ and moments $\tilde{\boldsymbol{\tau}}_c$ are also calculated using the inverse dynamics equations as previously developed in Eq. (11) and are plotted in Fig. 7(a–b) below. These are the forces and moments transferred from the two robots to the chaser spacecraft due to the inertial angular and linear accelerations of the links.

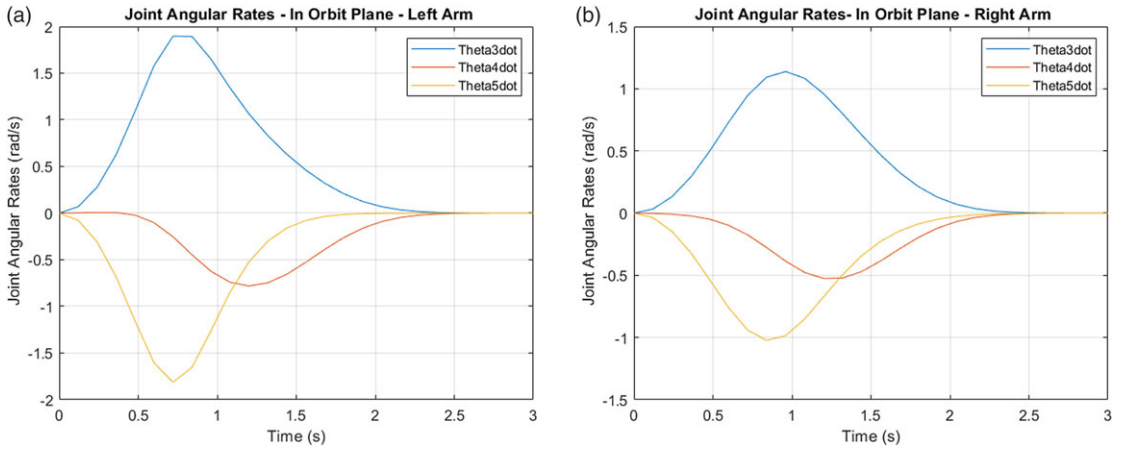


Figure 4. (a–b) Minimum norm of left and right arm joint rates.

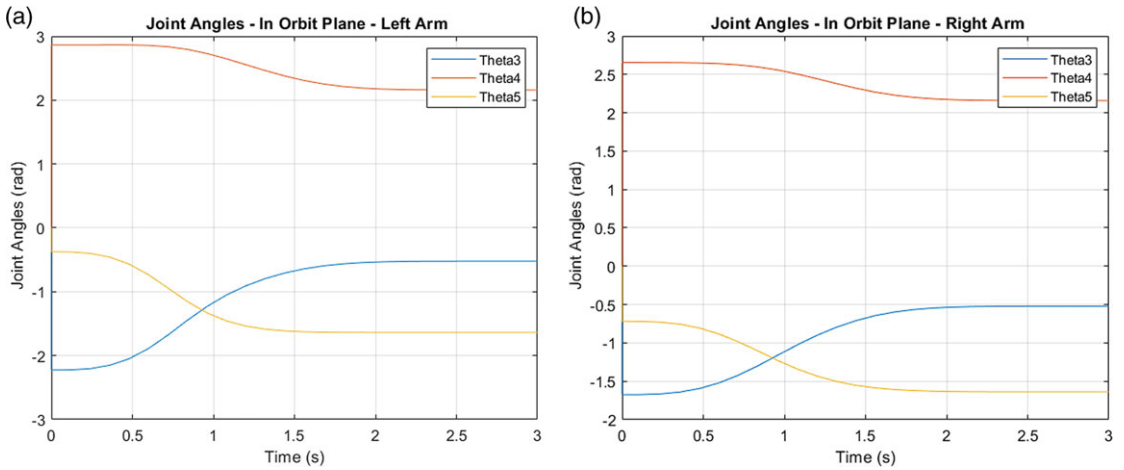


Figure 5. (a–b) Variation of optimal joint angles for left and right arms.

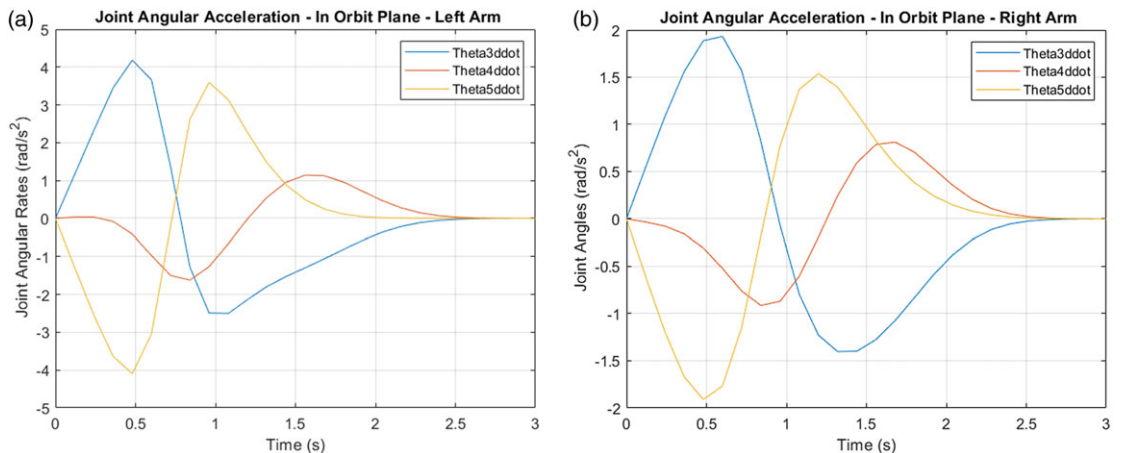


Figure 6. (a–b) Variation of optimal in-orbit joint angular accelerations for left and right arms.

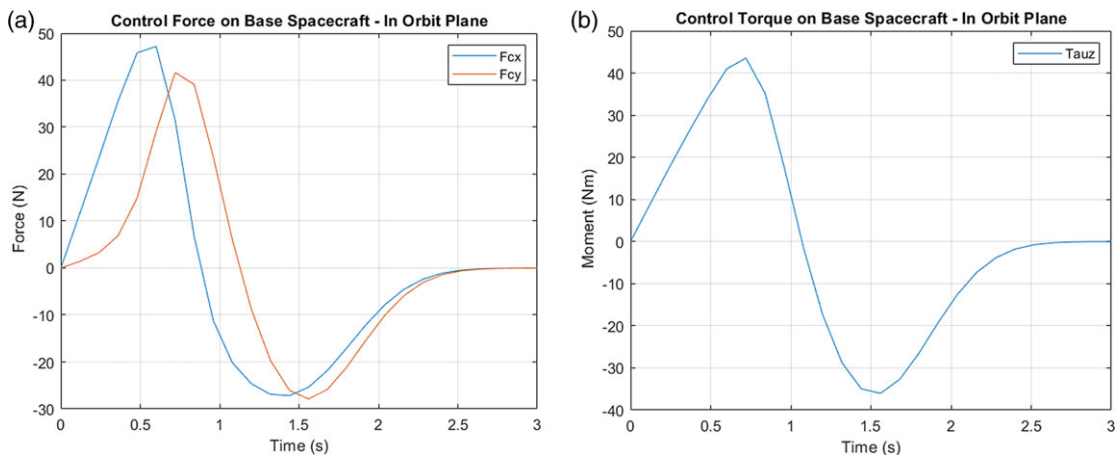


Figure 7. (a–b) Variation of chaser spacecraft control force and moments.

3.2. Post-capture mode of operation using OCA and NMPC

In this simulation, the desired trajectory for the center of mass of the target spacecraft was generated and in turn the trajectories for the left and right end-effectors were obtained while they were holding a common payload (i.e., the target spacecraft). Again, the impedance control technique as illustrated in the first stage of the control block diagram was employed for the trajectory generation. The main objective in this simulation is to obtain the minimum norm of the joint torques and chaser spacecraft control moments and forces, as well as the contact forces and moments exerted on the payload by the two robot manipulators while tracking a desired end-effector pose using both OCA and NMPC algorithms and perform a comparative analysis and discuss the relative merits of these two approaches. The desired trajectories for the end-effectors are shown in the orbital plane as shown in Fig. 8(a–f).

The minimum norm of the joint torques and the contact forces and moments exerted on the payload by the two robot manipulators while tracking a desired end-effector pose were plotted in Fig. 9(a–i) below using both OCA and NMPC techniques. The minimum norm in-orbit plane joint torques obtained by OCA (red lines) and NMPC (yellow lines) are plotted against the in-orbit joint torques (blue lines) obtained by only minimizing the joint rates for comparison purposes.

The comparison of the joint torques clearly demonstrates that the minimization scheme by the NMPC algorithm provides the best results (i.e., the lowest joint torques, followed by the OCA and then the conventional least-square technique which is commonly used in the literature).

The NMPC approach does better in minimizing the joint torques although following a slightly different trajectory, yet arriving at the same desired destination pose. One of the reasons for this is that the performance index (i.e., the cost function) defined in Eq. (38) involves the future state estimates as well as the control inputs, and the minimization process involves not only the current state errors and current joint torques but also the future state values, and joint torques over a specified prediction horizon/future timeline. The OCA solution can be deemed as a subset solution if the prediction horizon is only assigned to the current time. Furthermore, the NMPC is a natural progression from the OCA and can be employed depending on the availability of the on-board space computing resources.

The computational complexity of the NMPC is higher than that of OCA. Especially, if one wants to use a continuous model for the NMPC estimates. This was one of the main reasons as to why the literature to date mostly focused on linear models. However, the dual arm coordination is highly non-linear, and thus, there is no MPC application on dual arm coordination. The proposed discretized technique offers exact realizations (of a non-linear model) with elegance and simplicity and yet takes into account the full non-linear model of the dual arm coordinating system as demonstrated in the simulations.

The chaser-base spacecraft control force \vec{F}_c and moments $\vec{\tau}_c$ are also calculated using the inverse dynamics equations as previously developed in Eq. (11) and are plotted in Fig. 10(a–b) below. Again,

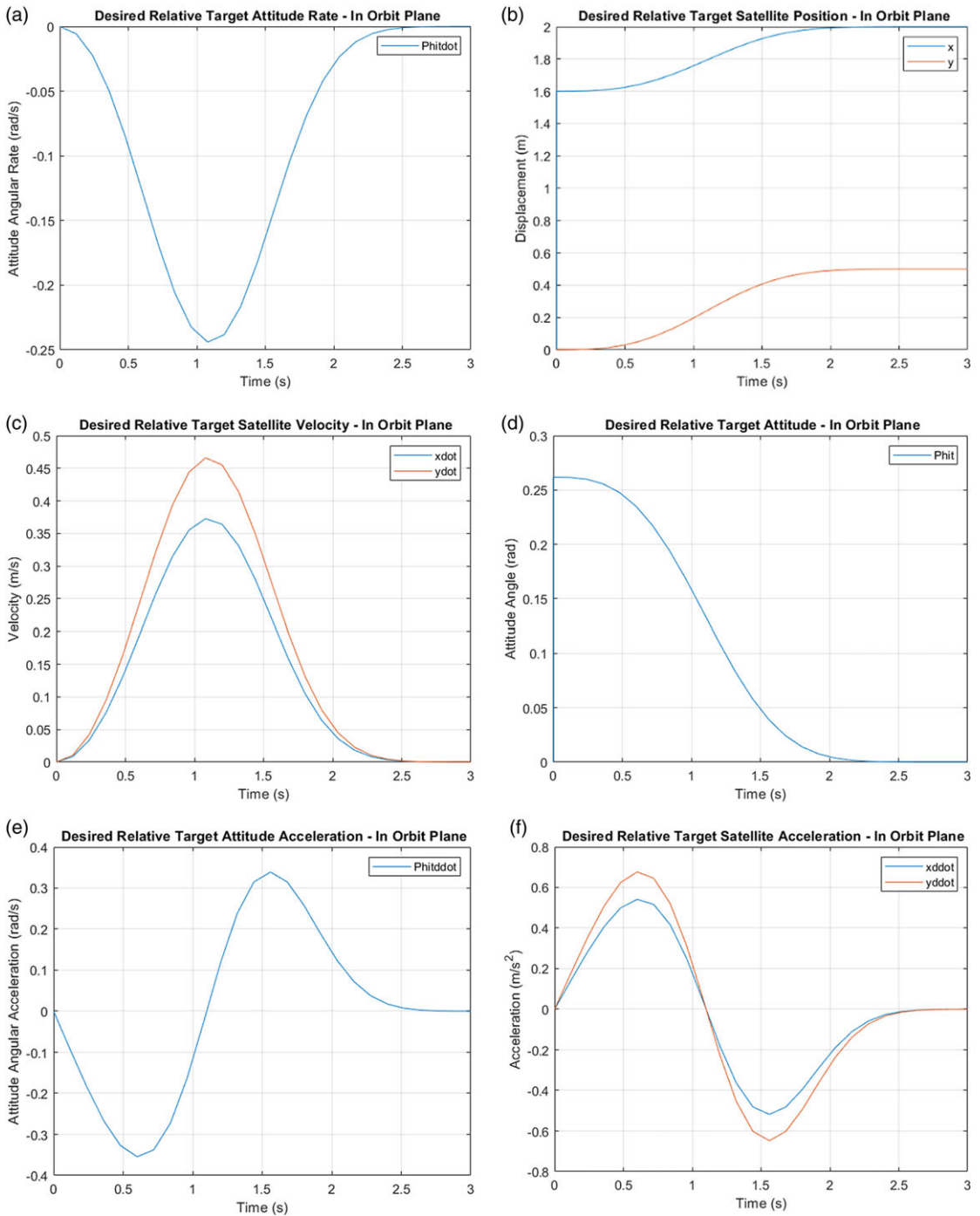


Figure 8. (a–f) Target linear and rotational displacement, velocity, and acceleration with time.

these are the forces and moments transferred from the two robots to the chaser spacecraft due to the inertial angular and linear accelerations of the links.

The joint accelerations for each robot are also calculated using Eq. (14) using both OCA and NMPC and are plotted with time in Fig. 11(a–b).

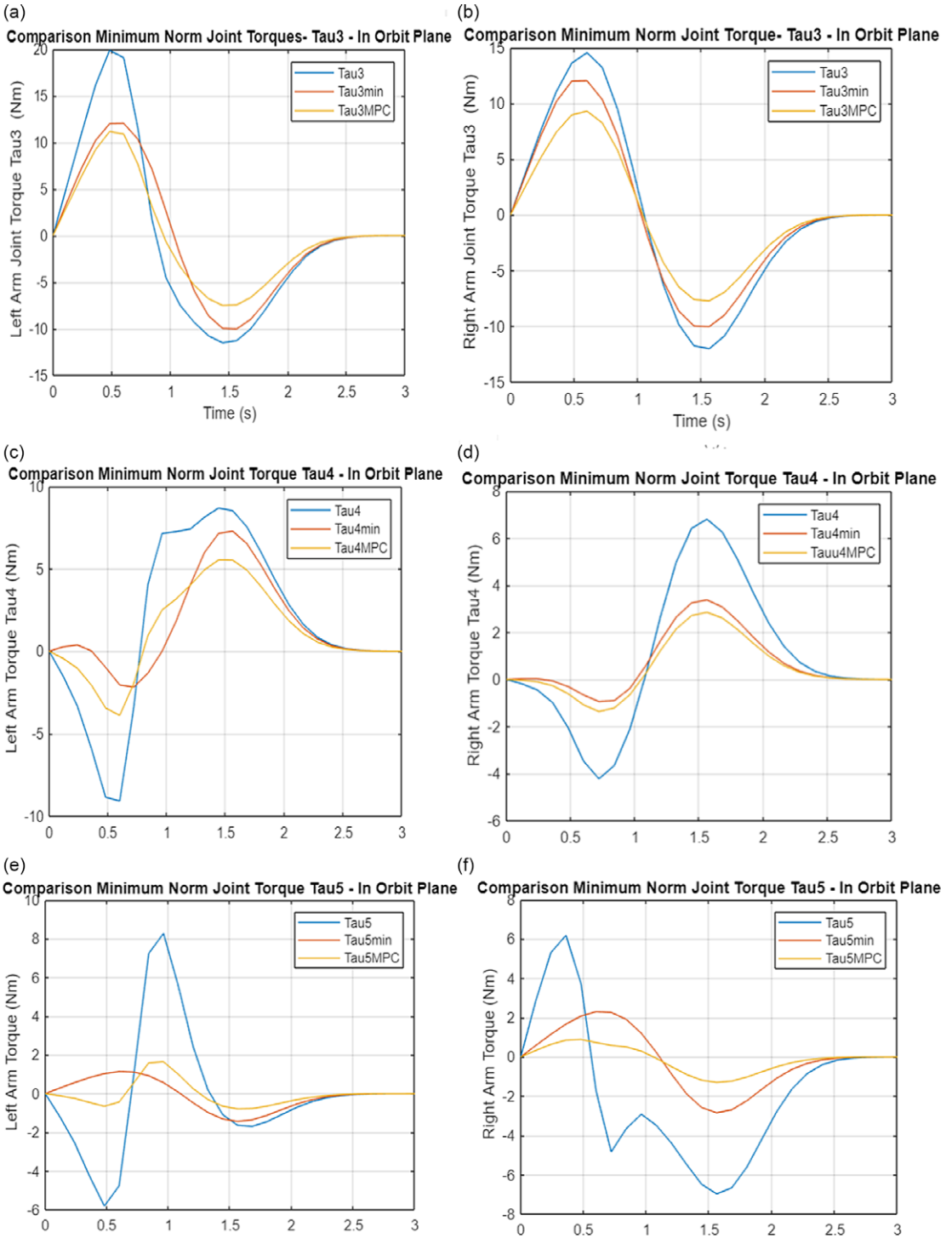


Figure 9. (a–i) Minimum joint torques (left and right arms) and the contact forces/moments.

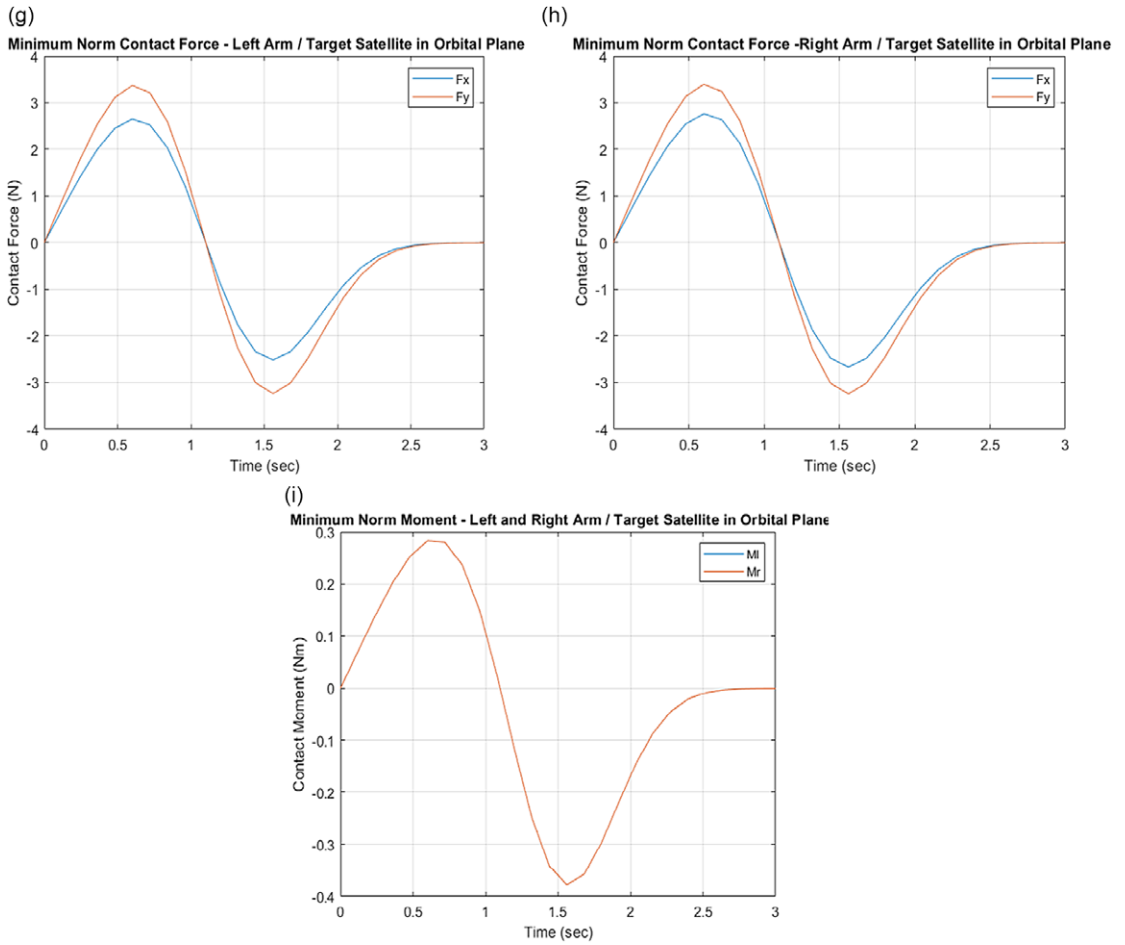


Figure 9. Continued.

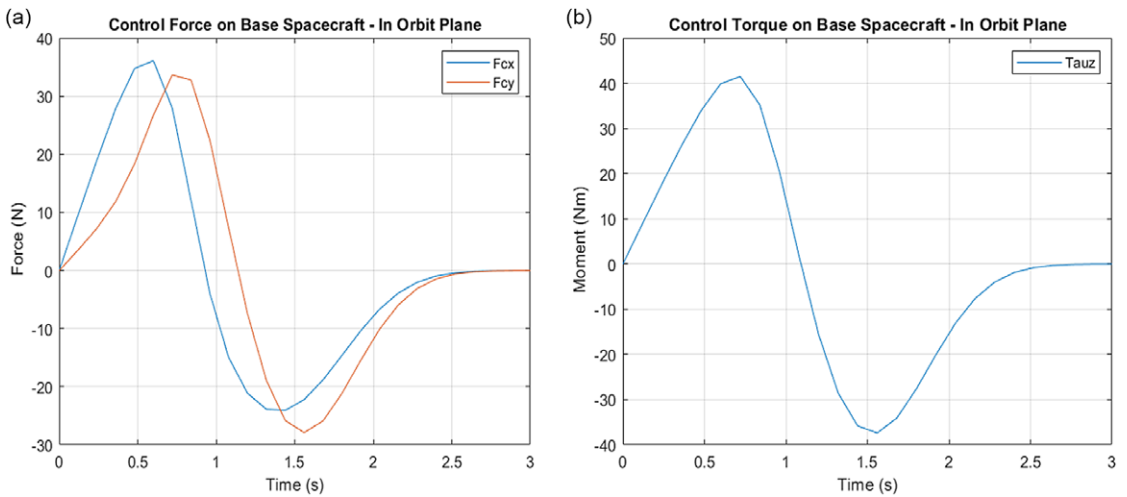


Figure 10. (a–b) Variation of chaser spacecraft control force and moments.

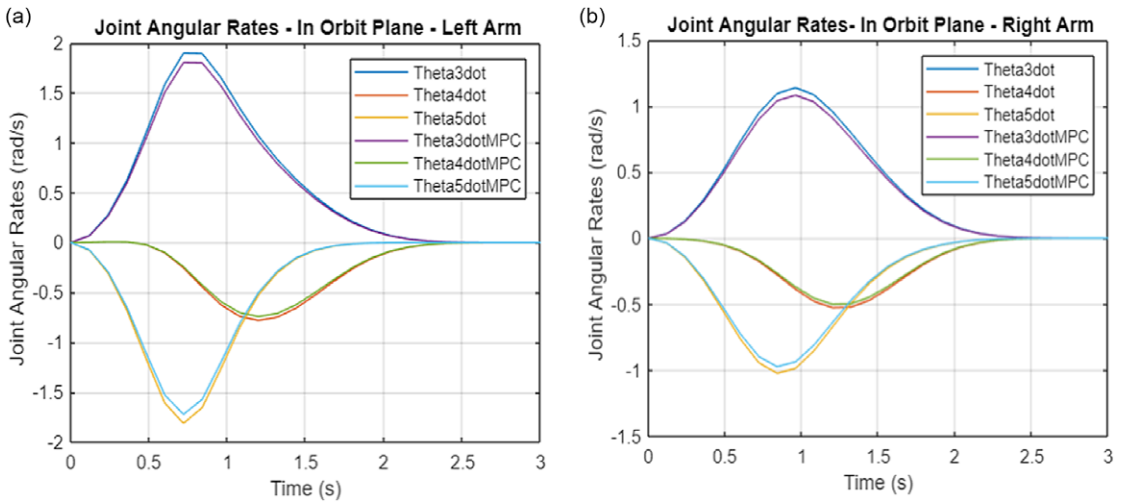


Figure 11. (a–b) Variation of in-orbit joint angular accelerations for left and right arms.

The joint angular rates and angles were obtained by integrating the joint accelerations using Eq. (13) and were plotted in Fig. 12(a–d).

3.3 Simulation demonstration of performance of the NMPC in tracking control

In this section, two sets of trajectories provided in [55] are simulated to demonstrate the performance of the developed NMPC in tracking control.

Trajectory 1: The first trajectory is a circle with a radius of 0.2 m centered at (0.8 m 1 m). The simulation results demonstrating tracking performance are presented in Fig. 13(a–e). The simulation results showed that the NMPC algorithm worked very efficiently to follow the circular trajectory. The Cartesian position of the end-effector is plotted in Fig. 13a while the Cartesian end-effector velocities are presented in Fig. 13b. The time variation of joint angles, angular rates, and the minimum norm joint torques values are presented in Fig. 13(c–e).

Trajectory 2: The second trajectory is highly intricate and first appeared in [56]. This highly complex and intricate trajectory showcases the capability of the proposed NMPC algorithm in tasks requiring accurate tracking of fast and complex trajectories while the robot actuators are subject to torque saturation limits, that is, constraints.

The end-effector is required to track the following trajectory:

$$\begin{aligned}
 x_d &= 0.5 + 0.02 \sin(2t) + 0.02 \sin(12t) \\
 y_d &= 0.26 + 0.02 \sin(2t + 1.6) + 0.02 \sin(12t + 1.6)
 \end{aligned}
 \tag{41}$$

Part 1: The complex trajectory tracking simulation in Part 1 established torque saturation limits $\tilde{S}_{\max/\min}$ of ± 5 N-m and ± 4 N-m for joints 3 and 4, respectively. Figure 14(a–g) displays the simulation results, which show that the NMPC algorithm is highly effective in tracking the complex trajectory. Figure 14(a) displays the Cartesian position of the end-effector, while Fig. 14(b–c) shows the tracking error in Cartesian displacement of the end-effector. Figure 14(d) illustrates time variation of the Cartesian velocity of the end-effector. Figure 14(e–g) displays the changes in joint angles over time, angular rates, and minimum norm joint torque values, respectively.

In Part 1, the complex trajectory tracking simulation established torque saturation limits of ± 5 N-m and ± 4 N-m for joints 3 and 4, as shown in Fig. 14(g), and these limits were not exceeded.

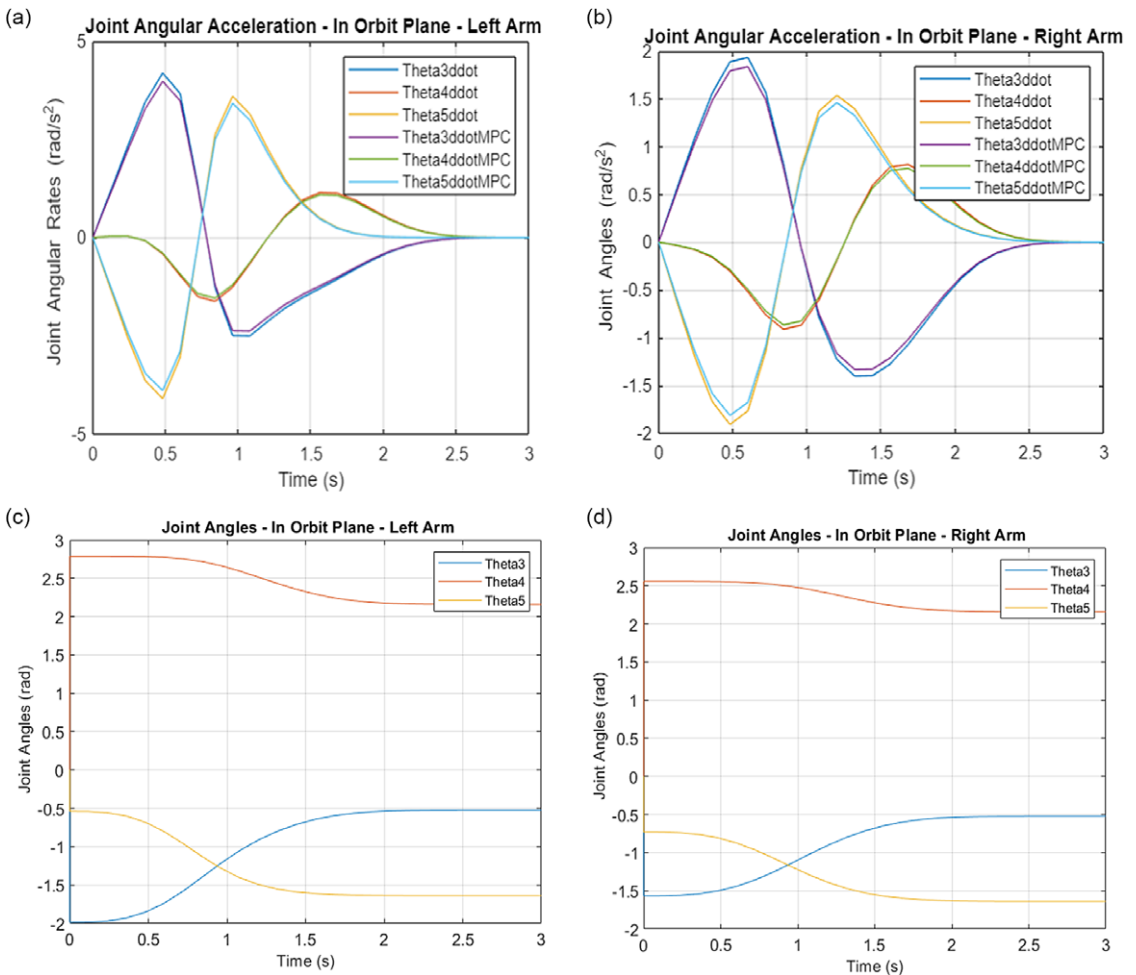


Figure 12. (a–d) Variation of in-orbit joint angular rates and angles for the left and right arms.

Part 2: To evaluate the impact of these constraints on tracking performance, the joint torque limits for joints 3 and 4 were deliberately reduced to ± 3 N-m and ± 2 N-m, respectively. Figure 15(a–d) shows the performance of the NMPC controller in tracking the complex trajectory with these tighter joint torque constraints.

As seen in Fig. 15(a–d), the tracking performance was negatively affected by the constraint on joint torque saturation limits.

Figure 16(a–b) displays the tracking errors linked to two distinct sets of joint torque constraints. The simulation outcomes demonstrate that implementing the lower torque limits (± 2 N-m) rather than the higher limits (± 3 N-m) resulted in relatively greater tracking errors. Nevertheless, the overall performance of the NMPC algorithm during a complex trajectory with these strict joint saturation limits remains impressive.

4. Conclusion and future work

The problem of saturation of spacecraft and robot actuator torques is a formidable challenge during the post-capture stage while controlling a target spacecraft with its uncontrolled large angular and linear momentums.

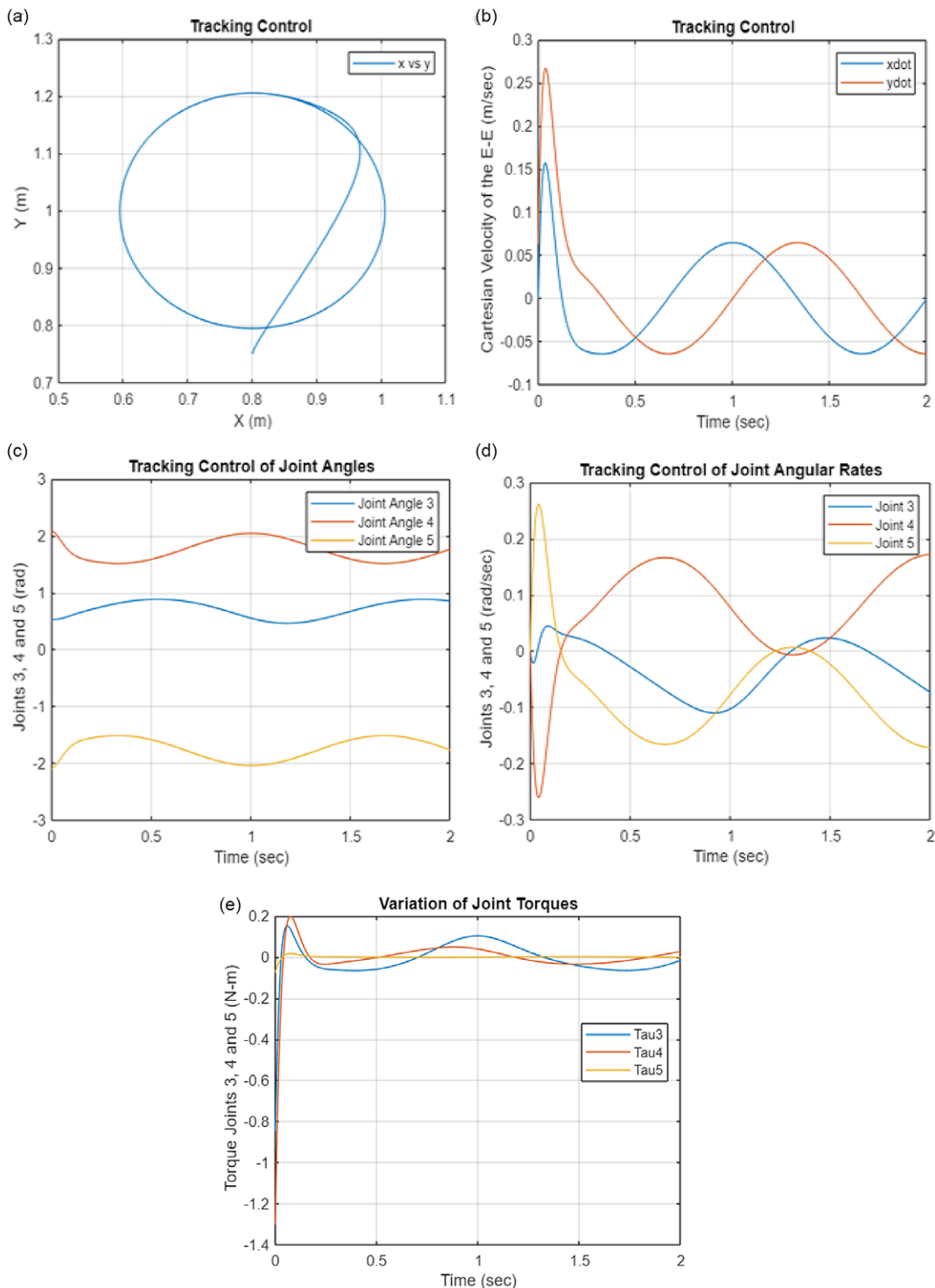


Figure 13. (a–b) Tracking performance. (c–d) Tracking performance – variation of joint angles and rates. (e) Tracking performance – variation of minimum joint torques.

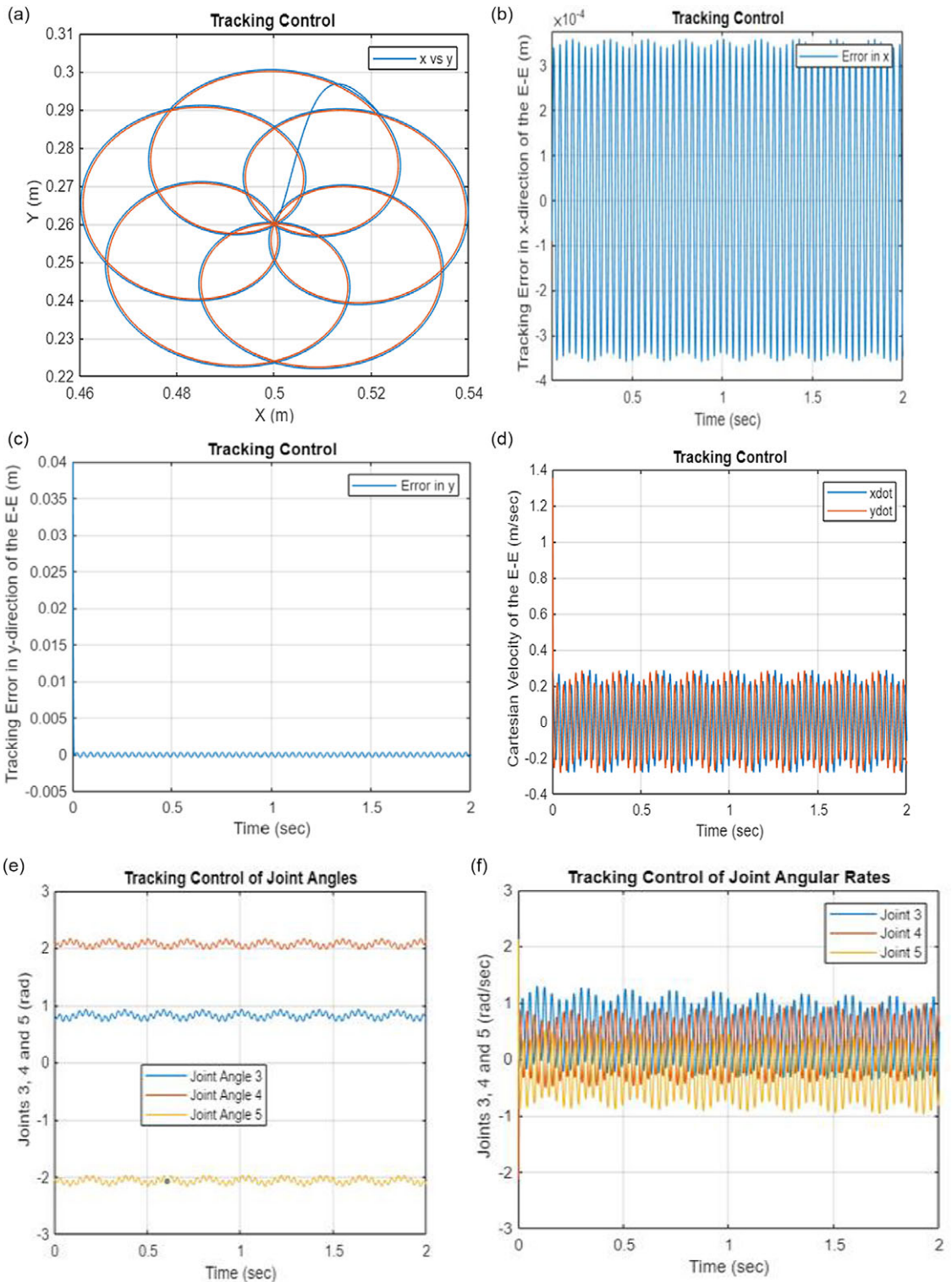


Figure 14. (a–d) Tracking performance. (e–f) Tracking performance – variation of joint angles and rates. (g) Tracking performance – variation of minimum joint torques.

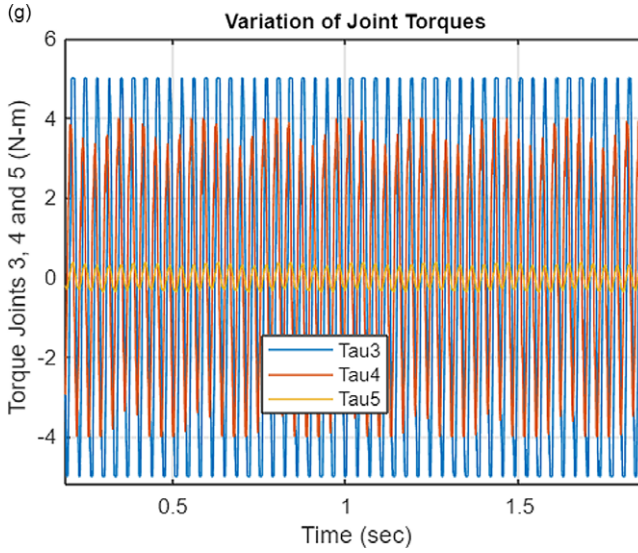


Figure 14. Continued.

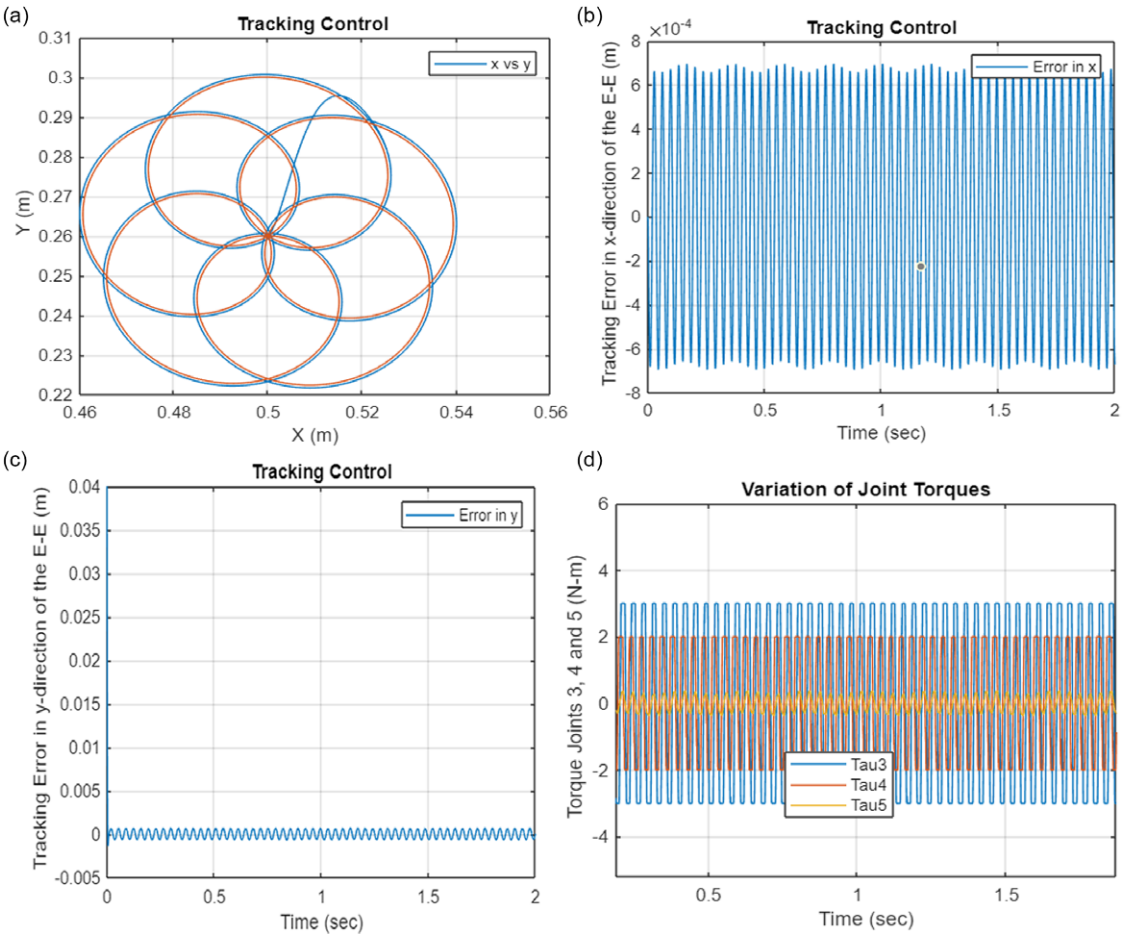


Figure 15. (a) Tracking performance. (b–d) Tracking performance – variation of minimum joint torques.

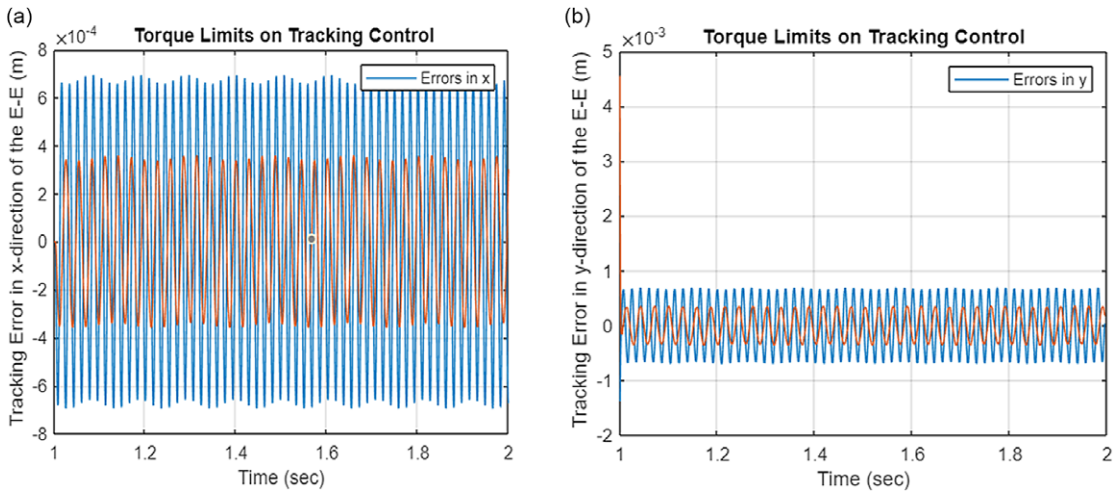


Figure 16. (a–b) Comparisons of tracking performance with different torque constraints.

This paper presented a novel solution to this problem by virtue of two control algorithms based on (i) Optimal Control Allocation (OCA) and (ii) Non-linear Model Predictive Control (NMPC). The main objective of this paper was to develop a robust and computationally efficient real-time control algorithm which minimized a performance index involving the norm of the joint torques, the chaser spacecraft control moments and forces, as well as the contact forces and moments exerted on the payload by two robot manipulators while tracking a desired end-effector pose. The computer simulation results showed that both the OCA and the NMPC algorithms worked efficiently, and the minimum joint torques, spacecraft control moments, and forces and the contact forces and moments were obtained in real-time successfully while the end-effectors were carrying a common payload and successfully tracking a desired payload trajectory. The comparison of the joint torques clearly demonstrated that the minimization scheme by the NMPC algorithm provided the best results (i.e., the lowest joint torques, followed by the OCA and then the conventional least-square technique which is commonly used in the literature).

The NMPC approach did better in minimizing the joint torques although following a slightly different trajectory, yet arriving the same desired destination pose. One can conclude that one of the reasons for this was that the performance index (i.e., the cost function) defined in Eq. (38) involved the future state estimates as well as the control inputs, and the minimization process involved not only the current state errors and current joint torques but also the future state values and joint torques over a specified prediction horizon / future timeline. Thus, the proposed OCA solution can be deemed as a subset solution if the prediction horizon is only assigned to the current time. Furthermore, the NMPC can be regarded as a natural progression from the OCA and be employed depending on the availability of the on-board space computing resources.

It is also observed that the computational complexity of the NMPC was higher than that of the OCA, especially if one wants to employ a continuous non-linear model for the NMPC estimates. One can state that this was perhaps one of the main reasons as to why the literature to date mostly focused on the linear models. However, the dual arm coordination is highly non-linear. The proposed NMPC discretized technique offered exact realizations (of a non-linear model) with elegance and simplicity and yet took into account the full non-linear model of the dual arm coordinating system as demonstrated in the simulations.

The NMPC algorithm was also tested in tracking a highly complex and intricate trajectory while the robot actuators were under strict torque saturation limits. The computer simulation revealed that while the joint torque saturation limits negatively impacted the tracking performance, the overall performance of the NMPC algorithm during the complex trajectory remained impressive despite these strict constraints.

The developed NMPC is facing certain limitations, which we plan to address in our upcoming research work. The performance of NMPC can be negatively impacted if there are inaccuracies in the modeling process. Furthermore, NMPC necessitates offline tuning of its parameters and may not handle disturbances effectively if they are not precisely modeled. Additionally, the current saturation policy lacks a systematic approach for handling output constraints and does not take into account future predictions of inputs.

Acknowledgments. This research was supported by the National Science and Engineering Research Council of Canada.

Authors' contributions. All authors contributed to the study conception and design. Initial material preparation and analysis were performed by Serdar Kalaycioglu. The first draft of the manuscript was written by Serdar Kalaycioglu and Anton de Ruiter. All authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Conflicts of interest. The authors declare no conflict of interest.

Ethical considerations. None.

References

- [1] *Space Foundation White Paper, U.S. Satellite Servicing Policy – An Overview* (Washington Strategic Operations, 2020). https://www.spacefoundation.org/white_papers/u-s-satellite-servicing
- [2] *NASA's Robotic OSAM-1 Mission Completes its Critical Design Review* (NASA Feature, 2022). <https://www.nasa.gov/feature/goddard/2022/nasa-s-robotic-osam-1-mission-completes-its-critical-design-review>
- [3] D. Krebs-Gunter, *OSAM 2 (Archinaut 1)* (Gunter's Space Page, 2022). https://space.skyrocket.de/doc_sdat/osam-2.htm
- [4] *Orbital Hub DLR Vision 2025* (DLR Institute of Space Systems Analysis Space Segment, Bremen, 2022). https://www.researchgate.net/publication/328495824_Orbital-Hub_DLR
- [5] A. Flores-Abad, O. Ma, K. Pham and S. Ulrich, "A review of space robotics technologies for on-orbit servicing," *Prog. Aerosp. Sci.* **68**, 1–26 (2014), <https://doi.org/10.1016/j.paerosci.2014.06.002>.
- [6] W. Xu, J. Peng, B. Liang and Z. Mu, "Hybrid modeling and analysis method for dynamic coupling of space robots," *IEEE Trans. Aerosp. Electron. Syst.* **52**(1), 85–98 (2016).
- [7] W. Xu, B. Liang and Y. Xu, "Survey of modeling, planning, and ground verification of space robotic systems," *Acta Astronaut.* **68**(11-12), 1629–1649 (2011).
- [8] S. Matsumoto, S. Jacobsen, S. Dubowsky and Y. Ohkami, "Approach Planning and Guidance for Uncontrolled Rotating Satellite Capture Considering Collision Avoidance," *In: International Symposium on Artificial Intelligence, Robotics and Automation in Space*, Nara, Japan (2003).
- [9] Z. Ma, O. Ma and B. Shashikanth, "Optimal approach to and alignment with a rotating rigid body for capture," *J. Astronaut. Sci.* **55**(4), 407–419 (2007).
- [10] A. Fejzic, "Development of Control and Autonomy Algorithms for Docking to Complex Tumbling Satellites," *Technical Report* (Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, 2008).
- [11] W. Xu, B. Liang, C. Li and Y. Xu, "Autonomous rendezvous and robotic capturing of non-cooperative target in space," *Robotica* **28**(05), 705–718 (2010).
- [12] Z. Vafa and S. Dubowsky, "The kinematics and dynamics of space manipulators: The virtual manipulator approach," *Int. J. Robot. Res.* **9**(4), 3–21 (1990).
- [13] K. Yamada, "Arm Path Planning for a Space Robot," *In: IEEE/RSJ International Conference on Intelligent Robots and Systems*, Yokohama, Japan, vol. 3 (1993) pp. 2049–55.
- [14] K. Yamada, S. Yoshikawa and Y. Fujita, "Arm path planning of a space robot with angular momentum," *Adv. Robot.* **9**(6), 693–709 (1994).
- [15] P. Gasbarri and A. Pisculli, "Dynamic/control interactions between flexible orbiting space-robot during grasping, docking and post-docking maneuvers," *Acta Astronaut.* **110**, 225–238 (2015), <https://doi.org/10.1016/j.actaastro.2014.11.001>.
- [16] X. Cyril, A. K. Misra, M. Ingham and G. J. Jaar, "Post capture dynamics of a spacecraft-manipulator-payload system," *J. Guid. Control Dyn.* **23**(1), 95–100 (2000).
- [17] N. Dimitrov and Yoshida K., "Momentum Distribution in a Space Manipulator for Facilitating the Post-impact Control," *In: IEEE/RSJ International Conference on Intelligent Robots and Systems* (2004) pp. 3345–3350.
- [18] D. Wang, P. Huang and Z. Meng, "Coordinated stabilization of tumbling targets using tethered space manipulators," *IEEE Trans. Aerosp. Electron. Syst.* **51**(3), 2420–2432 (2015).
- [19] J. Suresh and G. Kelkar, "Spacecraft Stabilization and Control for Capture of Noncooperative Space Objects," NASA/TMC2014-218668 (2014).
- [20] F. Aghili, "Coordination Control of a Free-Flying Manipulator and Its Base Attitude to Capture and Detumble a Noncooperative Satellite," *In: International Conference on Intelligent Robots and Systems* (2009) pp. 2365–2372.

- [21] W. Xu, C. Li, B. Liang, Y. Xu, Y. Liu and W. Qiang, "Target berthing and base reorientation of free-floating space robotic system after capturing," *Acta Astronaut.* **69**(2-3), 109–126 (2009).
- [22] C. C. Cheah, S. Kawamura and Arimoto S., "Feedback Control for Robotic Manipulator with Uncertain Kinematics and Dynamics," **In: IEEE International Conference on Robotics and Automation, Proceedings** (1998) pp. 3607–3612.
- [23] W. E. Dixon, "Adaptive regulation of amplitude limited robot manipulators with uncertain kinematics and dynamics," *IEEE Trans. Autom. Control* **52**(3), 488–493 (2007).
- [24] C. C. Cheah, C. Liu and J. J. E. Slotine, "Approximate Jacobian Adaptive Control for Robot Manipulators," **In: IEEE International Conference on Robotics and Automation, Proceedings** (2004) pp. 3075–3080.
- [25] D. Wang, P. Huang and Z. Meng, "Coordinated stabilization of tumbling targets using tethered space manipulators," *IEEE Trans. Aerosp. Electron. Syst.* **51**(3), 2420–2432 (2015).
- [26] B. Zhang, B. Liang, Z. Wang, Y. Mi, Y. Zhang and Z. Chen, "Coordinated stabilization for space robot after capturing a non-cooperative target with large inertia," *Acta Astronaut.* **134**, 75–84 (2017), <https://doi.org/10.1016/j.actaastro.2017.02.019>.
- [27] S. A. Schneider and R. H. Cannon Jr, "Object impedance control for cooperative manipulation: Theory and experimental results," *IEEE Trans. Robot. Autom.* **8**(3), 383–394 (1992).
- [28] S. A. A. Moosavian, R. Rastegari and E. Papadopoulos, "Multiple impedance control for space free-flying robots," *J. Guid. Control Dyn.* **28**(5), 939–947 (2005).
- [29] M. Vukob, S. Gros, G. Horn, G. Frison, K. Geebelen, J. B. Jørgensen, J. Swevers and M. Diehl, "Real-Time nonlinear MPC and MHE for a large-scale mechatronic application," *Control Eng. Pract.* **45**, 64–78 (2015), <https://doi.org/10.1016/j.conengprac.2015.09.009>.
- [30] J. B. Rawlings, "Tutorial overview of model predictive control," *IEEE Control Syst.* **20**(3), 38–52 (2000).
- [31] Y. Shi and K. Zhang, "Advanced model predictive control framework for autonomous intelligent mechatronic systems: A tutorial overview and perspectives," *Annu. Rev. Control* **52**, 170–196 (2021), <https://doi.org/10.1016/j.arcontrol.2021.06.001>.
- [32] P. D. Christofides, R. Scattolini, D. Muñoz de la Peña and J. Liu, "Distributed model predictive control: A tutorial review and future research directions," *Comput. Chem. Eng.* **51**, 21–41 (2013), <https://doi.org/10.1016/j.compchemeng.2012.07.011>.
- [33] H. Ellis, H. Durand and P. D. Christofides, "A tutorial review of economic model predictive control methods," *J. Process Control* **24**(8), 1156–1178 (2014).
- [34] M. Fin, "Implementation of Linear Model Predictive Control –Tutorial," (2021) arXiv preprint arXiv:2109.11986. <https://doi.org/10.48550/arXiv.2109.11986>.
- [35] R. Gangapersaud, G. Liu and A. de Ruiter, "Detumbling of a non-cooperative target with unknown inertial parameters using a space robot," *Adv. Space Res.* **63**(12), 3900–3915 (2019).
- [36] Y. Shuyou, R. Marcus, C. Hong and A. Frank, "Inherent robustness properties of quasi-infinite horizon nonlinear model predictive control," *Automatica* **50**(9), 2269–2280 (2014).
- [37] H. Wei, C. Shen and Y. Shi, "Distributed Lyapunov-based model predictive formation tracking control for autonomous underwater vehicles subject to disturbances," *IEEE Trans. Syst. Man Cybern. Syst.* **51**(8), 5198–5208 (2021).
- [38] H. Wei, Q. Sun, J. Chen and Y. Shi, "Robust distributed model predictive platooning control for heterogeneous autonomous surface vehicles," *Control Eng. Pract.* **107**, Article104655 (2021), <https://doi.org/10.1016/j.conengprac.2021.104655>.
- [39] K. Zhang, Q. Sun and Y. Shi, "Trajectory tracking control of autonomous ground vehicles using adaptive learning MPC," *IEEE Trans. Neur. Netw. Learn. Syst.* **32**(12), 5554–5564 (2021).
- [40] Y. Zou, X. Su, S. Li, Y. Niu and D. Li, "Event-Triggered distributed predictive control for asynchronous coordination of multi-agent systems," *Automatica* **99**, 92–98 (2019), <https://doi.org/10.1016/j.automatica.2018.10.043>.
- [41] K. Zhang and Y. Shi, "Adaptive model predictive control for a class of constrained linear systems with parametric uncertainties," *Automatica* **117**, Article108974 (2020), <https://doi.org/10.1016/j.automatica.2020.108974>.
- [42] J. S. Ladoiye, D. S. Neculescu and J. Z. Sasiadek, "Force Control of Surgical Robot with Time Delay using Model Predictive Control," **In: ICINCO** (2018).
- [43] T. Englert, F. M. a. Volz, S. Rhein and K. Graichen, "A software framework for embedded nonlinear model predictive control using a gradient-based augmented Lagrangian approach (GRAMPC)," *Optim. Eng.* **20**(3), 769–809 (2019).
- [44] K. M. M. Rathai, "Synthesis and Real-time Implementation of Parameterized NMPC Schemes for Automotive Semi-active Suspension Systems," *Ph.D. Thesis, Grenoble INP* (Communauté Université Grenoble Alpes, Grenoble, France, 2020).
- [45] R. Quirynen, M. Vukob, M. Zanon and M. Diehl, "Autogenerating microsecond solvers for nonlinear MPC: A tutorial using ACADO integrators," *Optim. Control Appl. Methods* **36**(5), 685–704 (2015).
- [46] F. Aghili, "Optimal Control of a Space Manipulator for Detumbling of a Target Satellite," **In: IEEE International Conference on Robotics and Automation** (2009) pp. 3019–3024.
- [47] E. Psomiadis and E. Papadopoulos, "Model-Based/Model Predictive Control Design for Free Floating Space Manipulator Systems," **In: Proceedings of the IEEE 30th Mediterranean Conference on Control and Automation (MED)**, Vouliagmeni, Greece (2022).
- [48] T. Rybus, K. Seweryn and J. Sasiadek, "Application of predictive control for manipulator mounted on a satellite," *Arch. Control Sci.* **28**(1), 105–118 (2018).
- [49] M. Wang, J. Luo and U. Walter, "A non-linear model predictive controller with obstacle avoidance for a space robot," *Adv. Space Res.* **57**(8), 1737–1746 (2016).
- [50] M. M. Morato, J. E. Normey-Rico and O. Sename, "Model predictive control design for linear parameter varying systems: A survey," *Annu. Rev. Control* **49**, 64–80 (2020), <https://doi.org/10.1016/j.arcontrol.2020.08.001>.
- [51] G. Sebestyen, "Low Earth Orbit Satellite Design," **In: Space Technology Library**, vol. 36 (Springer, Cham, 2018). doi: [10.1007/978-3-319-68315-7_7](https://doi.org/10.1007/978-3-319-68315-7_7).

- [52] J. Craig, *Introduction to Robotics: Mechanics and Control* (Addison-Wesley, Reading, MA, 1989).
- [53] R. Rouki, S. Kalaycioglu and Patel R., “Control Moment Gyro Based Spacecraft Control Using Feedback Linearization,” *In: Advances in Astronautical Sciences* (1997) pp. 983–997.
- [54] S. Kalaycioglu and A. de Ruiter, “Coordinated Motion and Force Control of Multi-Rover Robotics System with Mecanum Wheels,” *In: Proceedings of IEEE IEMTRONICS Conference*, Toronto, Canada (2022).
- [55] A. Izadbakhsh and S. Khorashadizadeh, “Polynomial-Based Robust Adaptive Impedance Control of Electrically Driven Robots,” *In: Robotica*, vol. 39 (Cambridge University Press, Cambridge, 2020) pp. 1181–1201. doi: [10.1017/S0263574720001009](https://doi.org/10.1017/S0263574720001009).
- [56] A. Izadbakhsh and M.-R. Rafiei, “Endpoint perfect tracking control of robots — A robust non-inversion-based approach,” *Int. J. Control Autom. Syst.* **7**(6), 888–898 (2009).

Appendix

Table I. System parameters.

Hardware configuration item	Mass (kg)	Dimensions (m) (prism)
Spacecrafts 1 and 2	40	$0.5 \times 0.5 \times 0.3$
Common Payload	10	$0.4 \times 1 \times 0.4$
Joint/Link 1	1	$0.1 \times 0.1 \times 0.1$
Joint/Link 2	1	$0.1 \times 0.1 \times 0.1$
Link 3	3	$1 \times 0.1 \times 0.1$
Link 4	5	$1 \times 0.1 \times 0.1$
Joint/Link 5	3	$0.1 \times 0.1 \times 0.1$
Joint/Link 6	1	$0.1 \times 0.1 \times 0.1$
Joint/Link 7	3	$1 \times 0.1 \times 0.1$