Passivity based nonlinear model predictive control (PNMPC) of multi-robot systems for space applications

Serdar Kalaycioglu and Anton De Ruiter*

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

In the past 2 decades, there has been increasing interest in autonomous multi-robot systems for space use. They can assemble space structures and provide services for other space assets. The utmost significance lies in the performance, stability, and robustness of these space operations. By considering system dynamics and constraints, the Model Predictive Control (MPC) framework optimizes performance. Unlike other methods, standard MPC can offer greater robustness due to its receding horizon nature. However, current literature on MPC application to space robotics primarily focuses on linear models, which is not suitable for highly non-linear multi-robot systems. Although Nonlinear MPC (NMPC) shows promise for free-floating space manipulators, current NMPC applications are limited to unconstrained non-linear systems and do not guarantee closed-loop stability. This paper introduces a novel approach to NMPC using the concept of passivity to multi-robot systems for space applications. By utilizing a passivity-based state constraint and a terminal storage function, the proposed PNMPC scheme ensures closed-loop stability and a superior performance. Therefore, this approach offers an alternative method to the control Lyapunov function for control of non-linear multi-robot space systems and applications, as stability and passivity exhibit a close relationship. Finally, this paper demonstrates that the benefits of passivity-based concepts and NMPC can be combined into a single NMPC scheme that maintains the advantages of each, including closed-loop stability through passivity and good performance through one-line optimization in NMPC.

KEYWORDS

space robotics, model predictive control, multi-robot system, robotic manipulators, passivity

1 Introduction

During the last 2 decades, there has been a significant interest in free-flying autonomous multi-robot systems for space applications (Nanos and Papadopoulos, 2017). Such an autonomous robotics system may be employed to assemble large space structures (e.g., communications antennae, and telescopes), manufacture on-orbit as well as to provide services for other space assets (i.e., exchange of orbital removal units (ORU), refuelling of spacecrafts, removing space debris, etc.). Until recently, some of these operations (e.g., inspection, exchange of ORUs) have been carried out by astronauts as Extravehicular Activities (EVA) in low Earth orbit. However, extensive astronauts training, operations planning and preparations, the associated cost, schedule and time delays and inherent risky

nature of space environment for astronauts convinced space agencies to look into other more efficient alternatives such as autonomous robotics spacecrafts (Papadopoulos et al., 2021). Furthermore, in high Earth orbit or other unmanned space exploration missions this may be the only option available. One of the pioneers in demonstrating robotics spacecraft capabilities was the Engineering Test Satellite ETS-VII of JAXA and it consisted of chasing and capturing spacecrafts (Yoshida, 2003). The ARCHINAUT, which is developed by NASA and Made in Space, features multiple robotic arms that are designed to perform on-orbit assembly and precision manufacturing. Other systems that are capable of performing similar tasks include the Kontur, METERON, and SPIDERFAB. The Gateway Moon Space Station is currently being developed by NASA and other international organizations. Other projects that are currently in development include the Moon Village and the Mars exploration mission of the European Space Agency (ESA) and OrbitalHub by the German Space Agency (DLR) (Orbital Hub DLR Vision, 2022). NASA and MAXAR Technologies Inc. are currently developing a robotic spacecraft known as OSAM-1 (NASA's Robotic OSAM-1 Mission, 2022; Krebs D., 2022). It will be able to perform various tasks on-orbit including the assembly of satellites, debris removal and the refueling of them. In 2016, China's Aolong-1 satellite was launched and demonstrated the capabilities of space robotic vehicles by capturing and handling space debris. Through a robotic arm, the satellite was able to grasp and send the object it collected back on a re-entry trajectory. Small-scale robotic systems, exemplified by the Future Space Debris Removal Orbital Manipulator (FSDROM), are poised to make a significant contribution to upcoming space debris removal missions, according to Shyam et al. (2021). As reported by the European Space Agency (ESA) in 2019, the amount of space debris in Earth's orbit has exceeded 8,000 tons and continues to rise, posing a significant threat to both satellites and astronauts (Chatterjee, 2014). Consequently, the removal of space debris has emerged as a top priority in space missions (Space Foundation White Paper, 2020). Direct capture of objects is one approach for mitigating the issue, and it can be executed through rigid or flexible capture methods, as classified by Zhao et al., 2020. A variety of techniques for flexible direct capture, including harpoons, nets, and tentacles, have been suggested by Billot et al., 2014; Zhang and Huang, 2016; Forshaw et al., 2017. Such mechanisms for capture help mitigate the risk of collisions between debris and space robots, thereby minimizing the likelihood of failed capture attempts that could result in further debris generation, as stated by Biesbroek et al., 2017.

Space manipulators often encounter unidentifiable, rotating debris, leading to harm to both the robot's structure and its actuators. Thus, it is essential to implement a controlling law that is robust enough to sustain its performance in the event of actuator malfunction or failure. This circumstance is particularly probable when employing direct capture methods since the capture process can lead to significant effects on the spacecraft, as stated by Seweryn et al., 2022.

However, there are major technological challenges in autonomous robotics spacecrafts. First of all, missions may involve execution of uncertain tasks in an unstructured environment. There may be inherent uncertainties in the system (e.g., friction, geometry, stiffness, and damping, etc.) and payload parameters (e.g., mass/inertia, geometry, momentum, etc.) (Aghili, 2020).

The conventional model-based controllers are not robust to deal with these uncertainties on orbit. Ground testing as well as on-orbit characterization and evaluations are difficult and proved to be limited. Furthermore, the contact models and payload characteristics cannot be determined in advance when dealing with unknown debris, etc. Moreover, the actuators are subject to saturation and the controller must deal with these constraints. In addition, there may be actuator or sensor failures and the system has to work with limited degrees of freedom and sensory feedback in case of occurrence of faults (Raisi et al., 2022).

Therefore, conventional controllers cannot cope with these types of tasks, system and payload uncertainties, unknown payload and contact dynamics behaviour, constraints such as problem of force/torque saturation, and partial system failures such as faulty actuator or sensor. Recently, a new type of control system known as Model Predictive Control (MPC) has gained widespread attention from industry and academia (Hewing et al., 2020). There are also variations of MPC techniques such as implicit, explicit, adaptive, gain-scheduled, and non-linear and others (Rawlings, 2000; Christofides et al., 2013; Morato et al., 2020; Fin, 2021; Shi and Zhang, 2021).

Achieving optimal performance, stability, and robustness is crucial in space operations. The Model Predictive Control (MPC) framework addresses this challenge by considering system dynamics and constraints during performance optimization. Its receding horizon nature also makes it more robust than other methods (Shuyou et al., 2014). However, current literature on MPC application to space robotics mainly focuses on linear models, which are not suitable for highly non-linear multi-robot systems commonly found in space applications.

Despite its numerous advantages, Model Predictive Control (MPC) presents several challenges that require further investigation, including feasibility, nonlinearity, closed-loop stability, and robustness (Rybus et al., 2018; Psomiadis and Papadopoulos, 2022). When the models of the plant or the constraints are nonlinear, Nonlinear Model Predictive Control (NMPC) schemes must be employed (Shuyou et al., 2014; Quirynen et al., 2015; Vukov et al., 2015; Wang et al., 2016; Englert et al., 2019; Rathai, 2020; Kalaycioglu and de Ruiter, 2023). However, it has been observed in (Jadbabaie et al., 1999) that NMPC does not always ensure closed-loop stability. Additionally, the issue of robustness arises in MPC when model uncertainty or noise is present, as is often the case due to the inability of the prediction model to precisely match the actual dynamics of the plant being controlled (Nanos and Papadopoulos, 2011).

NMPC presents a promising solution for free-floating space manipulators. However, existing NMPC applications are limited to unconstrained non-linear systems and cannot guarantee closed-loop stability. Passivity theory is a potent instrument for examining and managing nonlinear systems, as demonstrated by numerous works by Wang. and Xie, 2014; Schaft 2000; Raff et al. (2007) introduced a significant advancement in this field through the proposal of a Nonlinear Model Predictive Control (NMPC) scheme based on passivity. This development was motivated by the interdependencies between optimal control, passivity, and NMPC. This scheme incorporates particular passivity-based constraints that ensure both closed-loop stability and feasibility.

To address this gap in space robotics, this paper proposes a novel approach to NMPC that uses the concept of passivity for multi-robot systems specifically designed for space-related applications. The proposed NMPC scheme utilizes a passivity-based state constraint to ensure closed-loop stability. Thus, this approach provides an alternative to the use of control Lyapunov function for controlling nonlinear multi-robot space systems and related applications. This is because stability and passivity are highly interrelated. The passivity based NMPC scheme offers a robust solution for optimizing performance while maintaining stability and ensuring system robustness in space robotics.

The main contribution of this paper lies in the application of the concept of passivity-based NMPC to multi-robot systems for space applications. While the idea of passivity-based NMPC has been previously explored by Raff et al. (2007), their work only involved a simple simulation case concerning a quadruple tank system. Wang. and Xie, 2014 proposed a passivity-based attitude control approach for rigid bodies that can be used in spacecraft and satellite applications. In this paper, the authors have developed a novel passivity-based state constraint and a terminal storage function specifically applicable to the nonlinear dynamics of multi-robotic systems. The developed terminal storage function ensures asymptotic closed-loop stability and superior performance. It is important to note that while nonlinear MPC (NMPC) has shown promise for a single free-floating space manipulator system (as demonstrated in the works of Papadopoulos et al., 2021; Rybus et al., 2018), current NMPC applications do not guarantee closed-loop stability if the full nonlinear dynamics terms are retained in the controller. Moreover, it is worth noting that the majority of the studies in this area have implemented NMPC for a single planar space arm with two or three degrees of freedom.

Therefore, the paper addresses the full nonlinear dynamics while guaranteeing closed-loop stability for a multi-robot system. The approach presents a significant advancement in the field of NMPC for multi-robot systems in space applications, thanks to the novel passivity-based state constraint and terminal storage function specifically designed for multi-robotic systems.

A proper choice for the proposed control algorithm's hardware and software implementation for spacecraft robotics system would be a distributed processor architecture, which provides parallel control of the spacecraft and robots, reduced computer speed demands, and inherent redundancy for enhanced reliability. The proposed architecture may include three main processors: 1) Spacecraft Attitude Determination and Control System (ADACS) and Robot Control System (RCS), 2) Command and Data Handling (C&DH), and 3) Communications Processor. The proposed control algorithm can be housed in the ADACS and RCS computer, which controls the spacecraft's attitude, orbital maneuvers, and robots' end-effector position and orientation. Sensory data from sources like star trackers, joint encoders, and vision systems can also be collected. The Computer Software Configuration Items (CSCIs) may include modules for OCA, the proposed PNMPC, Forward and Inverse Kinematics Functions, Robot Dynamics Function, Jacobian, Jacobian Rate, and more, all of which can be located in the ADACS and RCS computer.

This paper is organized as follows: Section 2 provides the equations of motions for the compounded multi-robot spacecraft system. Section 3 introduces the concept of passivity and the proposed passivity based NMPC algorithm for a free-flying multi-robot space system. In Section 4, the computer simulation results, their analysis and discussion are presented. Section 5 outlines the conclusions and suggestions for future work.

2 System dynamics

The system being studied comprises of a chaser spacecraft, a target payload, and two redundant manipulators with n degrees of freedom. An illustration of a similar system with a chaser spacecraft, two n-degree robots, and a rigid payload can be seen in Figure 1.

The set of equations that govern the motions of two robots and a chaser spacecraft can be expressed as:

$$G\ddot{\tilde{q}}(t) + D\dot{\tilde{q}}(t) = \tilde{\Gamma} \tag{1}$$

$$\ddot{\tilde{q}}^T = \left[\ddot{\tilde{R}}_c, \dot{\tilde{\omega}}_c, \ddot{\tilde{\theta}}_L, \ddot{\tilde{\theta}}_R \right] \tag{1a}$$

$$\tilde{\Gamma}^T = \left[\tilde{F}_c, \tilde{\tau}_c, \tilde{\tau}_{\theta_L}, \tilde{\tau}_{\theta_R} \right] \tag{1b}$$

where $\ddot{\tilde{q}}^T$ is a vector containing the linear and angular acceleration of the chaser spacecraft and the joint angular accelerations for the left and right arm, and $\tilde{\Gamma}^T$ is a vector containing the external control force and moments for the chaser spacecraft and the joint control torques for the left and right arm. The matrix G is a positive definite matrix that represents the mass/inertia, while D is a matrix that includes non-linear terms such as Coriolis and centrifugal terms. The linear and angular acceleration of the chaser spacecraft are represented by $\ddot{\tilde{R}}_c$, $\dot{\tilde{\omega}}_c$, respectively. $\ddot{\tilde{\theta}}_L$, $\ddot{\tilde{\theta}}_R$ are the joint angular accelerations for the left and right arm respectively, while $\tilde{F}_c, \tilde{\tau}_c$ are the external control force and moments for the chaser spacecraft. The joint control torques for the left and right arm are represented by $\tilde{\tau}_{\theta_L}$, $\tilde{\tau}_{\theta_R}$, respectively.

To enhance the mathematical expression of the dynamics equations, it is possible to represent them in the Cartesian domain for the robots and in terms of Euler angles for the spacecraft. This can be achieved by utilizing the Jacobian of the system, denoted as $\underline{J}$, which facilitates the transformations from joint space to Cartesian space for the robots and from the body-fixed angular rates to Euler rates for the spacecraft. The resulting equations can provide a more insightful understanding of the system dynamics and can be particularly useful for control and optimization purposes:

$$G^*\ddot{\tilde{\mu}}(t) + D^*\dot{\tilde{\mu}}(t) = \tilde{F} \tag{2}$$

where

$$\tilde{\mu}^T(t) = [\tilde{R}_c, \varphi, \psi, \theta, \tilde{X}_L, \tilde{X}_R] \tag{2a}$$

$\tilde{R}_c$, the position vector of the center of mass of the spacecraft, the rotational transformation from orbital to body axes is represented by a 1-3-2 Euler sequence, with angles φ, ψ and θ, respectively and $\tilde{X}_L, \tilde{X}_R$ are the pose vectors for the left and the right end-effectors, respectively. By utilizing the Pseudo-inverse Jacobian matrix J^* for the redundant system, one can calculate the equivalent mass/inertia matrix G^*, Coriolis matrix D^*, and the force vector $\tilde{F}$ in the Cartesian domain for the compounded multi-robot spacecraft system.

$$\dot{\tilde{\mu}}(t) = J\dot{\tilde{q}}(t) \tag{3}$$
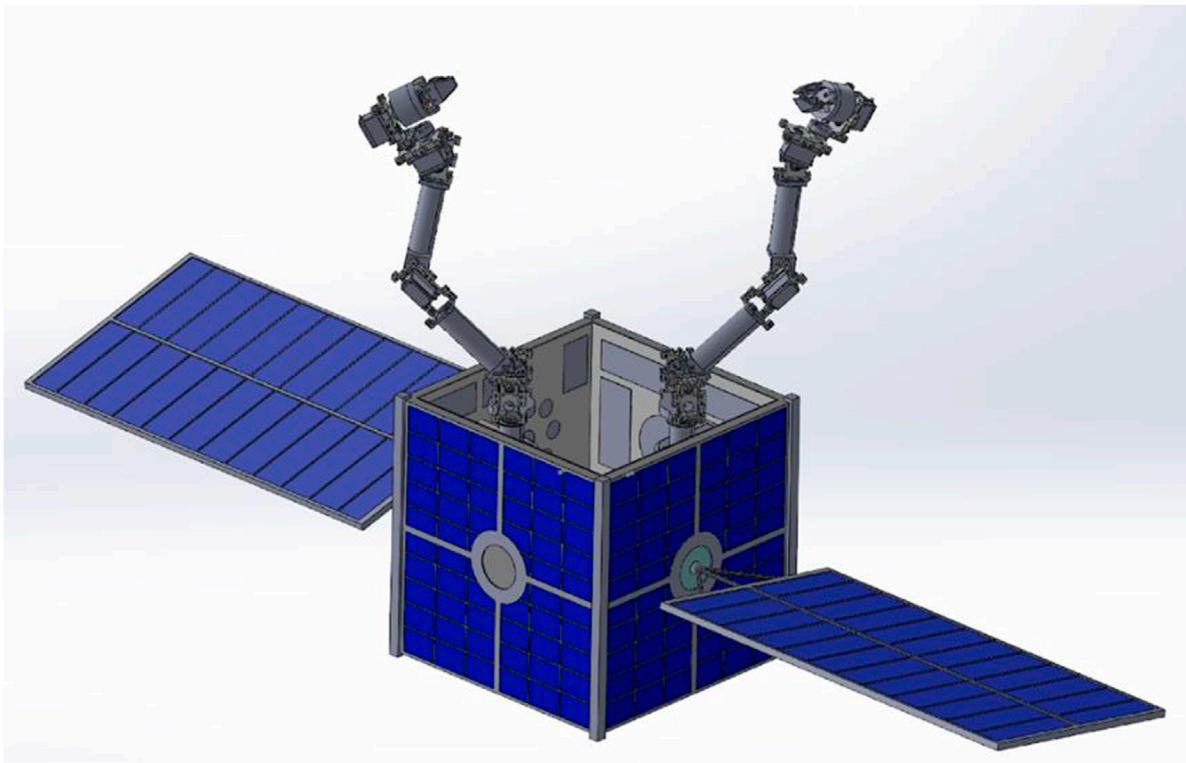
FIGURE 1
Multi-robot spacecraft system concept.

$$J^* = \left(J^T J\right)^{-1} J^T \qquad (4)$$

$$G^* = J^{*T} G J^* \qquad (5)$$

$$D^* = \left(J^{*T} D J^* - J^{*T} G J^* \dot{J} J^*\right) \qquad (6)$$

$$\widetilde{F} = J^{*T} \widetilde{\Gamma} \qquad (7)$$

It can be shown that $\dot{G}^* - 2D^*$ is a skew-symmetric matrix in the following form and this relationship can be used in developing an appropriate storage function in **Section 3**.

$$\dot{G}^* - 2D^* = -J^{*T} G J^* \dot{J} J J^* - 2 J^{*T} D J^* + J^{*T} \dot{J}^T G J^{*T} J^* \qquad (8)$$

Furthermore, the non-linear equations of the system dynamics can be written as:

$$\bar{y}^{\mathrm{T}}(t) = \left[\bar{\mu}^{\mathrm{T}}(t), \dot{\bar{\mu}}(t)\right] \qquad (9a)$$

$$\dot{\bar{y}}(t) = \bar{g}(\bar{y}) + f(\bar{y})\bar{u}(t) \qquad (9b)$$

$$\tilde{z}(t) = \bar{h}(\bar{y})$$

$$\bar{g}(\bar{y}) = \begin{bmatrix} 0 & 1 \\ 0 & -G^{*-1} D^* \end{bmatrix} \bar{y}(\mathrm{t}) \, f(\bar{y}) = \begin{bmatrix} 0 \\ G^{*-1} \end{bmatrix}$$

$$\bar{h}(\bar{y}) = C\bar{y}(\mathrm{t}) \; \bar{y}(0) = \bar{y}(t_0) \qquad (9c)$$

where the control input function is denoted by $\bar{u}(t)$, while $\tilde{z}(t)$ represents the output and C is the observation matrix The non-linear system function representing the system dynamics is denoted by $\bar{g}(\bar{y})$ while $f(\bar{y})$ is the input coefficient function, and $\bar{h}(\bar{y})$ is the non-linear function used to obtain the output vector based on the state variables.

3 The concept of passivity and passivity based NMPC

A dynamics system can be classified as passive if there is a storage function $V(\bar{y}_t)$ that meets the following condition:

$$V(\bar{y}_t) - V(\bar{y}_0) \le \int_{t_0}^t \bar{u}^T(\tau)\tilde{z}(\tau) \, d\tau \qquad (10a)$$

Furthermore, a formal characterization of a system as Input Feed-forward Output Feedback Passive (IF-OFP) is established when a storage function $V(\bar{y}_t)$ exists that satisfies the certain conditions:

$$V(\bar{y}_t) - V(\bar{y}_0) \le \int_{t_0}^t \left[\bar{u}^T(\tau)\tilde{z}(\tau) - \rho \, \tilde{z}(\tau)^T \tilde{z}(\tau) - \upsilon \, \bar{u}^T(\tau)\bar{u}(\tau)\right] d\tau \qquad (10b)$$

The positive passivity indices, denoted by ρ and υ, characterize the passivity properties of the system under consideration.

Raff et al. (2007) have established that the passivity-based nonlinear model predictive control scheme can achieve local asymptotic stability of a system, provided the system is passive and possesses a continuously differentiable storage function $V(\bar{y}_t)$. The conventional NMPC formulation for the dual arm nonlinear system can be written as:

$$min \; \frac{1}{2} \int_0^{T_p} \left[\left(\bar{y}(t) - \bar{y}_r(t)\right)^T K \left(\bar{y}(t) - \bar{y}_r(t)\right) + \bar{u}^T(t) \, W \, \bar{u}(t)\right] dt \qquad (11)$$

subject to Eq **9a**, Eq **9b**, Eq **9c** and where T_p is the prediction horizon; K and W are positive definite square weighting matrices

FIGURE 2
(A) NMPC Block diagram. **(B)** Bias computation with an independent model. **(C)** Robot geometry used in the simulations.
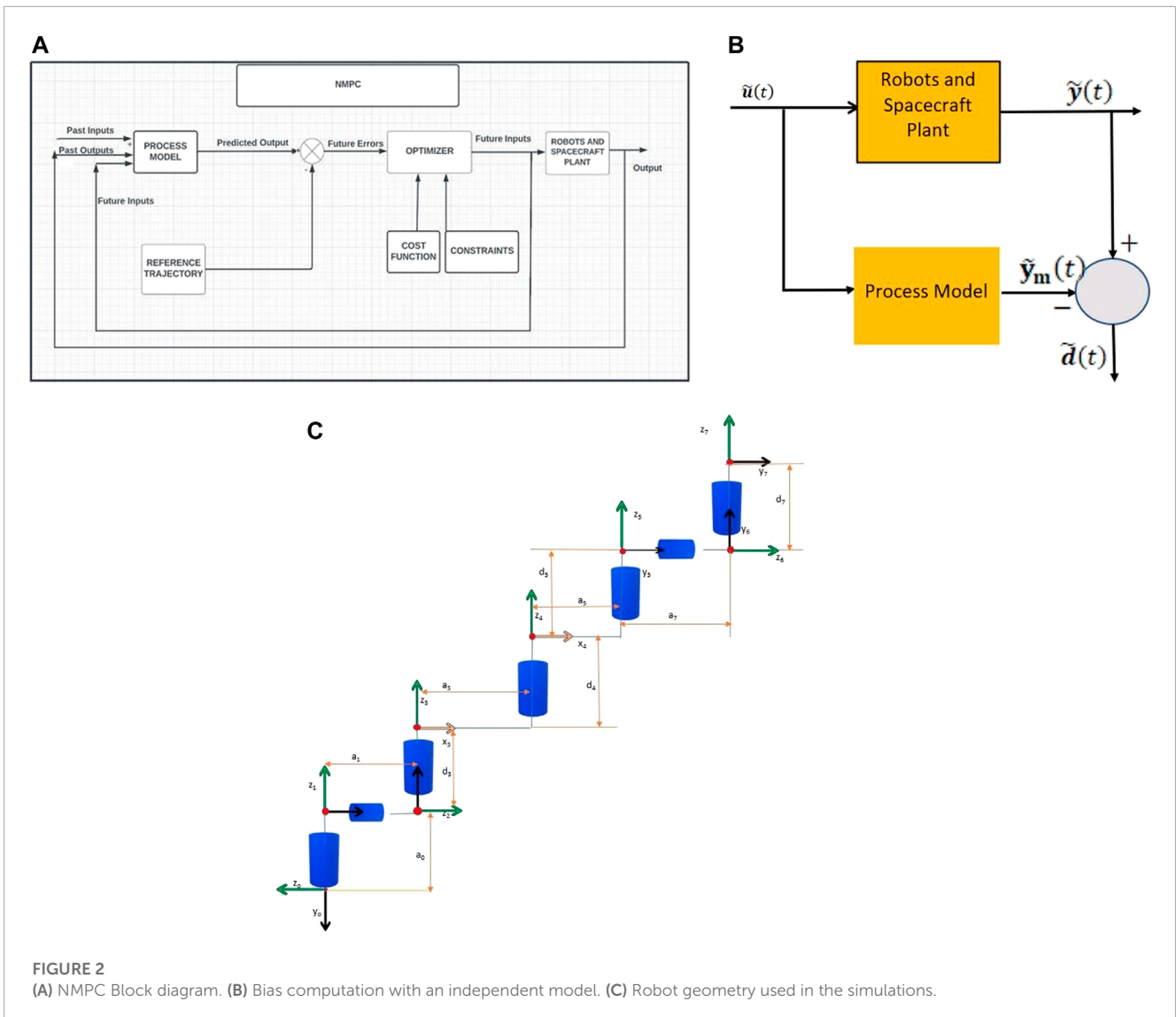
and $\tilde{\mathbf{y}}_{\mathbf{r}}(t)$ is the reference trajectory Nonlinear model predictive control (NMPC) is founded on the principle of iteratively solving the finite horizon optimal control problem in real-time to determine the optimal control input. **Figure 2A** depicts the NMPC block diagram. Schaft 2000 and Jadbabaie et al. (1999) highlighted that while NMPC can be useful, it does not always guarantee closed-loop stability.

Furthermore, the plant output $\tilde{\mathbf{y}}(t)$ might be different than the model output $\tilde{\mathbf{y}}_{\mathbf{m}}(t)$ in the presence of model parameter uncertainties. To overcome model parameter uncertainties in the process, both the model and the plant are simulated under the same control system input as shown in **Figure 2B**. The first output is the real plant output, represented as $\tilde{\mathbf{y}}(t)$ while the second output is the model output, represented as $\tilde{\mathbf{y}}_{\mathbf{m}}(t)$.

The calculations are performed within the control block diagram's "ROBOTS AND SPACECRAFT PLANT" and "PROCESS MODEL", as illustrated in **Figure 2A**. Subsequently, the bias $\tilde{d}(t)$ is determined, using the following equation:

$$\tilde{d}(t) = \tilde{y}(t) - \tilde{y}_{\mathbf{m}}(t) \tag{12}$$

The process predictions are then calculated from model predictions by adding a bias correction term to each prediction over the horizon. The details of these calculations are provided as part of Eq. **22c** when a set of discretized system equations is employed.

The Passivity-Based Nonlinear Model Predictive Control (PNMPC) scheme can be organized in the following manner through the incorporation of a terminal cost as the storage function $V\left(\tilde{\mathbf{y}}_{T_p}\right)$ to guarantee the closed loop stability (Mayne et al., 2000; Raff et al., 2007):

$$min \; V\left(\tilde{\mathbf{y}}_{T_p}\right) + \frac{1}{2}\int_0^{T_p}\left[\left(\tilde{\mathbf{y}}(t) - \tilde{\mathbf{y}}_{\mathbf{r}}(t)\right)^T \mathbf{K}\left(\tilde{\mathbf{y}}(t) - \tilde{\mathbf{y}}_{\mathbf{r}}(t)\right) + \tilde{\mathbf{u}}^T(t)\,\mathbf{W}\,\tilde{\mathbf{u}}(t)\right]dt \tag{13}$$

subject to Eq. **9a** and the following inequality constraints:

$$\frac{dV(\tilde{\mathbf{y}}_t)}{dt} \le \tilde{\mathbf{u}}^T(t)\,\tilde{\mathbf{z}}(t) \quad \text{where} \quad \tilde{\mathbf{z}}(t) = \dot{\tilde{\boldsymbol{\mu}}}(t) \tag{14}$$

$$\tilde{\mathbf{u}}^T(t)\,\tilde{\mathbf{z}}(t) \le -\rho\,\tilde{\mathbf{z}}(t)^T\tilde{\mathbf{z}}(t) - \upsilon\,\tilde{\mathbf{u}}^T(t)\tilde{\mathbf{u}}(t)\,\rho \ge \mathbf{0}, \upsilon \ge \mathbf{0} \tag{15}$$

To establish the passivity constraint, the initial step is to define the tracking error vector, which represents the deviation between the

TABLE 1 System parameters.

Hardware configuration item	Mass (kg)	Dimensions (m) (prism)
Spacecraft	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	$0.43 \times 0.1 \times 0.1$
Link 4	5	$0.64 \times 0.1 \times 0.1$
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	$1 \times 0.1 \times 0.1$

TABLE 2 Inertia parameters.

	S/C	Payload	Link1	Link2	Link3	Link4	Link5	Link6	Link7
M(kg)	40	10	1	1	3	5	3	1	3
I_{xx} (g-m\2)	1,130	830	1.7	1.7	5	8.3	5	1.7	1.7
I_{yy} (g-m\2)	1,130	830	1.7	1.7	48.7	174.8	5	1.7	1.7
I_{zz} (g-m\2)	1,660	830	1.7	1.7	48.7	174.8	5	1.7	1.7

output and the reference trajectory as follows:

$$\tilde{e}_y = \tilde{\mu}(t) - \tilde{\mu}_r(t) \tag{16}$$

In order to attain proficient tracking performance, one may contemplate utilizing the following storage function in Cartesian domain as proposed in this paper.

$$V(\tilde{y}_t) = \frac{1}{2}\left(\dot{\tilde{\mu}}^T(t)\, G^*\, \dot{\tilde{\mu}}(t) + \tilde{e}_y^T R \tilde{e}_y\right) \tag{17}$$

where $\underline{R}$ is a positive definite, symmetrical square matrix, and subsequently calculate the time derivative of $V\left(\tilde{y}_{t+T_p}\right)$ to obtain the following constraint equations.

$$\frac{d\,V(\tilde{y}_t)}{dt} = \frac{d}{dt}\left[\frac{1}{2}\left(\dot{\tilde{\mu}}^T(t)\, G^*\, \dot{\tilde{\mu}}(t) + \tilde{e}_y^T R\, \tilde{e}_y\right)\right] \tag{18}$$

Please note that $\dot{G}^* - 2D^*$ is a skew-symmetric matrix and enables the following simplification which was used in calculating the time derivative of the storage function $V(\tilde{y}_t)$ in Eq. 18.

$$\left[\dot{\tilde{\mu}}^T(t)\left(\dot{G}^* - 2D^*\right)\dot{\tilde{\mu}}(t)\right] = 0 \tag{19}$$

The time derivative of the proposed storage function can now be calculated using Eq. 18 as shown below:

$$\frac{dV(\tilde{y}_t)}{dt} = 1/2\left\{\ddot{\tilde{\mu}}(t)^T G^* \dot{\tilde{\mu}}(t) + \dot{\tilde{\mu}}(t)^T G^* \ddot{\tilde{\mu}}(t) + \dot{\tilde{\mu}}(t)^T \dot{G}^* \dot{\tilde{\mu}}(t) + \dot{e}_y^T R\, \tilde{e}_y \right.$$
$$\left. + \tilde{e}_y^T R\, \dot{e}_y\right\}$$
$$= \dot{\tilde{\mu}}(t)^T G^* \ddot{\tilde{\mu}}(t) + \dot{\tilde{\mu}}(t)^T \dot{G}^* \dot{\tilde{\mu}}(t) + \dot{e}_y^T R\, \tilde{e}_y \tag{20a}$$

Please note that both G^* and R matrices are symmetrical. Further simplification of Eq. 20a can be achieved by adding and subtracting $2D^*$ into equation and taking advantage of the skew-symmetry property represented in Eq. 19. The resulting equation is

$$\frac{dV(\tilde{y}_t)}{dt} = \dot{\tilde{\mu}}(t)^T G^* \ddot{\tilde{\mu}}(t) + \frac{1}{2}\left\{\dot{\tilde{\mu}}(t)^T\left(\dot{G}^* - 2D^* + 2D^*\right)\dot{\tilde{\mu}}(t)\right\} + \dot{e}_y^T R\, \tilde{e}_y \tag{20b}$$

By utilizing Eq. 19, it is possible to simplify Eq. 20b and which can then be written in a more concise form as follows.

$$\frac{dV(\tilde{y}_t)}{dt} = \dot{\tilde{\mu}}(t)^T G^* \ddot{\tilde{\mu}}(t) + \left\{\dot{\tilde{\mu}}(t)^T D^* \dot{\tilde{\mu}}(t)\right\} + \dot{e}_y^T R\, \tilde{e}_y \tag{20c}$$

where $\dot{e}_y = \dot{\tilde{\mu}}(t) - \dot{\tilde{\mu}}_r(t)$.

$$\frac{dV(\tilde{y}_t)}{dt} = \dot{\tilde{\mu}}(t)^T\left\{G^* \ddot{\tilde{\mu}}(t) + D^* \dot{\tilde{\mu}}(t) + R\tilde{e}_y\right\} \tag{20d}$$

By utilizing Eq. 2, the aforementioned expression can be reformulated as follows:

$$\frac{dV(\tilde{y}_t)}{dt} = \dot{\tilde{\mu}}(t)^T\left\{\tilde{F} + R\, \tilde{e}_y\right\} = \left(\tilde{F} + R\, \tilde{e}_y\right)^T \tilde{z}(t) \tag{20e}$$

Thus, the following terminal value of the predicted control input proposed in Eq. 21a satisfies the equality in Eq. 20e as well as guarantees the passivity-based inequality constraint in (Eq. 14).

$$\bar{u}(t) = \bar{u}\left(T_p\right) = \left(\tilde{F} + R\, \tilde{e}_y\right) t = T_p \tag{21a}$$
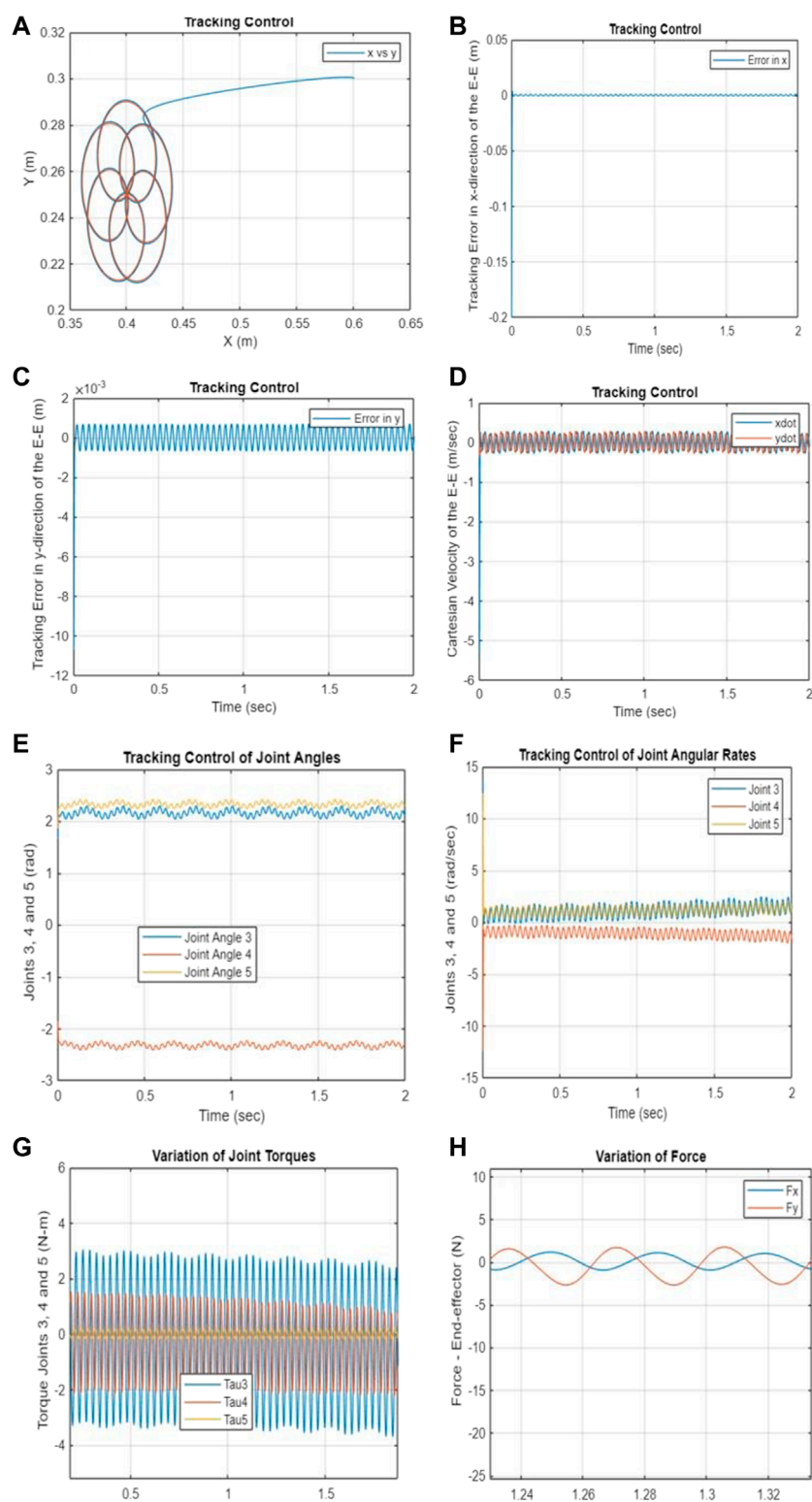
FIGURE 3
(A–H) Tracking performance–PNMPC with a storage function and passivity constraints. **(A)** Cartesian position of the end-effector. **(B)** Tracking error in the x-direction. **(C)** Tracking error in the y-direction. **(D)** Cartesian velocity of the end-effector. **(E)** Tracking control of joints 3–5. **(F)** Tracking control of angular rates for joints 3–5.

FIGURE 4
(A–H) Tracking performance–PNMPC1 without a terminal function only with passivity constraints. (A) Cartesian position of the end-effector. (B) Tracking error in the x-direction. (C) Tracking error in the y-direction. (D) Cartesian velocity of the end-effector. (E) Tracking control of joints 3–5. (F) Tracking control of angular rates for joints 3–5. (G) Variation of torques for joints 3–5. (H) Variation of end-effector contact force.

FIGURE 5
(A–B) The impact of the terminal/storage function in the NMPC performance index. **(A)** Comparison of tracking performances in the x-direction between PNMPC and PNMPC1. **(B)** Comparison of tracking performances in the y-direction between PNMPC and PNMPC1.

Hence, it is hereby proven that the proposed terminal value of predicted control input and the storage function satisfy the constraint provided in Eq. **14** as follows:

$$\frac{dV\left(\bar{\boldsymbol{y}}_{T_p}\right)}{dt} = \left(\widetilde{\boldsymbol{F}} + \boldsymbol{R}\,\tilde{\boldsymbol{e}}_{\mathbf{y}}\right)^T \dot{\tilde{\boldsymbol{\mu}}}(t) = \tilde{\boldsymbol{u}}^T(t)\dot{\tilde{\boldsymbol{\mu}}}(t) \leq \tilde{\boldsymbol{u}}^T(t)\,\tilde{\boldsymbol{z}}(t) \quad t = T_p$$
(21b)

where the value of $\widetilde{\boldsymbol{F}}$ is calculated by utilizing the model-based computed torque/force approach at $t = T_p$. It is part of the terminal value of the predicted input and not measured from the plant output as illustrated in **Figure 2A**. The first inequality constraint provided in (Eq. **14**) is now satisfied by the proposed storage function $V(\bar{\boldsymbol{y}}_t)$ (Eq. **17**) and the proposed control input (Eq. **21a**). In order to enforce the closed loop stability, the other passivity constraints provided in Eq. **15** are implemented as part of the NMPC model. i.e.

$$\tilde{\boldsymbol{u}}^T(t)\,\dot{\tilde{\boldsymbol{\mu}}}(t) \leq -\rho\dot{\tilde{\boldsymbol{\mu}}}^T(t)\,\dot{\tilde{\boldsymbol{\mu}}}(t) - \boldsymbol{v}\,\tilde{\boldsymbol{u}}^T(t)\,\tilde{\boldsymbol{u}}(t)$$
$$\rho \geq 0, \boldsymbol{v} \geq 0$$
(21c)

The following prediction equations are employed in conjunction with the system discrete state-space model at each sampling instant k_t (Kalaycioglu, S and de Ruiter A. 2023). These equations are the discretized version of Eq. **9b**. and written as follows:

$$\Delta\bar{\boldsymbol{y}}(k_t+j+1) = \widehat{\underline{\boldsymbol{A}}}(\widehat{\boldsymbol{g}}(k_t+j))\Delta\bar{\boldsymbol{y}}(k_t+j) + \widehat{\underline{\boldsymbol{B}}}(\widehat{\boldsymbol{g}}(k_t+j))\,\Delta\tilde{\boldsymbol{u}}(k_t+j)$$

$$\Delta\bar{\boldsymbol{y}}(k_t+j) = \bar{\boldsymbol{y}}(k_t+j) - \tilde{\mathbf{y}}_{\mathbf{r}}(k_t+j)$$

$$\Delta\tilde{\boldsymbol{u}}(k_t+j) = \tilde{\boldsymbol{u}}(k_t+j) - \tilde{\boldsymbol{u}}(k_t+N_p)$$

$$j = 1\ldots N_p$$
(22a)

Where $\tilde{\boldsymbol{u}}(k_t+N_p)$ is the terminal value of the predicted input at t = T_p. (i.e., $\widetilde{\boldsymbol{F}} + \boldsymbol{R}\,\tilde{\boldsymbol{e}}_{\mathbf{y}}$). These prediction equations are then substituted into the discretized version of the cost function (Eq. **13**) and the optimum predicted control inputs $\Delta\tilde{\boldsymbol{u}}(k_t+j)$ are obtained as follows:

$$\Delta\tilde{\boldsymbol{u}}(k+j) = -\left(\widehat{\underline{\boldsymbol{B}}}^T\,\underline{\boldsymbol{K}}\quad\widehat{\underline{\boldsymbol{B}}} + \underline{\boldsymbol{W}}\right)^{-1}\widehat{\underline{\boldsymbol{A}}}^T\,\underline{\boldsymbol{K}}\,\widehat{\underline{\boldsymbol{B}}}\left(\bar{\boldsymbol{y}}(k_t+j) - \tilde{\mathbf{y}}_{\mathbf{r}}(k_t+j)\right)$$

$$\boldsymbol{Q} = \left(\widehat{\underline{\boldsymbol{B}}}^T\,\underline{\boldsymbol{K}}\widehat{\underline{\boldsymbol{B}}} + \underline{\boldsymbol{W}}\right)^{-1}\widehat{\underline{\boldsymbol{A}}}^T\,\underline{\boldsymbol{K}}\,\widehat{\underline{\boldsymbol{B}}}$$

$$\Delta\tilde{\boldsymbol{u}}(k+j) = -\boldsymbol{Q}\left(\bar{\boldsymbol{y}}(k_t+j) - \tilde{\mathbf{y}}_{\mathbf{r}}(k_t+j)\right)$$

$$\tilde{\boldsymbol{u}}(k+j) = -\boldsymbol{Q}(\bar{\boldsymbol{y}}(k_t+j) - \tilde{\mathbf{y}}_{\mathbf{r}}(k_t+j)) + \tilde{\boldsymbol{u}}(k_t+N_p)$$
(22b)

Therefore, the passivity based NMPC predictive control yields a state feedback mechanism $\{-\boldsymbol{Q}\left(\bar{\boldsymbol{y}}(k_t+j) - \tilde{\mathbf{y}}_{\mathbf{r}}(k_t+j)\right)\}$ augmented with a feedforward component $\{\tilde{\boldsymbol{u}}(k_t+N_p)\}$ predicated on the terminal value of the inputs (i.e. $\widetilde{\boldsymbol{F}} + \boldsymbol{R}\,\tilde{\boldsymbol{e}}_{\mathbf{y}}$). The incorporation of the terminal cost as the storage function $V\left(\bar{\boldsymbol{y}}_{T_p}\right)$ guarantees the closed loop stability. The mathematical formulations present a novel approach to NMPC for multi-robot systems in space applications by incorporating the full non-linear system dynamics equations in the Cartesian domain using the concept of passivity.

Please also note that the process predictions are calculated from model predictions by adding a bias correction term to each prediction over the horizon $j = 1 \ldots Np$ to include model parameter uncertainties as part of Eq. **22a** when using a discretized set of system equations at the sampling time of k_t. where

$$\bar{\boldsymbol{y}}(k_t+j) = \tilde{\mathbf{y}}_{\mathbf{m}}(k_t+j) + \tilde{\boldsymbol{d}}(k_t)$$
(22c)

The proposed PNMPC scheme utilizes a novel passivity-based state constraint and a terminal storage function to ensure closed-loop stability and superior performance. The approach demonstrates that passivity-based concepts and NMPC can be combined into
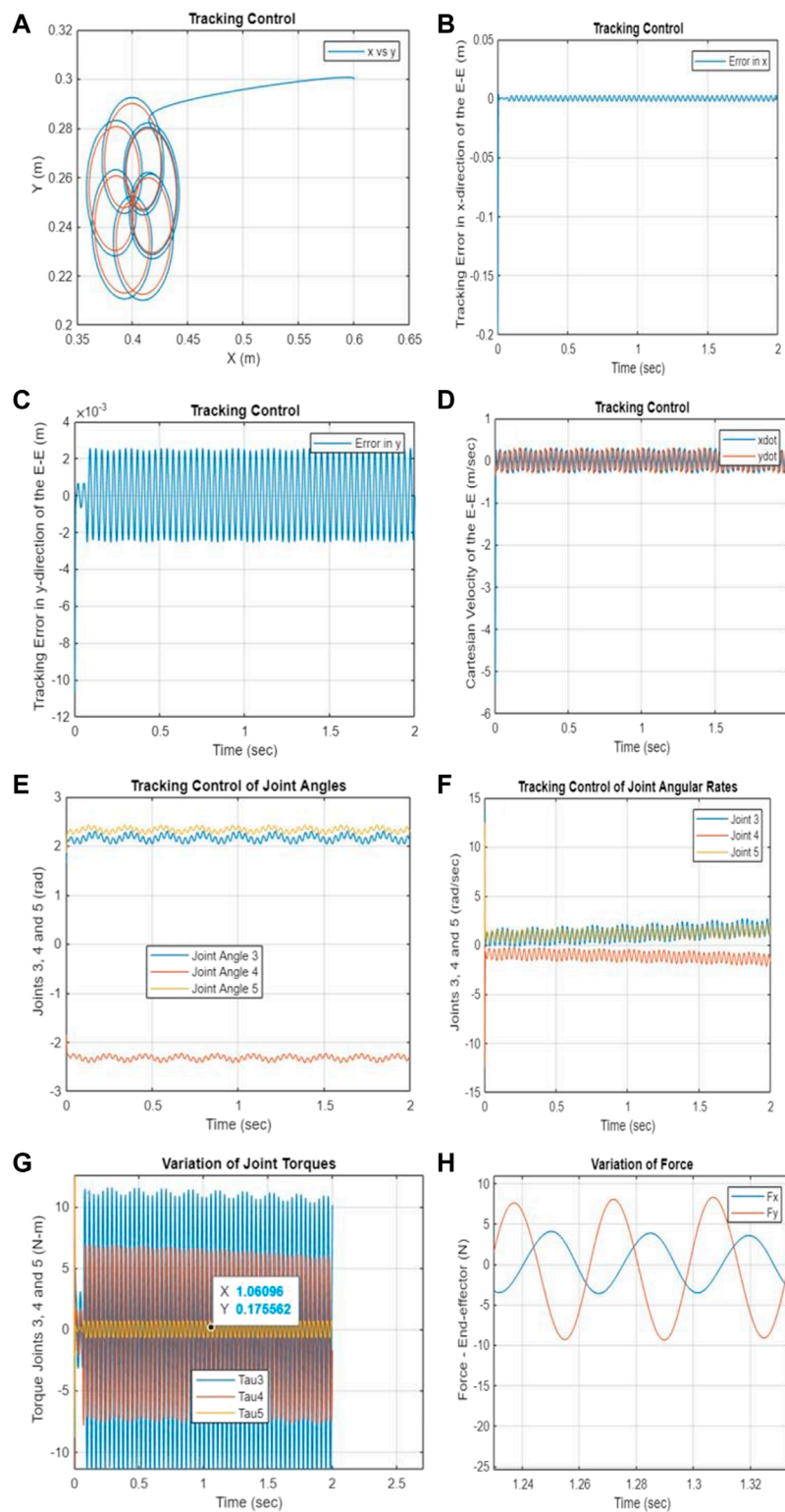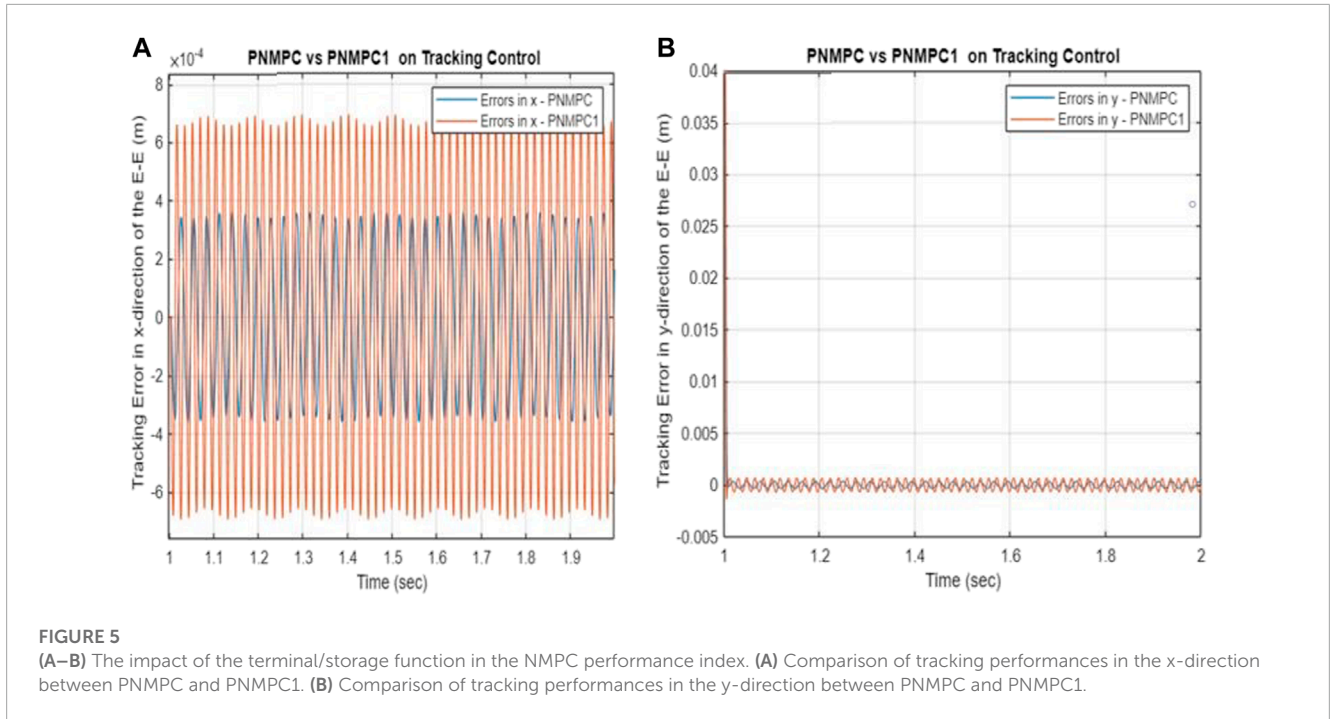
FIGURE 6
(A–J) Tracking performance–PNMPC without a terminal function only with passivity constraints. **(A)** Cartesian position of the end-effector. **(B)** Tracking error in the x-direction. **(C)** Tracking error in the y-direction. **(D)** Cartesian velocity of the end-effector. **(E)** Tracking control of joints 3−5. **(F)** Tracking control of angular rates for joints 3−5. **(G)** Variation of torques for joints 3−5. **(H)** Variation of end-effector contact force. **(I)** Tracking performance in the y-direction− PNMPC without a terminal function only with passivity constraints. **(J)** Tracking performance in the x-direction− PNMPC without a terminal function only with passivity constraints.

a single scheme that maintains the advantages of each, including closed-loop stability through passivity and good performance through one-line optimization in NMPC. This approach also provides an alternative method to the control Lyapunov function for controlling non-linear multi-robot space systems and applications, as stability and passivity exhibit a close relationship.

4 Result and discussion

This section presents and examines the outcomes of computer simulations. Three sets of simulation results are presented, which correspond to 1) the proposed Passivity-Based Nonlinear Model Predictive Control (PNMPC) approach with a storage terminal function and the passivity constraints; 2) PNMPC scheme operating without utilizing a storage terminal function, instead relying solely on passivity constraints and finally 3) a conventional NMPC that lacks the passivity conditions and the storage function in its cost function. The simulation results of these three sets are compared, analyzed, and discussed in detail.

4.1 Simulation results—Passivity-based nonlinear model predictive control (PNMPC)

In this section, the proposed passivity-based nonlinear model predictive scheme is applied to control a multi-robot spacecraft. In the simulation, the reference trajectory is characterized by a high level of intricacy. This intricate and complex trajectory serves as a demonstration of the effectiveness of the proposed PNMPC algorithm in applications that demand precise pursuit of intricate and rapid trajectories.

The end-effector is required to track the following trajectory on a plane where the payload is located:
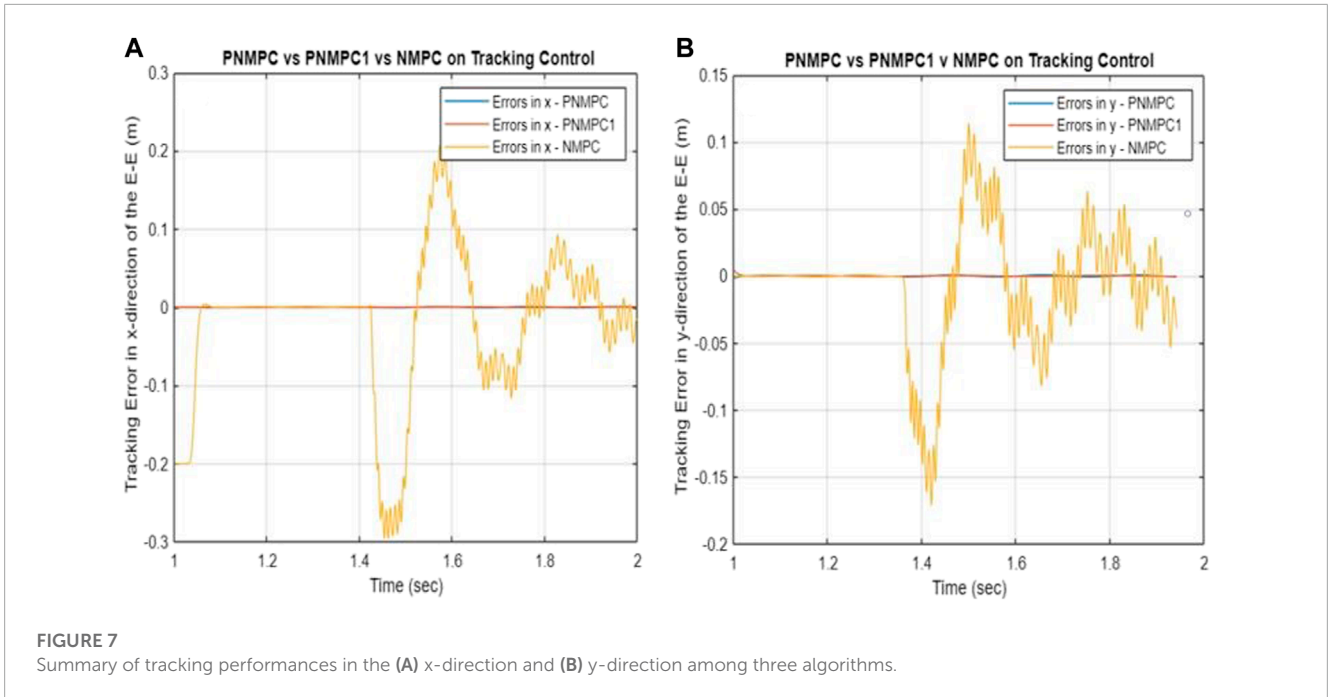
$$x_r = x_0 + a\sin(\omega_n t) + a\sin(\omega_p t)$$
$$y_r = y_0 + a\sin(\omega_n t + \phi_0) + a\sin(\omega_p t + \phi_0)$$
$$z_r = z_o \tag{23}$$

where $[x_0, y_0, z_o, \omega_n, \omega_p, \phi_0, a] = [0.4, 0.25, 0.5, 2, 12, 1.6, 0.02]$ also the initial conditions for the end-effector of the left robot are given as: $[x, y, z, \dot{x}_d, \dot{y}_d, \dot{z}_d] = [0.6, 0.3, 0.5, 0, 0, 0]$. The reference trajectory represents co-centric circles. **Tables 1**, **2** provides the system parameters used in the computer simulations for the two robots and the spacecrafts. **Figure 2C** illustrates the robot geometry used in the simulation.

4.2 Part 1: The proposed PNMPC with the storage terminal function and the passivity constraints

In this section, the following cost function provided in Eq. **13**, subject to Eq. **14** and Eq. **15** is minimized.

$$min\ V\left(\tilde{\mathbf{y}}_{T_p}\right) + \frac{1}{2}\int_0^{T_p}\left[(\tilde{\mathbf{y}}(t) - \tilde{\mathbf{y}}_\mathbf{r}(t))^T K(\tilde{\mathbf{y}}(t) - \tilde{\mathbf{y}}_\mathbf{r}(t)) + \tilde{\mathbf{u}}^T(t)\ W\tilde{\mathbf{u}}(t)\right]dt$$

FIGURE 7
Summary of tracking performances in the (A) x-direction and (B) y-direction among three algorithms.

Figures 3A–H exhibit the simulation results, which indicate that the developed PNMPC algorithm is remarkably adept at tracking the intricate trajectory. **Figure 3A** displays the Cartesian position of the end-effector, while **Figures 3B,C** illustrate the error in Cartesian displacement of the end-effector. **Figure 3D** illustrates the time variation of the Cartesian velocity of the end-effector. Moreover, **Figures 3E–H** demonstrate the variations in joint angles over time, angular rates, joint torque, and the equivalent force values, correspondingly.

4.3 Part 2: PNMPC1 scheme without the storage terminal function

The following PNMPC1 scheme operates without utilizing a storage terminal function, instead relying solely on passivity constraints. The simulation results presented in this section, the storage function $V(\tilde{y}_t)$ is removed from the NMPC performance index as a terminal function. **Figures 4A–H** illustrate the effect of the storage function $V(\tilde{y}_t)$ on the tracking performance and stability. The same reference trajectory and the initial conditions are used to compare the simulation results.

By satisfying both Eq. **14** and Eq. **15**, the simulation results are generated through the minimization of the following cost function subject to negative feedback control law.

$$min \frac{1}{2}\int_0^{T_p}\Big[\big(\tilde{\mathbf{y}}(t)-\tilde{\mathbf{y}}_{\mathbf{r}}(t)\big)^T\mathbf{K}\big(\tilde{\mathbf{y}}(t)-\tilde{\mathbf{y}}_{\mathbf{r}}(t)\big)+\tilde{\boldsymbol{u}}^T(t)\mathbf{W}\tilde{\boldsymbol{u}}(t)\Big]dt$$

$$\tilde{\boldsymbol{u}}(t) = -\mathbf{Q}\,\tilde{e}_{\mathbf{y}} \qquad (24)$$

Based on the results shown in **Figures 3A–G** and **Figures 4A–G**, it is evident that the proposed Passivity-based Nonlinear Model Predictive Control (PNMPC) outperforms the PNMPC system that lacks a terminal function (PNMPC1), while still adhering to

passivity constraints. While both systems demonstrate stability, the PNMPC approach with the proposed terminal storage function provides notably superior tracking performance. **Figures 5A,B** display the tracking errors for both algorithms, highlighting the influence of the terminal function in the cost function (or performance index) of the developed NMPC scheme.

4.4 Part 3: NMPC scheme without the storage terminal function and passivity constraints

In this section, the following cost function provided in Eq. **24** without being subject to Eq. **14** and Eq. **15** is minimized subject to negative feedback control law. **Figures 6A–J** demonstrate the impact of the absence of both the storage function and passivity constraints on tracking performance and stability. The simulation was conducted using identical reference trajectory and initial conditions to ensure a fair comparison of the results.

The presented **Figures 6A–J** indicate that the Nonlinear Model Predictive Control (NMPC) algorithm is unable to track the reference trajectory when both the storage function and passivity constraints are absent. The tracking errors are deemed unacceptable, and the results are catastrophic.

4.5 Part 4: Summary comparisons of tracking errors for three sets of simulations

Based on the simulation results illustrated in **Figures 7A,B**, it is evident that the PNMPC system proposed in this study surpasses both the PNMPC system that lacks a terminal function (PNMPC1), while still adhering to passivity constraints, and the regular NMPC scheme (NMPC) that lacks a storage terminal function and is not subject to passivity constraints. While the passivity constraints

contribute to stability, the proposed PNMPC approach with the terminal storage function provides notably superior tracking performance. **Figures 7A,B** display the tracking errors for all three algorithms, emphasizing the impact of the terminal function in the cost function (or performance index) and the presence of passivity constraints in the Nonlinear Model Predictive Control (NMPC) schemes.

5 Conclusion and future work

This paper introduced a nonlinear model predictive control method predicated on the concept of passivity for multi-robot systems in space applications. The study and the simulation results demonstrated that utilizing specific passivity-based state constraints, along with a terminal storage function, guaranteed closed-loop stability and resulted in superior tracking performance in simulation results. The proposed approach integrated passivity, optimal control, and nonlinear model predictive control to cater to spacecraft-mounted multi-robot systems, specifically in the context of space applications. Due to the close relationship between stability and passivity, the proposed passivity-based nonlinear model predictive control scheme provided an alternative approach to the control Lyapunov function-based method for redundant nonlinear space robotics systems. The approach was applied to a free-flying spacecraft-based multi-robot system. This paper demonstrated that the benefits of passivity-based concepts and NMPC can be combined into a single NMPC scheme that maintains the advantages of each, including closed-loop stability through passivity and good performance through one-line optimization in NMPC. However, the current study has identified several limitations of the developed PNMPC scheme, which we intend to address in future research. One potential drawback is that inaccuracies in the modeling process could adversely affect its performance. Another limitation is that the PNMPC requires offline tuning of its parameters. Furthermore, the lack of experimental testing in the current study highlights a need for further research to validate the simulation results.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

Author contributions

The study conception and design were contributed by all authors. SK performed the initial material preparation and analysis, and the first draft of the manuscript was written by SK and AR. All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Aghili, F. (2020). Optimal trajectories and robot control for detumbling a non-cooperative satellite. *J. Guid. Cont. Dyn.* 43, 981–988. doi:10.2514/1.g004758

Biesbroek, R., Innocenti, L., Wolahan, A., and Serrano, S. M. (2017). "e. deorbit ESA's active debris removal mission," in *Proceedings of the 7th European conference on space debris* (Darmstadt, Germany: ESA Space Debris Office).

Billot, C., Ferraris, S., Rembala, R., Cacciatore, F., Tomassini, A., and Biesbroek, R. (2014). "E. Deorbit: Feasibility study for an active debris removal," in *3rd European workshop on space debris modeling and remediation* (Paris, France: Centre National d'Etudes Spatiales).

Chatterjee, J. (2014). Legal issues relating to unauthorised space debris remediation. *65th Int. Astronaut. Congr.* 22, 1–20.

Christofides, P., Scattolini, D., Muñoz de la Peña, D., and Liu, J. (2013). Distributed model predictive control: A tutorial review and future research directions. *Comput. Chem. Eng.* 51, 21–41.

Englert, T., Volz, a., Mesmer, F., Rhein, S., and Graichen, K. (2019). A software framework for embedded nonlinear model predictive control using a gradient-based augmented Lagrangian approach (GRAMPC). *Optim. Eng.* 20 (3), 769–809. doi:10.1007/s11081-018-9417-2

Fin, M., "Implementation of linear model predictive control –tutorial", doi:10.48550/arXiv.2109.119862021).

Forshaw, J., Aglietti, G., Salmon, T., Retat, I., Burgess, C., Chabot, T., et al. (2017). "The remove debris mission: Preparing for an international space station launch," in 7th European Conference on Space Debris, Darmstadt, Germany, 18–21 April.

Hewing, L., Wabersich, K. P., Menner, M., and Zeilinger, M. N. (2020). Learning-based model predictive control: Toward safe learning in control. *Annu. Rev. Control Robot. Auton. Syst.* 3, 269–296. doi:10.1146/annurev-control-090419-075625

Jadbabaie, A., Yu, J., and Hauser, J. (1999). "Stabilizing receding horizon control of nonlinear systems: A control Lyapunov function approach," in Proceedings of the American Control Conference, San Diego, CA, USA, 02-04 June 1999, 1535–1539.

Kalaycioglu, S., and de Ruiter, A. (2023). Nonlinear model predictive control of rover robotics system. *Adv. Sci. Technol. Eng. Syst. J.* 4 (2).

Krebs, D. (2022). *OSAM 2 (archinaut 1)"*. Hessen, Germany: Gunter's Space Page.

Mayne, D. Q., Rawlings, C. V., Rao, C. V., and Scokaert, P. O. M. (2000). Constrained model predictive control: Stability and optimality. *Automatica* 36, 789–814. doi:10.1016/s0005-1098(99)00214-9

Morato, M. M., Normey-Rico, J. E., and Sename, O. (2020). Model predictive control design for linear parameter varying systems: A survey. *Annu. Rev. Control* 49, 64–80. doi:10.1016/j.arcontrol.2020.04.016

Nanos, K., and Papadopoulos, E. G. (2017). On the dynamics and control of free-floating space manipulator systems in the presence of angular momentum. *Front. Robot. AI* 4, 26. doi:10.3389/frobt.2017.00026

Nanos, K., and Papadopoulos, E. (2011). On the use of free-floating space robots in the presence of angular momentum. *Intell. Serv. Robot.* 4, 3–15. doi:10.1007/s11370-010-0083-2

NASA's Robotic OSAM-1 Mission (2022). NASA's robotic OSAM-1 mission completes its critical design review. Available at: https://www.nasa.gov/feature/goddard/2022/nasa-s-robotic-osam-1-mission-completes-its-critical-design-review.

Orbital Hub DLR Vision (2022). DLR institute of space systems analysis space segment, bremen. Available at: https://www.researchgate.net/publication/328495824_Orbital-Hub_DLR.

Papadopoulos, E., Aghili, F., Ma, O., and Lampariello, R. (2021). Robotic manipulation and capture in space: A survey. *Front. Robot. AI* 8, 686723. doi:10.3389/frobt.2021.686723

Psomiadis, E., and Papadopoulos, E. (2022). "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, 28 June 2022 - 01 July 2022.

Quirynen, R., Vukov, M., Zanon, M., and Diehl, M. (2015). Autogenerating microsecond solvers for nonlinear MPC: A tutorial using acado integrators. *Optim. Control Appl. Methods* 36 (5), 685–704. doi:10.1002/oca.2152

RaffEbenbauer, T. C., and Allgower, F. (2007). "Nonlinear model predictive ¨ control: A passivity-based approach," in *Assessment and future directions of nonlinear model predictive control, ser. Lecture notes in control and information sciences* (Berlin, Heidelberg: Springer), 358, 151–162.

Raisi, M., Noohian, A., and Fallah, S. (2022). A fault-tolerant and robust controller using model predictive path integral control for free-flying space robots. *Front. Robot. AI* 9, 1027918. doi:10.3389/frobt.2022.1027918

Rathai, K. M. M. (2020). "Synthesis and real-time implementation of parameterized NMPC schemes for automotive semi-active suspension systems,". Ph.D. thesis in *Grenoble INP* (Grenoble, France: Communaut´e Universit´e Grenoble Alpes).

Rawlings, J. B. (2000). Tutorial overview of model predictive control. *IEEE Control Syst.* 20 (3), 38–52.

Rybus, R., Seweryn, K., and Sasiadek, J.,(2018) "Application of predictive control for manipulator mounted on a satellite". *Archives Control Sci.*, 28(1), pp. 105–118.

Schaft, A. (2000). *L2-Gain and passivity techniques in nonlinear control*. London: Springer.

Seweryn, K., Basmadji, F. L., and Rybus, T. (2022). Space robot performance during tangent capture of an uncontrolled target satellite. *J. Astronaut. Sci.* 69, 1017–1047. doi:10.1007/s40295-022-00330-2

Shi, Y., and Zhang, K. (2021). Advanced model predictive control framework for autonomous intelligent mechatronic systems: A tutorial overview and perspectives. *Annu. Rev. Control* 52, 170–196. doi:10.1016/j.arcontrol.2021.10.008

Shuyou, Y., Marcus, R., Hong, C., and Frank, A. (2014). Inherent robustness properties of quasi-infinite horizon nonlinear model predictive control. *Automatica* 50, 2269–2280. doi:10.1016/j.automatica.2014.07.014

Shyam, R. A., Hao, Z., Montanaro, U., Dixit, S., Rathinam, A., Gao, Y., et al. (2021). Autonomous robots for space: Trajectory learning and adaptation using imitation. *Front. Robot. AI* 8, 638849. doi:10.3389/frobt.2021.638849

Space Foundation White Paper (2020). U.S. Satellite servicing policy – an overview", Washington strategic operations. Available at: https://www.spacefoundation.org/white_papers/u-s-satellite-servicing.

Vukov, M., Gros, S., Horn, G., Frison, G., Geebelen, K., Jørgensen, J. B., et al. (2015). Real-time nonlinear MPC and MHE for a large-scale mechatronic application. *Control Eng. Pract.* 45, 64–78. doi:10.1016/j.conengprac.2015.08.012

Wang, H., and Xie, Y. (2014). Passivity based attitude control of rigid bodies. *Asian J. Control* 16, 802–817. doi:10.1002/asjc.616

Wang, M., Luo, J., and Walter, U. (2016). A non-linear model predictive controller with obstacle avoidance for a space robot. *Adv. Space Res.* 57 (8), 1737–1746. doi:10.1016/j.asr.2015.06.012

Yoshida, K. (2003). Engineering test satellite VII flight experiments for space robot dynamics and control: Theories on laboratory test beds ten years ago, now in orbit. *Int. J. Robot. Res.* 22, 321–335. doi:10.1177/0278364903022005003

Zhang, F., and Huang, P. (2016). Releasing dynamics and stability control of maneuverable tethered space net. *IEEE- ASME. Trans. Mechatron.* 22, 983–993. doi:10.1109/tmech.2016.2628052

Zhao, P., Liu, J., and Wu, C. (2020). Survey on research and development of on-orbit active debris removal methods. *Sci. China Technol. Sci.* 63, 2188–2210. doi:10.1007/s11431-020-1661-7