POLITECNICO DI TORINO
Repository ISTITUZIONALE

Modeling and Real-Time Simulation of Spacecraft Berthing: Orbital Dynamics and Lumped Parameter Systems in Proximity Operations

Original
Modeling and Real-Time Simulation of Spacecraft Berthing: Orbital Dynamics and Lumped Parameter Systems in Proximity Operations / Ferrauto, M., Scantamburlo, E., Sorli, D., Mauro, S., Romano, M.. - (2025), pp. 1048-1059. (76th International Astronautical Congress) [10.52202/083088-0116].

Availability:
This version is available at: 11583/3011718 since: 2026-06-05T08:00:04Z

Publisher:
International Astronautical Federation (IAF)

Published
DOI:10.52202/083088-0116

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
IAC/IAF postprint versione editoriale/Version of Record

(Article begins on next page)

19 June 2026

76th International Astronautical Congress (IAC 2025), Sydney, Australia, 29 Sep-3 Oct 2025.
Copyright ©2025 by the International Astronautical Federation (IAF). All rights reserved.

IAC-25 ,C2, IPB, 3 ,x95694

Modelling and real-time simulation of spacecraft berthing: orbital dynamics and lumped parameters systems in proximity operations

Martina Ferrauto[a]*, Erica Scantamburlo[a] , Davide Sorli [a], Stefano Mauro[a], Marcello Romano[b]

[a] *Dipartimento di Ingegneria Meccanica e Aerospaziale (DIMEAS), Politecnico di Torino, Corso Duca degli Abruzzi, 24 | 10129, Italy*
[b] Institute of Astrodynamics, Technical University of Munich (TUM)
* Corresponding Author

Abstract

The research, conducted within the Multi-Satellite Proximity Operations for Rendezvous and Docking Missions in Earth and Moon Orbits (MUSAPOEM) project funded by the Italian Space Agency (ASI), presents advanced mathematical models and algorithms for autonomous guidance, navigation, and control (GNC) in proximity operations in both terrestrial and lunar environments. Future collaborative missions such as lunar sample return, in-orbit assembly, and satellite servicing require precise modelling of the dynamics involved in berthing and docking manoeuvres. A hybrid modelling approach is proposed for berthing operations between a chaser and a target spacecraft, selecting the most suitable model depending on their relative distance. For distances above a critical threshold, the relative motion is described using orbital dynamics; for closer ranges, a detailed robotic system model is adopted to simulate the extension and positioning of the manipulator arm towards the docking point. Upon contact, a lumped parameter model (LPM) simulates the interaction phase, ensuring real-time computation where simulation time is always greater than computation time. This multi-phase modelling strategy enables complete mission simulation, including berthing and unberthing manoeuvres, with high fidelity representation of contact forces. These forces, derived from multibody simulations, include both normal and frictional components, where static and dynamic friction significantly influence docking stability. A Particle Swarm Optimization (PSO) algorithm is employed to tune the parameters of both the lumped parameter and multibody models, achieving accuracy and efficiency in real-time scenarios. The results demonstrate the framework's ability to simulate the full mission timeline while dynamically switching between modelling approaches according to mission phase requirements. This methodology contributes to robust, efficient, and autonomous GNC solutions, supporting future cooperative space missions in Earth and lunar orbits.

Keywords: proximity operations, spacecraft berthing, autonomous GNC, lumped parameter model, multibody simulation, lunar missions.

1. Introduction

The development of innovative operational concepts for proximity operations (RPO), such as rendezvous, docking, and berthing between satellites orbiting a celestial body, has become a key focus of research. These capabilities are essential for future cooperative missions, including lunar sample return, in-orbit assembly (IOA), and on-orbit servicing (OOS) [1]. The growing interest of space agencies in RPO is driven by the need to implement autonomous and automated guidance, navigation, and control (GNC) algorithms capable of extending the operational lifetime of artificial satellites [2, 3]. Notable pioneering OOS missions include Orbital Express (2007) and the Mission Extension Vehicle (launched in 2019) [4]. Future missions such as NASA's OSAM-1 [18] continue to advance these capabilities, utilizing sophisticated robotic manipulation systems including specialized appliques designed for in-space servicing vehicles [19].

For RPOs involving a controlled chaser satellite approaching a target satellite, the interaction is typically divided into two distinct phases based on their relative distance: the orbital phase (OP) and the contact phase (CP). During the OP, the satellites' motion is governed by classical orbital mechanics [6], in which no mutual interaction between the satellites occurs. The relative motion of two satellites is often described using linearized equations, such as the Hill-Clohessy-Wiltshire equations for circular orbits of the target satellite or the Tschauner-Hempel equations for elliptical orbits of the target satellite [5, 7]. The linearized equations of motion are derived under the assumption that the distance between the satellites is negligible with respect to the attractor distance. The transition from the OP to the CP occurs when the relative distance between the chaser, also named servicer, and the target falls below a specific threshold, ϵ. During the CP, the satellites are modelled as a single multibody system, whose dynamics is driven by the mechanical interactions at the interface between the chaser's end-effector and the target [8]. Accurate modeling of these interactions is essential to ensure mission success and prevent structural damage [9]. This

https://doi.org/10.52202/083088-0116

Copyright ©2025 by the International Astronautical Federation (IAF). All rights reserved.

mechanical connection can be established through docking or berthing [2]. Berthing, in particular, requires a robotic manipulator, like the well-known Canadarm [10], to capture and align the two satellites at a designated interface. The effectiveness of robotic arms has been demonstrated in multiple missions, including the Hubble Space Telescope servicing and the assembly of the International Space Station [11].

This work is conducted within the MUSAPOEM project, funded by ASI. The main objective is to develop a hybrid modeling framework that can seamlessly switch between orbital dynamics, robotic multibody simulation, or a lumped parameter model (LPM), depending on the relative distance between two or more satellites. This approach ensures a balance between high-fidelity physics representation and computational efficiency for real-time simulation, which is critical for GNC systems and hardware-in-the-loop testing.

In our previous works [13, 14, 15], we detailed the overall simulation architecture for proximity operations. That study described the transition logic from the OP—governed by relative orbital mechanics—to the CP—described through multibody or lumped parameter models. It also introduced a distance threshold criterion (ϵ=3.603 m) defining the switch between phases, as well as control strategies for berthing and unberthing maneuvers.

The present paper focuses specifically on the contact phase modeling, where the LPM is employed to capture contact forces [15], friction effects, and structural compliance, while maintaining real-time capability. The novelty of this study, compared to previous works, lies in three key aspects:

- The integration of an LPM for CP simulation, which is calibrated directly from high-fidelity multibody simulations [14].
- The use of Particle Swarm Optimization (PSO) to calibrate stiffness and damping parameters.
- The performance assessment under different approach velocities, highlighting the trade-off between accuracy and computation time.

A key feature of the proposed methodology is that the stiffness and damping parameters of the LPM are determined directly from multibody simulations. By extracting these quantities from a detailed model, the LPM can replicate the essential dynamic behavior with reduced complexity, while remaining consistent with parameters typically used in robotic arm designs for on-orbit servicing and debris removal [16, 17, 20]. This approach is particularly relevant for systems like the inflatable robotic arm POPUP [12], designed for debris capture, which presents unique challenges in modeling its structural compliance and contact dynamics.

2. LPM conceptual layout

The LPM is developed to replace the multibody simulation during the contact phase, preserving sufficient accuracy to reproduce the key dynamic quantities while significantly reducing computational cost, thus enabling real-time simulation.

The physical system, shown in Fig. 1, consists of three main masses:

- Chaser satellite – primary satellite responsible for approaching and docking with the target.
- Target satellite – the object that remains stationary and must be captured.
- Robotic arm with end-effector – simplified as a single mass.

The robotic arm's complex multibody dynamics are replaced with translational spring–damper elements along the x, y, and z axes between the chaser and the end-effector. This configuration models the compliance and damping of the manipulator joints.

The contact with the target satellite is modelled using two sets of springs and dampers, one for each of the left and right clamp of the gripper. These elements simulate the interaction forces between the robot's grippers and the target satellite. However, the gripper clamps are not fixed. They are allowed to rotate, which means that the normal contact forces between the gripper clamps and the target are balanced and do not affect the whole system. In other words, the normal contact forces, although necessary for gripping, are internally compensated by the grippers' freedom of rotation and depend on the gripper's acceleration.

Instead, the forces that affect the overall system are primarily those of friction, which act in directions other than the normal axis of contact. These frictional forces are proportional to the normal contact force, i.e., they depend on the magnitude of the normal forces but act tangentially to the contact surfaces. This frictional interaction is critical to the stability of the docking operation, preventing slippage or unwanted movement between the gripper and the target satellite. The scheme of overall system is shown in Fig. 2.

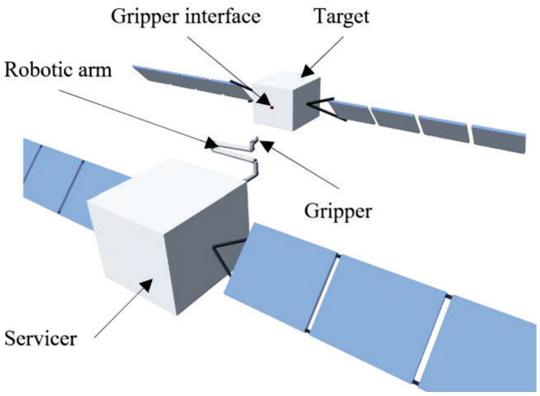

Fig. 1. Multibody model of the berthing system.

https://doi.org/10.52202/083088-0116

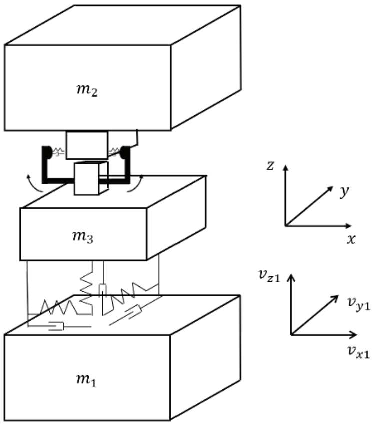

Fig. 2. Schematic representation of the Lumped Parameter Model (LPM).

3. Theoretical Framework and Governing Equations

Fig. **3** shows the Free Body Diagram (FBD) of the system, in which the following quantities appear:

- m_1 : Mass of chaser satellite
- m_2: Mass of robotic arm with end-effector
- m_3: Mass of target satellite
- x_1, y_1, z_1 : Position coordinates of Mass 1 (Chaser)
- x_2, y_2, z_2 : Position coordinates of Mass 2 (Target)
- x_3, y_3, z_3 : Position coordinates of Mass 3 (End-Effector)
- $k_{RX}, c_{RX}, k_{RY}, c_{RY}, k_{RZ}, c_{RZ}$: Stiffness and damping coefficients for the springs and dampers connecting the end-effector to the chaser along the x, y, and z axes
- F_{RX}, F_{RY}, F_{RZ}: Resulting forces along the x, y, and z axes consist of both elastic and damping contributions
- $k_{dx}, c_{dx}, k_{sx}, c_{sx}$: Stiffness and damping coefficients for the springs and dampers between the end-effector and the target along the x-direction for the right and left clamps respectively
- $F_{ay_{sx}}, F_{az_{sx}}, F_{ay_{dx}}, F_{az_{dx}}$: Forces of friction acting on the end-effector during contact

The reference frames for each body (1, 2, and 3) are defined with the origin in the centre of mass of each of them when the lumped parameter simulation starts. Initial conditions are defined at the instant of first contact in the multibody simulation. All subsequent displacements, velocities and accelerations are expressed relative to this initial equilibrium configuration.

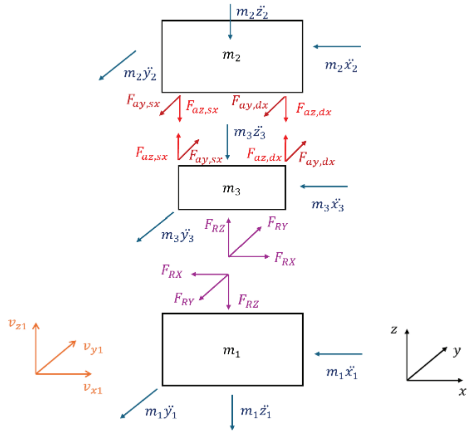

Fig. 3. Free Body Diagram (FBD) of the LPM.

3.1 Chaser (Mass 1)

The chaser is modelled with initial velocities applied along each of the x, y, and z axes. These velocities simulate the initial approach errors introduced by the guidance, navigation, and control (GNC) system.

The chaser is connected to the end-effector through springs and dampers along each axis. The resulting forces along the x, y, and z axes consist of both elastic and damping contributions, modeled as:

$$F_{R,i} = -k_{R,i},(q_i - q_j) - c_{R,i}(\dot{q}_i - \dot{q}_j) \qquad (1)$$

Stiffness and damping coefficients $k_{R,i}$ and $c_{R,i}$ enable the representations of the relative motions between end effector and target due to the movements caused by the stiffness of the robot joints.

3.2 End-Effector (Mass 3)

The end-effector is connected to both the chaser and the target. It undergoes the forces of each. The connection to the chaser generates reaction forces on the end-effector along the x, y, and z axes due to the spring and damper forces defined above.

3.3 Normal contact force

The end-effector is connected to the target by two springs and dampers, one for each gripping point. This setup simulates the resistance to penetration at each contact point. While the normal contact force is a crucial component in modelling the physical interaction between the end-effector and the target, the mass 3 itself is not directly subjected to it due to the rotational degree of freedom of the grippers, as explained in section 2. Instead, the end-effector only experiences the frictional forces, which depend on the magnitude of the normal contact force. These frictional forces act tangentially to the contact surface and play a key role in maintaining the stability of the joint and simulating the necessary reaction forces for the system.

Copyright ©2025 by the International Astronautical Federation (IAF). All rights reserved.

The formulation of the normal contact force for each clamp is given by:

$$F_p = -k\delta - c\dot{\delta} \tag{2}$$

where:

- k is the stiffness of the spring, representing the material and structural properties of the contact area, which defines the amount of resistance provided per unit of penetration.
- c is the damping coefficient, which captures the velocity-dependent dissipative characteristics of the contact. This damping term controls the rate at which penetration velocity is counteracted, reducing the oscillatory behavior that can occur due to rapid contact changes.
- δ represents the penetration depth, which is the overlap distance between the end-effector and the target.
- $\dot{\delta}$ is the rate of change of this penetration (i.e., the rate of penetration).

3.4 Friction Model

As mentioned above, the end-effector is subject to frictional forces in directions parallel to the contact surface, which depend on the relative velocities between the end-effector and the target along the non-contact axes. In this model, friction is treated using a combination of static and dynamic friction coefficients and the implementation of a stick-slip model.

Static friction is the force that must be overcome to initiate motion between two objects that are not moving relative to each other. In this model, static friction is represented by a coefficient denoted as μ_s, set to a value of 0.5. This choice is based on the contact parameters used to define the multibody model, as discussed in [14]. The static friction force acts when the relative velocity between the two masses is close to zero.

The equation governing this force is:

$$F_{static} = \mu_s F_p \tag{3}$$

where F_p is the normal contact force between the end-effector and the target. In the implemented model, when the end-effector and target are in contact and static, the model applies static friction proportional to the normal force. This ensures that the system can remain stationary under certain conditions until a sufficient external force is applied.

Dynamic friction occurs when surfaces begin to slide against each other. It is usually less than static friction, represented by the coefficient μ_d, which is set at 0.3 , from [14]. The dynamic friction force is computed when the relative velocity exceeds a small threshold, denoting the onset of sliding.

The equations used are similar to those for static friction, but use the dynamic coefficient:

$$F_{dynamic} = \mu_d F_p \left(-\frac{v_{rel}}{v_{slip}}\right) \tag{4}$$

where v_{rel} is the relative velocity and v_{slip} is the magnitude of the slip velocity. In the model, the dynamic friction determination includes a control of the relative slip velocity. If the slip velocity exceeds a threshold ($\varepsilon_v = 1 \times 10^{-5}$), from [14], the model applies dynamic friction. If the slip velocity is negligible, indicating a transition phase, static friction is applied instead.

The model calculates the relative velocities between the end effector and the target in the x, y and z directions. This calculation is fundamental in determining whether the surfaces are sliding or sticking:

$$v_{rel,y} = v_{y3} - v_{y2}, \qquad v_{rel,z} = v_{z3} - v_{z2} \tag{5}$$

When a contact force F_p is applied.

If $F_p > 0$ (indicating contact), the model checks the relative slip velocity:

$$v_{slip} = \sqrt{v_{rel_x} + v_{rel_y} + v_{rel_z}} \tag{6}$$

If the slip velocity is above the defined threshold, dynamic friction is calculated, applying forces in the opposite direction of motion.

If the slip velocity is below the threshold, static friction forces are applied, preventing any motion until the applied force exceeds the static friction threshold.

The same logic applies symmetrically to both sides of the end effector. The forces for the left and right sides are calculated using similar equations, ensuring that the model effectively captures the dynamics regardless of the direction of motion.

3.5 Target (Mass 2)

The target is subject to forces caused by contact with the end-effector through the spring-damper system and frictional forces in the other directions. Normal contact forces are balanced by reaction forces on the target, while friction forces act in directions parallel to the contact interface.

3.6 Equation of motion

To provide a complete overview of system dynamics, it is essential to present the governing equations that define the motion of each mass in the model. Below are the nine equations derived from the equations of motion

https://doi.org/10.52202/083088-0116

for each mass, taking into account the forces acting due to springs, dampers, normal contact forces and friction.

At this stage of the analysis, the model neglects rotational degrees of freedom. This modeling choice simplifies the formulation and reduces the dimensionality of the optimization problem. Including rotations introduces a significantly larger number of variables to be tuned by the particle swarm optimization (PSO) algorithm, which makes the search for an optimal solution more complex and computationally demanding. The governing equations for the motion of the three masses are therefore formulated as follows:

$$\begin{cases} m_1\ddot{z}_1 = -k_{RZ}(z_1 - z_3) - c_{RZ}(\dot{z}_1 - \dot{z}_3) \\ m_1\ddot{x}_1 = -k_{RX}(x_1 - x_3) - c_{RX}(\dot{x}_1 - \dot{x}_3) \\ m_1\ddot{y}_1 = -k_{RY}(y_1 - y_3) - c_{RY}(\dot{y}_1 - \dot{y}_3) \\ m_3\ddot{z}_3 = k_{RZ}(z_1 - z_3) + c_{RZ}(\dot{z}_1 - \dot{z}_3) + \\ +F_{az_{dx}} + F_{az_{sx}} \\ m_3\ddot{x}_3 = +k_{RX}(x_1 - x_3) + c_{RX} * (\dot{x}_1 - \dot{x}_3) \\ m_3\ddot{y}_3 = k_{RY}(y_1 - y_3) + c_{RY} * (\dot{y}_1 - \dot{y}_3) + \\ +F_{ay_{sx}} + F_{ay_{dx}} \\ m_2\ddot{z}_2 = -F_{az_{sx}} - F_{az_{dx}} \\ m_2\ddot{x}_2 = 0 \\ m_2\ddot{y}_2 = -F_{ay_{dx}} - F_{ay_{sx}} \end{cases} \quad (7)$$

4. Optimization of Stiffness and damping Parameters for Robotic Arm Movement

To accurately model the stiffness and damping characteristics of the robotic arm along the x-, y- and z-axes, a particle swarm optimization (PSO) algorithm is applied. This optimization technique is chosen for its robustness in handling complex, multidimensional search spaces, ideal for parameter tuning in dynamic systems.

The PSO algorithm adjusted the stiffness and damping $k_{RX}, c_{RX}, k_{RY}, c_{RY}, k_{RZ}, c_{RZ}$ parameters of the arm, minimizing the error between the observed behavior of the multi-body model and the results of lumped parameter model.

In PSO, a swarm of particles, each representing a possible solution, explores the search space to find the optimal values that minimize (or maximize) a specified objective function. In this case, PSO is used to minimize the error between the positions and velocities generated by the simplified lumped parameter model and those from the detailed multi-body model.

An initial population (swarm) of particles is defined, where each particle corresponds to a unique combination of stiffness and damping parameters for the robotic arm. The position of each particle in parameter space represents a potential solution, and each is initialized with a random position and velocity within the predefined limits for the parameters.

The objective function for the PSO algorithm calculates the mean squared error (MSE) between the simulated results of the simplified model and the outputs of the multibody model. This MSE is computed over all three spatial dimensions (x, y, z) and for each mass in the system.

The PSO algorithm aims to find the parameter values that yield the lowest MSE, thus ensuring our model's output closely aligns with the actual multibody simulation.

During each iteration, each particle evaluates its current position by calculating the objective function value. It then updates two main attributes that guide the search, as explained in [21]:

• Personal Best Position (pBest): the best position a particle has visited so far, corresponding to its lowest error.

• Global Best Position (gBest): the best position found among all particles in the swarm, representing the overall lowest error across iterations.

To handle the potentially wide range of parameters, a scaling factor of 10^5 is applied to the limits, ensuring that parameters such as stiffness (which could cover a wide range in physical systems) remained manageable and within realistic values during optimization.

The limits were further tightened to prevent the PSO from selecting unphysical or extreme values, thus focusing the search on plausible parameter values.

The PSO algorithm continues to iterate until one of two conditions is met: either the variation of the objective function has fallen below a defined threshold (Function Tolerance set to 10^{-20}) or a maximum number of iterations has been reached (Max Iterations).

In this case, the algorithm successfully terminates when the particles converge to a parameter set that minimized the MSE, indicating that the behavior of the model closely matches that of the multi-body simulation.

5. Contact Stiffness and Damping for Gripping Points

Within the model, the contact normal forces, the penetrations and the penetration velocities at the gripping points are obtained directly from the multibody simulation. With these values, the contact stiffness and damping coefficients of the gripper interface can be calculated.

The multibody simulation provides the following data:

- Normal forces $F_{p,sx}, F_{p,dx}$ at the left and right contact points;
- Penetration Δ_{sx}, Δ_{dx} at the respective contact points;
- Penetration velocity : $\dot{\Delta}_{sx}$ and $\dot{\Delta}_{dx}$.

These parameters are related by a contact model that follows Hooke's law for stiffness and a damping law for energy dissipation. The contact behavior in terms of

https://doi.org/10.52202/083088-0116

Copyright ©2025 by the International Astronautical Federation (IAF). All rights reserved.

penetration and penetration velocity is described by the following relation:

$$F_n = k\Delta + c\dot{\Delta} \tag{10}$$

Th damping coefficient c is set as a fixed percentage of the stiffness coefficient k, based on empirical data for the system. Specifically:

$$c = \alpha \cdot k \tag{11}$$

where α is a constant factor representing the ratio between damping and stiffness, depending on the expected damping behavior and materials property.

Once the damping coefficient c is defined, the stiffness is determined by rearranging the equation for each time step:

$$k = \frac{F_n}{\Delta + \alpha\dot{\Delta}} \tag{12}$$

4. Results

The results of the simulation provide insight into the dynamic interaction between the chaser satellite, the robotic arm, and the target. Using the lumped parameter model with optimized contact stiffness and damping coefficients, the behavior of the system is simulated and analyzed under different conditions.

Two analyses are carried out with different target velocities in all directions (x, y and z) for the chaser. In the first analysis the chaser reaches a target velocity of $1\ [mm/s]$ in all directions, while in the second analysis a target velocity of $3\ [mm/s]$ is imposed. These first two analyses allow the response of the model to different approach velocities to be evaluated and provide valuable insights into how the interaction dynamics between the chaser, robot arm and target adapt to changes in velocity.

To evaluate the performance of the LPM and quantify the error compared to the multibody model, the coefficient of determination R^2 is used. This metric assesses how well the simulated values reproduce the variability of the measured data and is defined as:

$$R^2 = 1 - \frac{RSS}{TSS}$$

where:

- RSS (Residual Sum of Squares): Represents the error between the measured data (y_i) and the predicted data $(\hat{y}_i)$ from the model, and is given by:

$$RSS = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

- TSS (Total Sum of Squares): Represents the total variability of the observed data (y_i) relative to their mean $(\bar{y})$, calculated as:

$$TSS = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

Here,

- y_i are the measured data points from the multibody model,
- $\hat{y}_i$ are the predicted data points from the lumped parameter model,
- $\bar{y}$ is the mean of the observed data.

The R^2 metric measures the proportion of variability in the target data y_i that is captured by the model $\hat{y}_i$.

The possible range of R^2 are:

- If $R^2 = 1$, the model output perfectly matches the measured data, indicating an excellent fit.
- If R^2 is close to 1 but slightly less, the model closely aligns with the real behavior of the system.
- If $R^2 = 0$, the model fails to explain any variability in the data, suggesting poor performance.

For simplicity in writing, in subsequent analyses we will denote R^2 with r. By using this metric, the accuracy of the LPM in replicating the multibody model can be quantitatively assessed under different velocity conditions.

4.1 Analysis with 1 mm/s initial velocity of the chaser

The simulation timeframe is defined to capture the relevant contact period, with time starting from $t = 6.975\ [s]$ to $t = 7\ [s]$, based on multibody simulation data. During this period, the parameters are optimized to match the behavior observed in the multi-body model, ensuring that the kinematic and dynamic responses were in agreement.

The mass properties for each component have been set as follows:

- Chaser satellite: $m_1 = 1500\ [kg]$
- Target: $m_2 = 500\ [kg]$
- Robotic arm and end-effector: $m_3 = 14\ [kg]$

Initial position and velocity values for each body have been taken from the multi-body simulation to ensure that the initial states are consistent between the two models, and are listed in Table 1,2 and 3.

Table 1. Chaser velocities at the initial instant of contact — analysis for a chaser approach velocity of 1 mm/s

Chaser velocity at first instant of contact	
$v_{1x}(6.975)$	$9.2192 \cdot 10^{-04}\ [m/s]$
$v_{1y}(6.975)$	$9.9673 \cdot 10^{-04}\ [m/s]$
$v_{1z}(6.975)$	$9.8205 \cdot 10^{-04}\ [m/s]$

https://doi.org/10.52202/083088-0116

Table 2. Target velocities at the initial instant of contact — analysis for a chaser approach velocity of 1 mm/s

Chaser velocity at first instant of contact	
$v_{2x}(6.975)$	$2.7928 \cdot 10^{-04}\ [m/s]$
$v_{2y}(6.975)$	$1.5147 \cdot 10^{-04}\ [m/s]$
$v_{2z}(6.975)$	$8.2543 \cdot 10^{-04}\ [m/s]$

Table 3. End-Effector/Robot velocities at the initial instant of contact — analysis for a chaser approach velocity of 1 mm/s

Chaser velocity at first instant of contact	
$v_{3x}(6.975)$	$-0.0025\ [m/s]$
$v_{3y}(6.975)$	$6.8 \cdot 10^{-04}\ [m/s]$
$v_{3z}(6.975)$	$-0.0012\ [m/s]$

Through the optimization process, the stiffness and damping coefficients have been achieved for each axis (x, y, z) to minimize the error between the LPM and the multi-body reference and are listed in Table 4. Instead, the contact stiffness and damping parameters are given in Table 5.

Table 4. Robot Stiffness and Damping Coefficients— analysis for a chaser approach velocity of 1 mm/s

Robot Stiffness and Damping Coefficients	
$k_{RX}\ \ [N/m]$	10^5
$k_{RY}\ \ [N/m]$	$2 \cdot 10^5$
$k_{RZ}\ \ [N/m]$	0
$c_{RX}\ \ [Ns/m]$	$1 \cdot 10^5$
$c_{RY}\ \ [Ns/m]$	0
$c_{RZ}\ \ [Ns/m]$	$0.0133 \cdot 10^5$

Table 5. Contact Stiffness and Damping Coefficients— analysis for a chaser approach velocity of 1 mm/s

Contact Stiffness and Damping Coefficients	
$k_{dx}\ \ [N/m]$	$3.126 \cdot 10^7$
$k_{sx}\ \ [N/m]$	$3.3682 \cdot 10^7$
$c_{dx}\ \ [Ns/m]$	1.9297
$c_{sx}\ \ [Ns/m]$	2.0790

These values are obtained by particle swarm optimization, with each parameter tuned to achieve minimum deviation from the multi-body reference under the imposed conditions. Optimization constraints are set within the range of realistic stiffness and damping values to ensure that the model could replicate the physical properties of the system. The resulting parameters show effective contact behavior, with the derived stiffness and damping coefficients closely matching the expected system dynamics.

Graphical results of simulation are presented and analyzed below, focusing on absolute displacements of chaser, target and end effector, and normal forces acting on both sides of system (right and left).

The graphs compare the results of two modelling approaches: the multibody simulation (blue solid line), which represents a detailed and high-fidelity model, and the lumped parameter model (red dashed line), which provides a simplified representation intended to approximate the behavior of the multibody system with reduced computational complexity.

The position of the chaser along the x, y and z axes over time is shown in Fig. 4. Both the multibody simulation (blue solid line) and the LPM (red dashed line) are shown to compare the two approaches.

The chaser shows a continuous increase in displacement along the x axis, suggesting steady motion in this direction. The two models are in close alignment, indicating that the LPM accurately reproduces the behavior of the multibody model. Along the y-axis, a similar trend is observed with steady linear motion, and again the LPM closely follows the multibody simulation, although slight deviations become apparent towards the end of the time period.

For the z-axis, the pattern remains consistent with minimal deviation between the two models. This suggests that the LPM is a good approximation in capturing the movement of the chaser in three dimensional space.

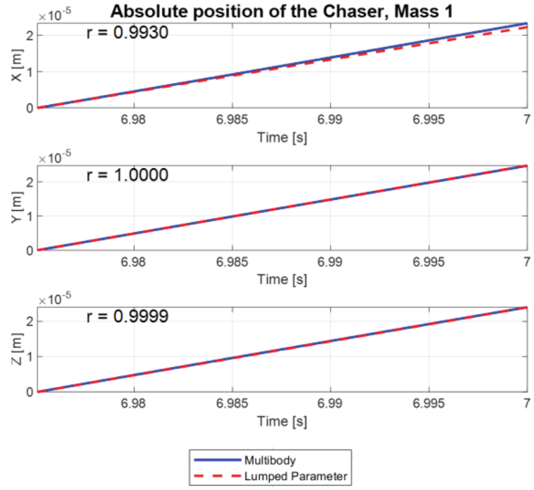

Fig. 4. Absolute position of the chaser— analysis for a chaser approach velocity of 1 mm/s

The absolute position of the target along the x, y and z axes is shown in Fig. 5. The displacement behavior in these directions provides insight into the differences between the multi-body simulation and the LPM.

In the x axis displacement, the target exhibits a more complex motion pattern compared to the chaser. While the LPM successfully captures the overall trend, there are noticeable deviations, particularly towards the later stages of the simulation. These differences suggest that

https://doi.org/10.52202/083088-0116

the lumped parameter model has some limitations in fully representing the complex dynamic interactions that occur in the multi-body system, especially under certain conditions. For the y-axis displacement, the lumped parameter model underestimate the movement of the target, especially after $t = 6.99\ [s]$. This underestimation is probably due to the simplifications inherent in the lumped parameter approach, which may not account for all the dynamic forces and constraints acting on the target. Nevertheless, the general behavior is still useful for the simulation of the target dynamic.

In Fig. 6 in shown the absolute position of the end effector along the x, y and z axes. On the x axis, the two models show a close match, with LPM effectively capturing the general pattern of motion observed in the multi-body model, although there are some deviations.

In particular, the multibody model shows an initial oscillatory behavior that is not reflected in the LPM. This suggests that certain vibrational dynamics inherent to the multibody simulation are not accounted for in the simplified approach, resulting in less accurate predictions at this early stage. In any case, the estimation of the position of the end effector is not relevant to the analysis in focus.

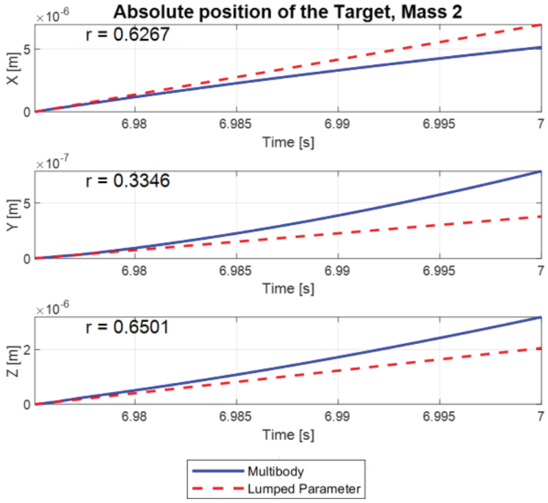

Fig. 5. Absolute position of the target— analysis for a chaser approach velocity of 1 mm/s

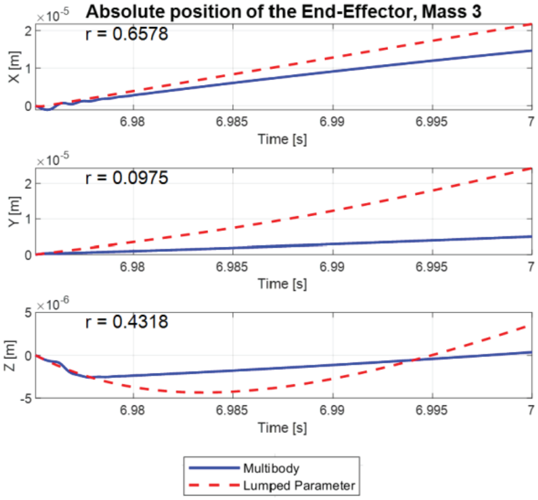

Fig. 6. Absolute position of the End-Effector— analysis for a chaser approach velocity of 1 mm/s

Fig. 7 and Fig. 8 shows the normal forces acting on the right and left gripping points. Overall, both the LPM and multi-body models show similar trends, capturing the general behavior of the forces acting on the gripping points. In the transient phase some discrepancies are observed between the two models. Despite these differences during the initial transient, the LPM provides a reasonable approximation of normal forces. Its ability to match the steady-state values effectively demonstrates its potential as a computationally efficient tool for capturing the primary dynamics of the system.

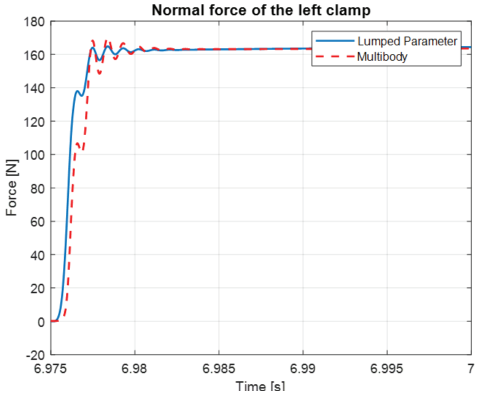

Fig. 7. Normal force of the left clamp— analysis for a chaser approach velocity of 1 mm/s

Copyright ©2025 by the International Astronautical Federation (IAF). All rights reserved.

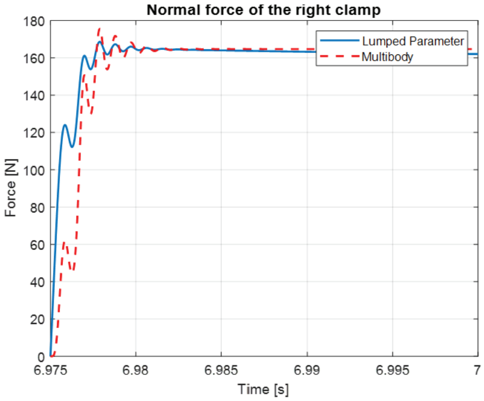

Fig. 8. Normal force of the right clamp— analysis for a chaser approach velocity of 1 mm/s

4.2 Analysis with 3 mm/s initial velocity of the chaser

In this analysis the system behavior is examined with an increased target velocity of the chaser, set to 3 mm/s along each axis. With this higher velocity, the contact occurs slightly earlier at $t = 6.7366\ [s]$ and the time frame considered extends to $t = 7\ [s]$.

An additional optimization is carried out in order to simulate the movements due to the joint stiffness of the robot in the best possible way. The resulting stiffness are shown in Table 6. All remaining parameters are unchanged.

Table 6. Robot Stiffness and Damping Coefficients— analysis for a chaser approach velocity of 3 mm/s

Robot Stiffness and Damping Coefficients	
$k_{RX}\ [N/m]$	$5.6695 \cdot 10^3$
$k_{RY}\ [N/m]$	$4.3492 \cdot 10^3$
$k_{RZ}\ [N/m]$	$4.5822 \cdot 10^3$
$c_{RX}\ [Ns/m]$	$0.3659 \cdot 10^3$
$c_{RY}\ [Ns/m]$	$3.9841 \cdot 10^3$
$c_{RZ}\ [Ns/m]$	$0.7166 \cdot 10^3$

By comparing the results in these conditions with those from the previous analysis carried out at a lower chaser initial velocity, it is possible to assess how the higher initial velocity affects the accuracy and dynamic response of the lumped parameter model compared to the multi-body model. Notably, in this case r deviates more from 1.

With the initial velocity of the chaser set to 3 mm/s the LPM and the multibody model shows a similar trend across the x, y and z positions for each component, but some deviations are noticeable compared to the lower velocity scenario.

In the analysis of the position of the chaser, shown in Fig. 9 both models show similar trends in the x, y, and z-axes, as observed at the lower velocity. However, at this higher target velocity, small discrepancies become more pronounced. The LPM slightly overestimates the displacement towards the end, probably due to the simplified handling of higher dynamic forces.

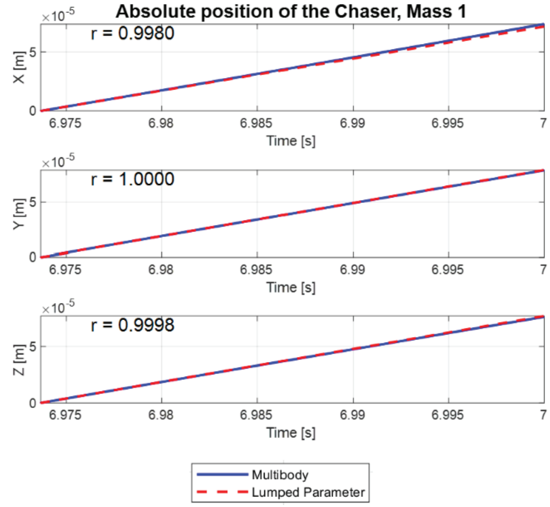

Fig. 9. Absolute position of the chaser— analysis for a chaser approach velocity of 3 mm/s

For target position, shown in Fig. 10, the lumped model captures the overall trend on the x-axis, but deviates more significantly in the later stages, indicating challenges in modelling dynamic interactions at higher velocities.

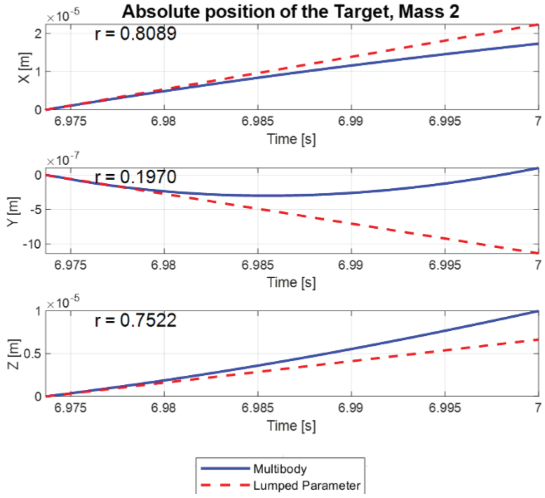

Fig. 10. Absolute position of the target— analysis for a chaser approach velocity of 3 mm/s

For the end effector, shown in Fig. 11, both models agree reasonably well on the x-axis, although the lumped model misses certain oscillations seen in the multi-body results. On the y and z axes, the lumped model initially overestimates displacement but aligns more closely

https://doi.org/10.52202/083088-0116

towards the end, highlighting its limitations in capturing rapid transients at higher velocities.

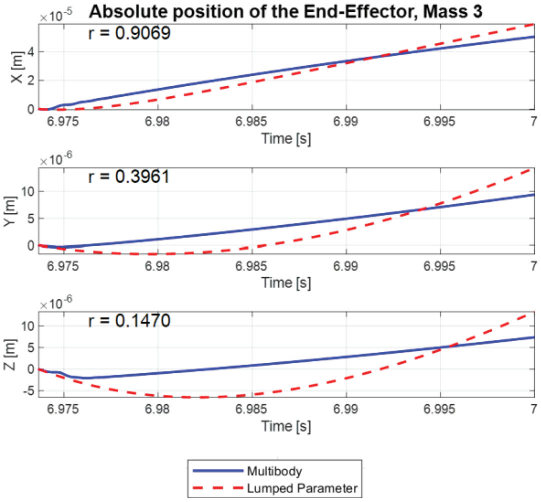

Fig. 11. Absolute position of the End-Effector— analysis for a chaser approach velocity of 3 mm/s

The analysis of the normal forces at 3 mm/s target velocity, illustrated in Fig. 12 and Fig. 13, reveals similar patterns to the lower-speed scenario.

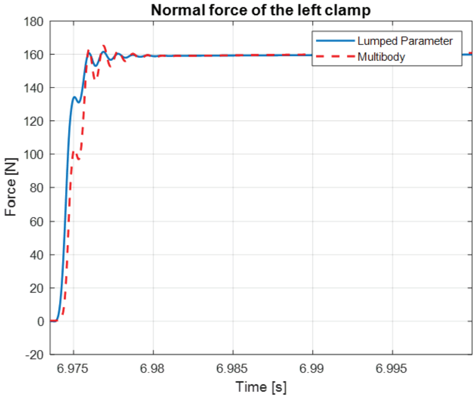

Fig. 12. Normal force of the left clamp— analysis for a chaser approach velocity of 3 mm/s

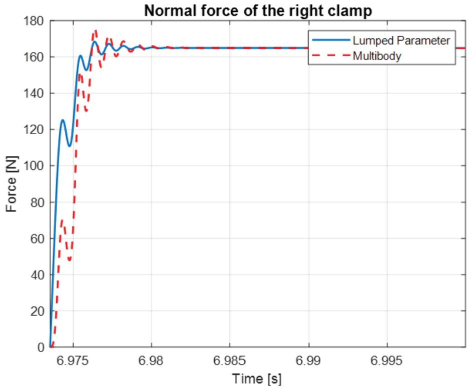

Fig. 13. Normal force of the right clamp— analysis for a chaser approach velocity of 3 mm/s

5. Discussion

The lumped parameter model is employed because of the significant reduction in computational time compared to the multi-body model. The simulation time is mainly influenced by the processing power of the hardware (CPU) on which the model is running and the choice of numerical solver used during the simulation. Table 7 shows the execution times achieved with different solvers, for the first analysis performed where simulated time is 0.025 s. The table shows how variations in solver algorithms and their associated computational complexity can have a significant impact on performance ode23t completes the simulation in a time shorter than simulated time, enabling real-time operation.

Table 7. Simulation Times for Different Solvers in Lumped Parameter Model.

Solver type	Simulation time	Mean error r of displacements between the LPM and the multi-body model
ode23	0.0405 [s]	0.6434[−]
ode23t	0.0186 [s]	0.6434 [−]
ode 23tb	0.0520 [s]	0.6434 [−]
ode23s	0.1990 [s]	0.6434 [−]
ode45	0.0795 [s]	0.6434 [−]
ode15s	0.0657 [s]	0.6440 [−]
ode23	0.0405 [s]	0.6434 [−]
ode113	0.0785 [s]	0.6434[−]
ode78	0.1243 [s]	0.6434 [−]

Ode23tsolver uses a stabilized trapezoidal Runge-Kutta method specifically designed to solve moderately stiff systems. Its efficiency in handling such systems contributes to the reduced computation time, making it a suitable choice when faster simulation performance is required. While the solver ode15s achieves slightly better model accuracy, the improvement is not significant enough to justify its use, given that its simulation time is considerably higher than that of ode23t.

With a simulation time interval of $0.0263\ [s]$, this model is capable of executing code in real time, ensuring that the simulation can run concurrently with the actual system processes without significant delays.

The data presented in the table are obtained using a Dell Inspiron laptop with an Intel Core i7 processor, 16GB of RAM, and a solid-state drive (SSD). The simulation times are influenced by the hardware specifications, including CPU performance and memory capacity, as well as the chosen solver algorithms. Due to the inherent variability in computational performance caused by factors such as background processes, CPU load, memory usage, and the specific solver algorithm used the execution times may fluctuate across different runs. The values reported in Table 7 represent the average simulation times calculated over 5 simulations to account for these variations.

Although the lumped parameter model introduces minor discrepancies, particularly at higher velocities, it successfully captures the fundamental dynamics of the system. This computational efficiency is particularly beneficial in preliminary analysis phases or scenarios that require fast iteration cycles. By balancing computational efficiency with acceptable accuracy, the lumped parameter model provides an optimal trade-off, making it highly suitable for simulating the dynamics of chaser-target satellite interactions.

6. Conclusions

This paper successfully presented a new hybrid modelling framework for real-time spacecraft berthing simulations. The LPM effectively reproduced the key dynamic features of a more complex multibody simulation while drastically reducing the required computational time. It was demonstrated that Particle Swarm Optimization is a robust method for calibrating the stiffness and damping parameters, ensuring the accuracy of the model across various approach conditions. Continuous simulation across the orbital, proximity and contact phases guarantees real-time execution, making this a suitable methodology for preliminary mission analysis and onboard applications. Future work will extend this framework to cooperative multi-satellite missions. These extensions will be validated using hardware-in-the-loop testbeds and applied to advanced scenarios, including in-orbit servicing and lunar assembly missions.

Acknowledgements

This research is coordinated and funded by the Italian Space Agency (Agenzia Spaziale Italiana, ASI) in the framework of the RESEARCH DAY “GIORNATE DELLA RICERCA ACCADEMICA SPAZIALE” initiative through the contract no. ASI-BVTECH-2023-2-E.0

The authors thank ASI for its support and coordination.

References

[1] A. Flores-Abad, O. Ma, K. Pham, and S. Ulrich, A review of space robotics technologies for on-orbit servicing, Progress in Aerospace Sciences, vol. 68, 2014, pp. 1–26. doi: 10.1016/j.paerosci.2014.03.002.

[2] E. Papadopoulos, F. Aghili, O. Ma, and R. Lampariello, Robotic Manipulation and Capture in Space: A Survey, Frontiers in Robotics and AI, vol. 8, 2021. doi: 10.3389/frobt.2021.686723.

[3] X. L. Ding, Y. C. Wang, Y. B. Wang, and K. Xu, A review of structures, verification, and calibration technologies of space robotic systems for on-orbit servicing, Science China Technological Sciences, vol. 64, no. 3, 2021, pp. 462–480. doi: 10.1007/s11431-020-1737-4.

[4] NASA Goddard Space Flight Center, *On-Orbit Satellite Servicing Study: Project Report*, NASA GSFC, Greenbelt, MD, October 2010.

[5] K.T. Alfriend, S.R. Vadali, P. Gurfil, J.P. How and L.S. Breger, Spacecraft Formation Flying. Dynamics, control and navigation, Butterworth-Heinemann, 2010. doi: 10.1016/C2009-0-17485-8

[6] R.H. Battin, An Introduction to the Mathematics and Methods of Astrodynamics, AIAA, 1999. doi: 10.2514/4.861543

[7] K. Yamanaka, F. Ankersen, New state transition matrix for relative motion on an arbitrary elliptical orbit, *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 1, 2002, pp. 60–66. doi: 10.2514/2.4875.

[8] D. Arney, J. Mulvaney, C. Williams, R. Sutherland, C. Stockdale, *In-space Servicing, Assembly, and Manufacturing (ISAM) State of Play 2022 Edition*, NASA Langley Research Center, Hampton, VA, 2022.

[9] S. Wu, F. Mou, Q. Liu, and J. Cheng, Contact dynamics and control of a space robot capturing a tumbling object, Acta Astronaut, vol. 151, 2018, pp. 532–542. doi: 10.1016/j.actaastro.2018.06.052.

[10] Canadian Space Agency, About Canadarm, https://www.asc-csa.gc.ca/eng/canadarm/about.asp, (accessed 18.08.2025).

[11] European Space Agency, European Robotic Arm, https://www.esa.int/Science_Exploration/Human_and_Robotic_Exploration/International_Space_Station/European_Robotic_Arm, (accessed 18.08.2025).

[12] P. Palmieri, M. Gaidano, A. Ruggeri, L. Salamina, M. Troise, S. Mauro, An inflatable robotic assistant for onboard applications, *Proceedings of the International Astronautical Congress (IAC)*, Dubai, UAE, 25–29 Oct. 2021.

[13] E. Scantamburlo, M. Ferrauto, D. Sorli, S. Mauro, Modeling the transition between orbital and contact phase in rendezvous and proximity operations, *Proceedings of the 12th International Workshop on Metrology for Aerospace (MetroAeroSpace)*, Naples, Italy, 18–20 June 2025. doi: 10.1109/MetroAeroSpace64938.2025.11114679

[14] D. Sorli, M. Ferrauto, M. Melchiorre, P. Palmieri, L. Salamina, S. Mauro, A simulation tool to evaluate different capture strategies in a berthing maneuver, *Proceedings of the ASME International Mechanical Engineering Congress and Exposition (IMECE 2024)*, Portland, OR, USA, 17–21 Nov. 2024. doi: 10.1115/IMECE2024-144113.

[15] D. Sorli, G. Calvo, M. Ferrauto, F. Giardina, M. Troise, S. Mauro, Study of the influence of contact model parameters on a berthing operation, Proceedings of the 75th International Astronautical Congress (IAC 2024), Milan, Italy, 14-18 Oct 2024.

[16] P. Palmieri, M. Troise, L. Salamina, M. Gaidano, M. Melchiorre, and S. Mauro, An Inflatable 7-DOF Space Robotic Arm for Active Debris Removal, in: M. Okada (eds), Advances in Mechanism and Machine Science. IFToMM WC 2023, Springer, Cham, 2023, pp. 580–589. doi: 10.1007/978-3-031-45770-8_58.

[17] P. Palmieri, M. Troise, M. Gaidano, M. Melchiorre, and S. Mauro, Inflatable Robotic Manipulator for Space Debris Mitigation by Visual Servoing, in: 2023 9th International Conference on Automation, Robotics and Applications, ICARA 2023, IEEE, 2023, pp. 175–179. doi: 10.1109/ICARA56516.2023.10125753.

[18] NASA, OSAM-1 Mission, https://www.nasa.gov/mission/on-orbit-servicing-assembly-and-manufacturing-1/, (accessed 18.08.2025).

[19] E. Tunstel, C. Thayer, B. Hayashi, R. Saltus, ModuLink: A robotic manipulation applique for in-space servicing vehicles, *Proceedings of the IEEE Aerospace Conference*, Big Sky, MT, USA, 4–11 March 2023. doi: 10.1109/AERO55745.2023.10115712.

[20] P. Palmieri, M. Gaidano, M. Troise, L. Salamina, A. Ruggeri, S. Mauro, A deployable and inflatable robotic arm concept for aerospace applications, *Proceedings of the IEEE International Workshop on Metrology for Aerospace (MetroAeroSpace)*, Naples, Italy, 23–25 June 2021, pp. 453–458. doi: 10.1109/MetroAeroSpace51421.2021.9511654.

[21] D. Wang, D. Tan, L. Liu, Particle swarm optimization algorithm: An overview, *Soft Computing*, vol. 22, no. 2, 2018, pp. 387–408. doi: 10.1007/s00500-016-2474-6.