Politecnico di Torino

Politecnico di Torino

Master's Degree in Aerospace Engineering

Academic Year: 2025 - 2026

Graduation Session: March - April 2026

Multibody analysis of a deployable device for orbital debris removal

Advisors:
Prof. Alfonso Pagani
Ing. Daniele Catelani

Candidate:
Antonello Marino
S308824

Abstract

The progressive congestion of Earth's orbital environment, primarily due to the accumulation of end-of-life satellites, represents one of the most critical challenges for the sustainability of the space environment and the success of future missions. The risk of cascading collisions, known as Kessler Syndrome, has made the development of Active Debris Removal technologies a central focus of space research in recent decades. In this context, space debris recovery systems play a fundamental role, as they have to guarantee absolute reliability in a hostile operating environment, with no possibility of maintenance and in the presence of non-cooperative targets.

This work focuses on the multibody analysis of a deployable origami mechanism designed to capture space debris. While the conceptual design of the device has already been defined, previous studies exhibit some limitations in assessing the effective operability over different size scales. In fact, these investigations, being preliminary in nature, have focused on the general feasibility of the project, being based on analytical formulations whose validity is restricted within very narrow operating ranges. In the space domain, this approach is well suited only for very early design phases, as a small variation in dimensions/mass and consequently in inertia can lead to interruptions in analysis or unexpected behavior. Therefore, as a subsequent step of investigation, this work introduces a preliminary parametric multibody analysis carried out using MSC Adams software. The main objective was to overcome the limitations of the deterministic model by defining the key design variables in a parametric manner. MSC Adams' design variables or automated external scripts have been used in order to map the influence of the main parameter variations on the system's dynamic behavior.

This analysis revealed several latent critical issues that were not detectable in previous approaches. In particular, the mechanism was found to be highly sensitive even to minimal variations in its characteristic geometric dimensions. As a consequence, it was necessary to make a function of a scaling parameter each relationship, namely: the tiles' thickness, the inertial characteristics, the spatial and temporal decisions related to the effective start of each mission phase and the stiffness and damping coefficients related to the Proportional–Derivative (PD) control simulating the actuator.

In conclusion, the results show that parametric multibody analysis can provide a valuable basis for the operational success of the mission. In fact, this approach makes it possible to define operational safety margins and to propose design optimization, thus providing essential guidelines for the future space qualification of the device.

Contents

List of Figures

List of Tables

List of Acronyms

Adams Automated Dynamic Analysis of Mechanical Systems. iv, vi, 12, 14, 21, 23, 25–27, 29, 32, 33, 37, 41, 51, 54, 55, 62, 65, 71, 75

AR&C Autonomous Rendezvous and Capture. 74

AZ Approach Zone. 35

CS-1 ClearSpace-1. 35

DOF Degree of freedom. 35

ESA European Space Agency. 1–3, 5–7, 68

FOV Field of view. 67, 69

GEO Geostationary Earth Orbit. 5

GNC Guidance Navigation and Control. 35

ISS International Space Station. iv, 8

KOZ Keep out zone. 35

LEO Low Earth Orbit. 3–5

LiDAR Light Detection and Ranging. 66–69

LOS Line of sight. 36, 42, 45, 65

MBD Multibody Dynamics. 12, 13, 32

MEO Medium Earth Orbit. 5

NASA National Aeronautics and Space Administration. 4, 8, 73

PD Proportional–Derivative. i, 20, 29, 33, 35, 40, 41, 44, 48, 51, 53, 54, 60–62, 71, 73, 75

PID Proportional–Integral–Derivative. 33

TRL Technology Readiness Level. 65, 66, 68, 69, 71, 73

VBN Vision based navigation. 69

VEGA Advanced Generation European Vehicle. 35

VESPA Vega Secondary Payload Adapter. 35

WSU Washington State University. iv, 9

1. Introduction

Orbiting satellites are indispensable for today's society since they support a wide range of domains like space science, Earth observation, meteorology, climate monitoring, telecommunications, navigation systems and human space exploration. Thanks to their orbital position, they make it possible to gather scientific data, enable commercial operations and deliver all essential services, unlocking exceptional opportunities for research and exploration.[1] However, over the past decades, the continuous growth of space activities has led to the emergence of a new and largely unforeseen hazard: **space debris**.

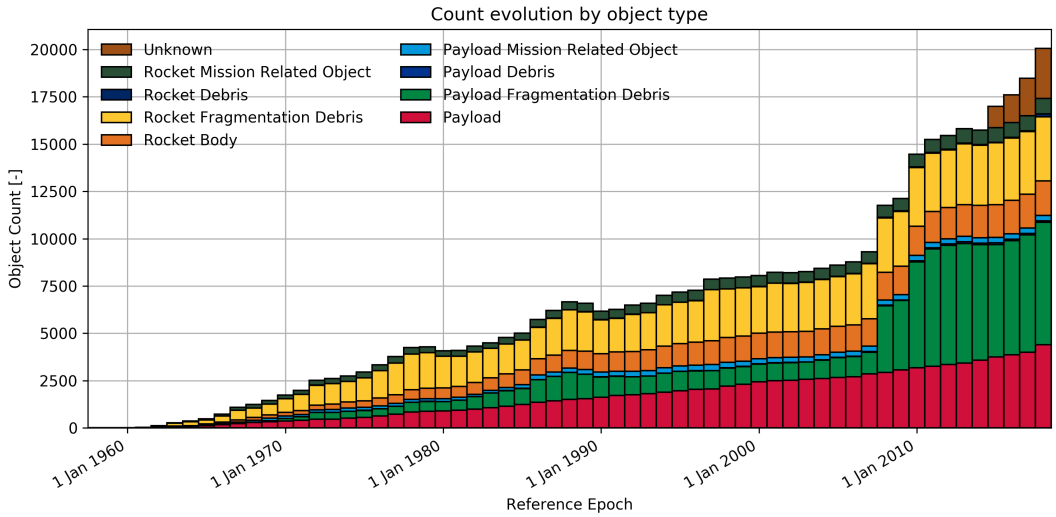


Figure 1.1.: Distribution of space debris in Earth orbit.

Credits: © European Space Agency (ESA).

The graph clearly illustrates the growing issue of space debris. Since the first launch in 1957, there has been a steady upward trend. While the increase in payloads placed in orbit follows an almost linear trend, the amount of debris generated is growing exponentially, with a significant acceleration in the last decade. A similar pattern is observed for launch vehicles, though with lower growth rates. Furthermore, it can be noted the appearance of "unknown objects" since 2016 suggesting updates in ESA's statistical and predictive methodologies.

1.1. The space debris problem

The accumulation of space debris is a direct consequence of decades of space activity conducted without comprehensive end-of-life management strategies. Since the launch of the first satellites, orbital explosions and colli-

sions have progressively increased the density of hazardous debris in Earth's orbits. Despite long-standing mitigation strategies and the continuous implementation of new end-of-life disposal policies, these events are increasingly common and their trend shows no signs of reversing. Nowadays, a set of international standards and best practices outlines the main actions required to ensure the sustainable use of outer space, including:

- minimising the amount of "shedding" [1] during nominal operations;
- "passivating" spacecraft once at the end of their lives to prevent explosions
- removal of defunct objects from operational orbits, either by de-orbiting them or moving them to a "graveyard orbit"
- prevention of collisions through careful orbital design and collision avoidance manoeuvres

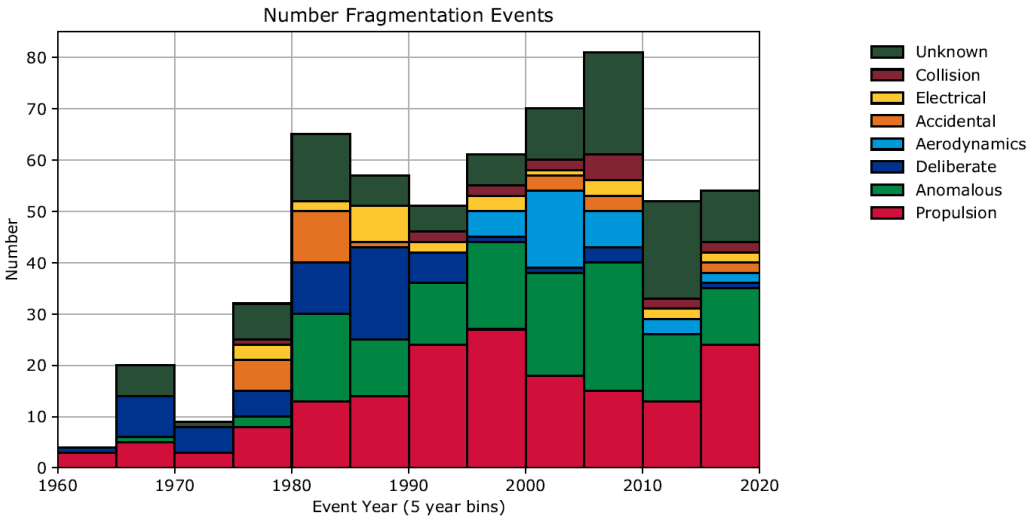
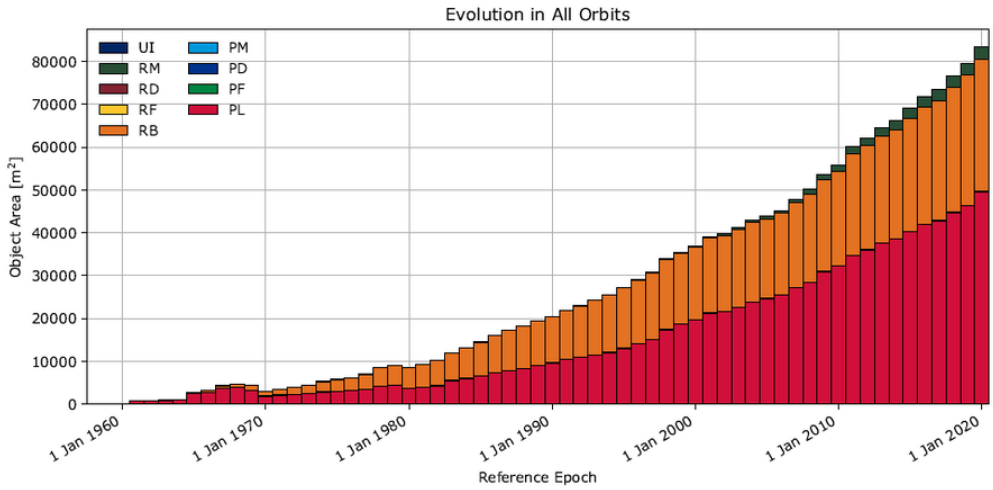


Figure 1.2.: Increasing number of fragmentation events

Credits: © ESA.

However, considering the growth of space operations, additional technological advancements and stricter regulatory oversight may be required to achieve a really sustainable space environment [2]. Recent analyses suggest that regulatory oversight is critical not just for safety, but also for the orbital economy. Studies conducted by the University of Malaga and highlighted by BusinessCom describe the debris issue as a "market failure", where the absence of property rights leads to the overexploitation of resources and "negative externalities". Without intervention, the continuous accumulation of fragments could render space non viable for future commercial and scientific activities [3].

[1] In the context of space operations, "shedding" refers to the unintentional release of small components or materials from a spacecraft during its nominal lifecycle, including the detachment of paint flakes, thermal insulation fragments, unburnt solid propellant particles, or debris from pyrotechnic deployment mechanisms.



(c) Evolution of area.

Figure 1.3.: Distribution of space debris in Earth orbit.

Credits: © ESA.

The graph provides further insight into what has been analyzed in the Figure 5.2. Even when considering spatial occupation in terms of surface area (m^2), it is clear that all orbits are becoming increasingly saturated. The legend also follows the one shown in figure 5.2.

1.1.1. The Kessler Syndrome

Donald Kessler, a NASA expert on space debris, observed that beyond a critical point, debris accumulation becomes unavoidable. Collisions between objects produce new fragments, which increase the probability of further collisions, creating a chain reaction. This phenomenon attracted NASA's attention in the 1970s, when abandoned Delta rocket stages left in orbit exploded and generated large clouds of fragments. Kessler showed that as soon as debris in a specific orbit reaches a critical mass, collisions continue even if no new objects are launched and the risk to satellites and spacecraft of colliding with debris grows until the orbit becomes inaccesible. It was estimated that this threshold would be reached within 30 to 40 years, and today some experts believe that Low Earth Orbit (LEO) has reached critical levels, as Kessler predicted. Therefore, the urgency of the problem is clear: debris levels in LEO have increased by 50% in the last five years alone, leading many experts to warn that orbital region is already at risk of a cascading debris scenario [4, 5].

1.1.2. Space Debris Environment and Mitigation Approaches

As highlighted in the previous paragraph, this theoretical threat is strongly supported by recent empirical findings. According to ESA's debris modelling tool *MASTER*[6] and NASA's *Orbital Debris Quarterly News* [7], the density of objects in Low Earth Orbit (LEO) has reached critical levels. Specifically, at an altitude of approximately 550 km, the number of debris objects is now comparable to that of operational satellites. While these altitude ranges remain a concentration

peak for communication constellations, recent trends indicate a shift in operational strategies: compared to previous years, a significant proportion of active payloads are now operating at lower altitudes, with about 25% flying below 500 km. This altitude is heavily populated by the Starlink constellation, which relies on atmospheric drag to burn up satellites after 5–6 years. However, the density of traffic has severe implications: reports indicate that between December 2022 and May 2023 alone, Starlink had to make 25,000 collision-avoidance maneuvers—approximately 137 per day. On the other hand, constellations at higher altitudes (around 1200 km), such as OneWeb, cannot rely on drag and must employ active mitigation strategies. Those satellites are equipped with sufficient fuel for de-orbiting and specific magnetic grappling fixtures to facilitate capture by future debris removal solutions [3].
Before examining quantitative data on orbital population, it is necessary to define the physical boundaries of the space regions surrounding Earth:

Table 1.1.: Classification of Earth orbital regions by altitude [8].

Orbital Region	**Altitude Range (h)**
Low Earth Orbit	180 km – 2000 km
Medium Earth Orbit	2000 km – 35786 km
Geostationary Orbit	~ 35786 km

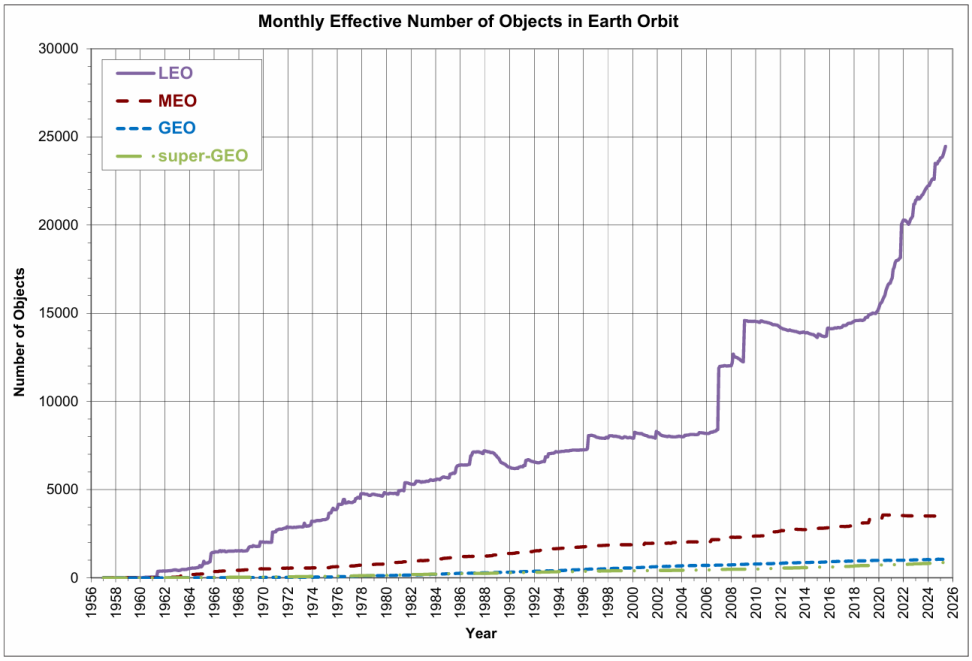


Figure 1.4.: Monthly effective number of objects in Earth orbit.

Credits: © NASA.

The graph, based on National Aeronautics and Space Administration (NASA)'s Orbital Debris Quarterly News [7], clearly shows that most space objects are in LEO. The trend, initially almost linear, accelerated sharply in 2007, transforming into exponential growth over the last decade. That is linked to the new industrial paradigm of space, which prefers LEO in order to facilitate the launch

operations and communications in terms of energy. Different trends can be observed in other orbital regions: Geostationary Earth Orbit (GEO) and super-GEO orbits show linear growth but with very low rate, constrained by the specificity of the services hosted and the strict management of available orbital positions. The Medium Earth Orbit (MEO) sector, on the other hand, experienced a growth with slightly higher rates until 2020, driven by the consolidation of satellite navigation systems, before stabilizing in a phase of maintenance of existing infrastructure. Considering the trend in the graph there is no doubt that the year 2007 should be investigated by consulting the quarterly news reports for that year [9]. The jump is attributable to an anti-satellite test conducted by China on the Fengyun-1C satellite, that generated thousands of traceable fragments, causing an instantaneous increase in the population of debris in LEO never before recorded.

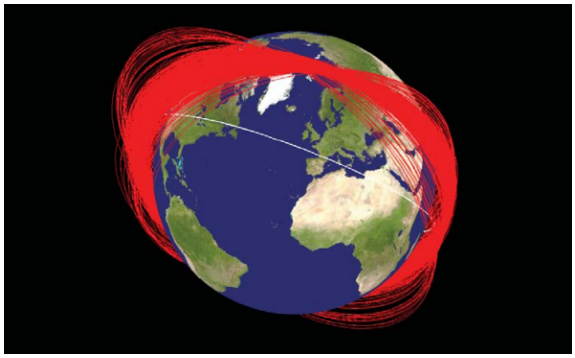


Figure 1.5.: Space debris' orbits after collision of Fengyun-1C satellite

Credits: © NASA.

Therefore it is possible to state that the primary driver of orbital debris generation is in-orbit fragmentation. Also in 2024, due to an increase in major fragmentation events, more than 3,000 new objects were added to the orbital environment.

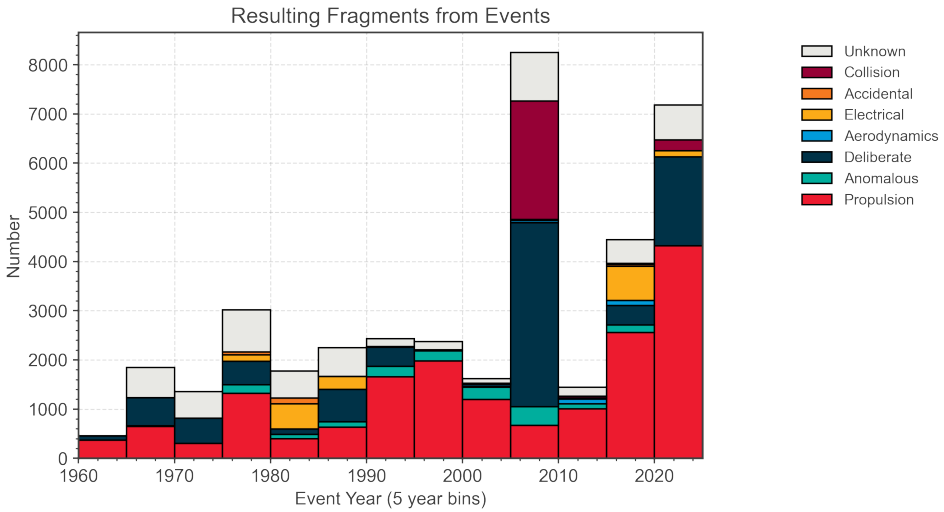


Figure 1.6.: Resulting fragments from events

Credits: © ESA.

This growing congestion also leads to a yearly rise in events triggering collision avoidance procedures, not only for resident satellites but also for spacecraft that merely need to cross these congested regions. Consequently, while applying proper mitigation measures at the end of a satellite's operational life is fundamental, it is no longer sufficient to guarantee the long-term sustainability of the space environment. This highlights the urgent need for Active Debris Removal (ADR) missions and On-Orbit Servicing. In this context, capture systems play a fundamental role. In fact, over the last years, various architectures have been proposed to grapple and remove non-cooperative targets, ranging from flexible solutions (such as nets or harpoons) to rigid systems (such as robotic arms or clamping mechanisms).

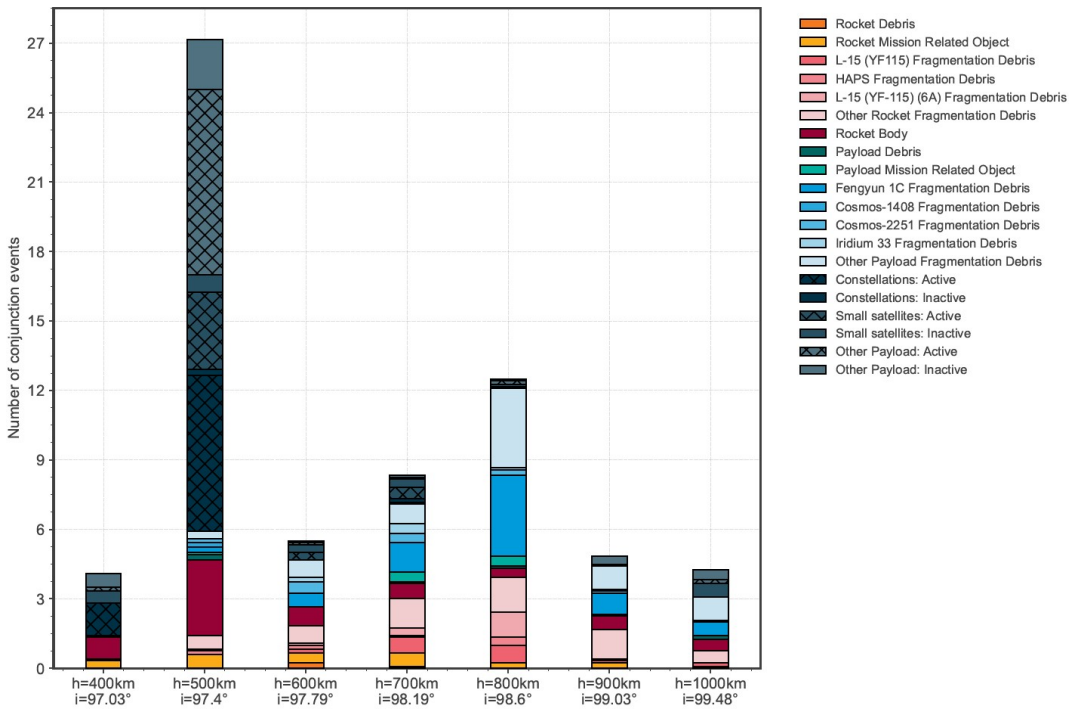


Figure 1.7.: Number of conjunction events

Credits: © ESA.

Overall, a strong space traffic coordination system is necessary to allow satellites to avoid both active spacecraft and non-maneuverable space debris. This situation raises an important question:

> *Will the risk become too high to consider the space environment safe and sustainable?*

While tools such as ESA's Health Index provide an objective diagnosis of risk, the solution is the evolution of recovery and disposal techniques. From the figures below it is clear that the space industry has begun transitioning toward proactive orbital management. The fact that controlled re-entries reached a record high in 2024 shows that mitigation strategies are not only theory. Safely removing satellites at the end of their life and developing active recovery mechanism are now essential operational standards. These measures are essential to safeguard space infrastructure and ensure the continuity of services that modern societies increasingly rely on.

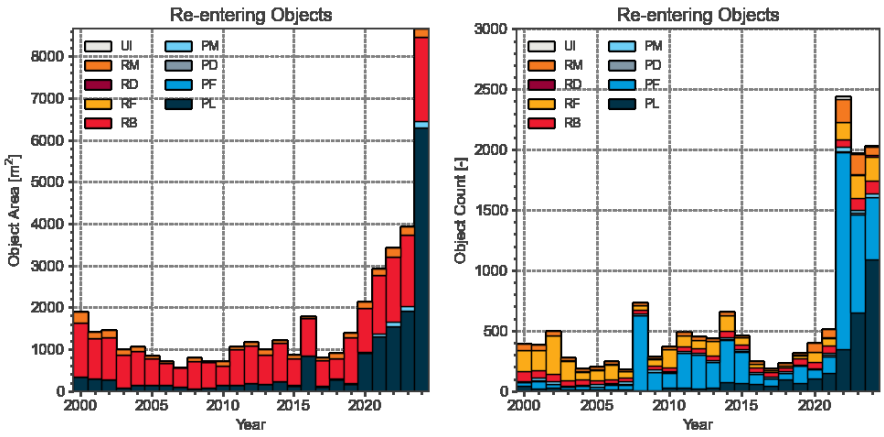


Figure 1.8.: Re - entering objects

Credits: © ESA.

1.2. Origami-based deployable structures

The origins of origami date back to ancient Japan and refers to the traditional art of creating three-dimensional shapes by folding a flat sheet, without using cuts or glue, in fact, the term results from the combination of the Japanese words:[10]

- **Ori** (折り) – folded
- **Kami** (紙) – paper

Although origami was originally linked to artistic and cultural practices, over time it has attracted increasing interest in scientific and engineering fields. This is mainly due to its ability to transform two-dimensional surfaces into compact, reconfigurable, and efficient three-dimensional structures. In fact, since aircraft and spacecraft have limited space and load capacity, the equipment they transport needs to be light and able to fold and unfold, the origami based deployable structures offer the best solution to ensure those requirements.[11]

1.2.1. Origami-Based Solutions for Space Systems

In aerospace engineering, origami inspired structures are commonly classified as rigid, membrane, or adaptive. Rigid origami is based on stiff panels connected by hinges and allows controlled and predictable deployment. Membrane origami uses flexible materials to achieve compact stowage and large deployed surfaces. Adaptive origami structures can change their shape or behavior during operation, improving stress distribution and flexibility.

International Space Station - ISS

International Space Station clearly reflects the main and most up-to-date use of origami-based mechanisms: solar panels and radiators. Solar panels are

deployed by unrolling from a base anchored to the ISS, while radiators are deployed using an accordion-like folding system.

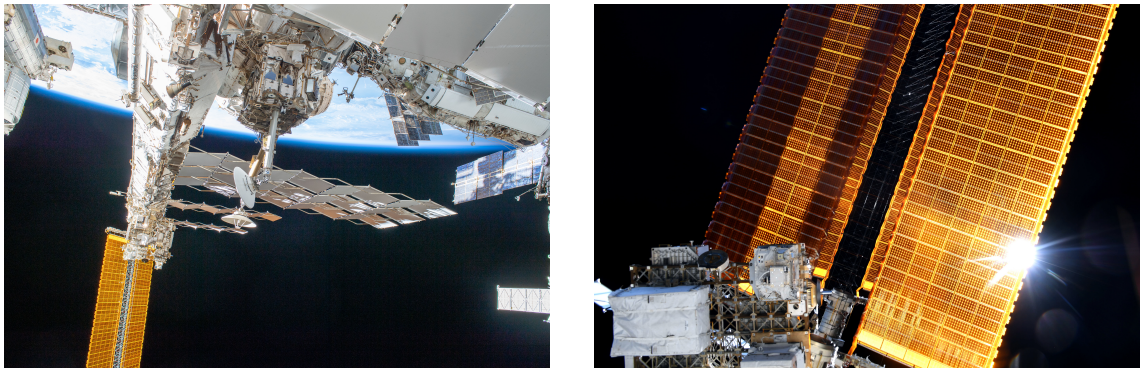


Figure 1.9.: ISS solar panels and radiators

Credits: © European Space Agency (ESA).

Starshade

The starshade is an innovative origami-inspired deployable system designed to support the direct imaging of exoplanets. It acts as an external occulter by suppressing stellar light and enabling the observation of an exoplanet's light. This approach allows a more accurate characterization of exoplanets, especially small and rocky ones located in the habitable zones of nearby stars. Due to the extreme brightness contrast, such observations would be impossible without effective starlight suppression made by starshade. To accomplish this task, a shield with a diameter of several tens of meters would be required, which could not be accommodated by any existing launch vehicle. For this reason, NASA is developing an origami-inspired mechanism composed of 24 petals, each approximately 8 m long, compactly folded around a central disk with a diameter of about 10 m.[12]

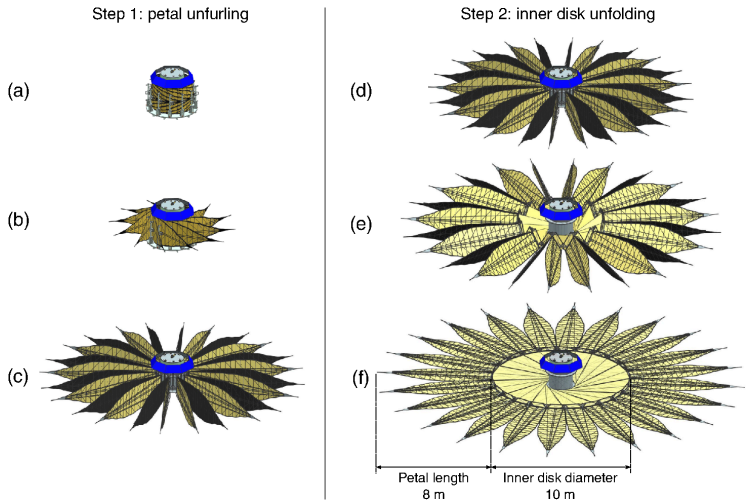


Figure 1.10.: Starshade unfolding system

Credits: [12].

Fuel bladder by Washington State University

This application can be considered niche, as origami mechanisms are used for the storage and transportation of liquid propellants. Research carried out in the United States in the 1980s and 1990s suggested that origami-based structures, thanks to their complex folding patterns, could exploit the mechanical behavior of folds to improve stress distribution and reduce the likelihood of material failure.[10]

Figure 1.11.: Fuel Bladder by WSU

Credits: © Washington State University [13].

2. Origami-based model for satellite recovery

This chapter presents the origami mechanism analyzed in this thesis and outlines the main findings of previous studies that served as reference milestones for the analyses carried out.

2.1. Origami - multimodel capture mechanism

A deployable mechanism made up of 32 pentagonal and hexagonal tiles is examined. These tiles are arranged into five branches connected to a central pentagon; the branches are identical, save for one which contains an extra external pentagonal tile. The capture mechanism proposed in this study is folded into a five-level petaloid configuration prior to launch and unfolds into a planar structure after release, as shown in Figure 2.1. During the capture phase, it can form either a soccer-like grasping configuration or a two- or three-finger grasp. The advantages of the mechanism include its modular architecture, reduced mass, and compact design.[14]

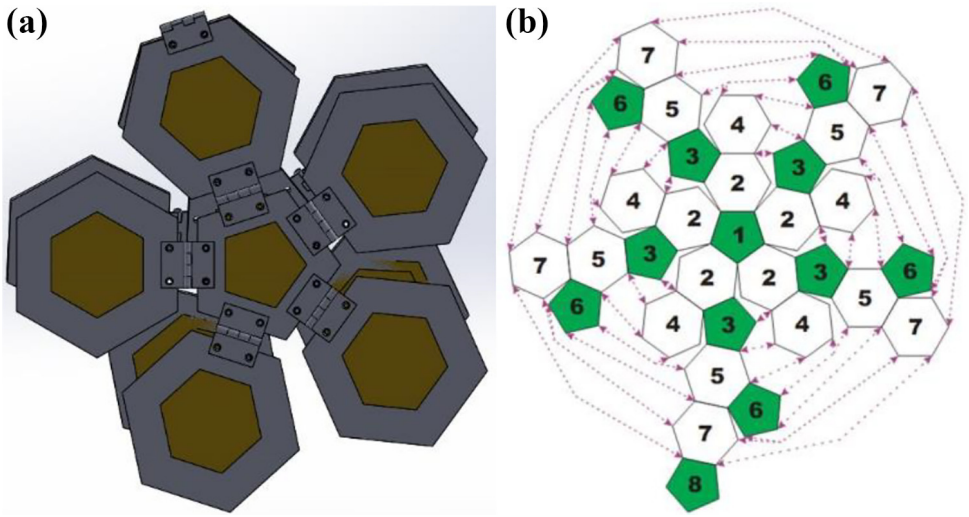


Figure 2.1.: The folded five-level planar structure and unfolded planar structure.

Credits: [14].

2.1.1. Deploying and Capture configuration

The tiles composing the mechanism are all designed as regular polygons with equal side lengths. During the deployment phase, all the tiles of the mechanism, initially arranged in the closed configuration shown in Figure 2.1, rotate by 180° about their respective reference hinges, reaching the fully deployed configuration.

For the capture of small space debris, with characteristic dimensions in the order of a few centimeters, a nearly spherical final configuration can be achieved by properly tuning the actuators (revolute hinges) so as to impose specific relative angles between adjacent tiles. In particular, the target spherical geometry is obtained by setting the following dihedral angles:

- between a regular pentagon and a regular hexagon: $142.62°$;
- between two adjacent regular hexagons: $138.19°$.

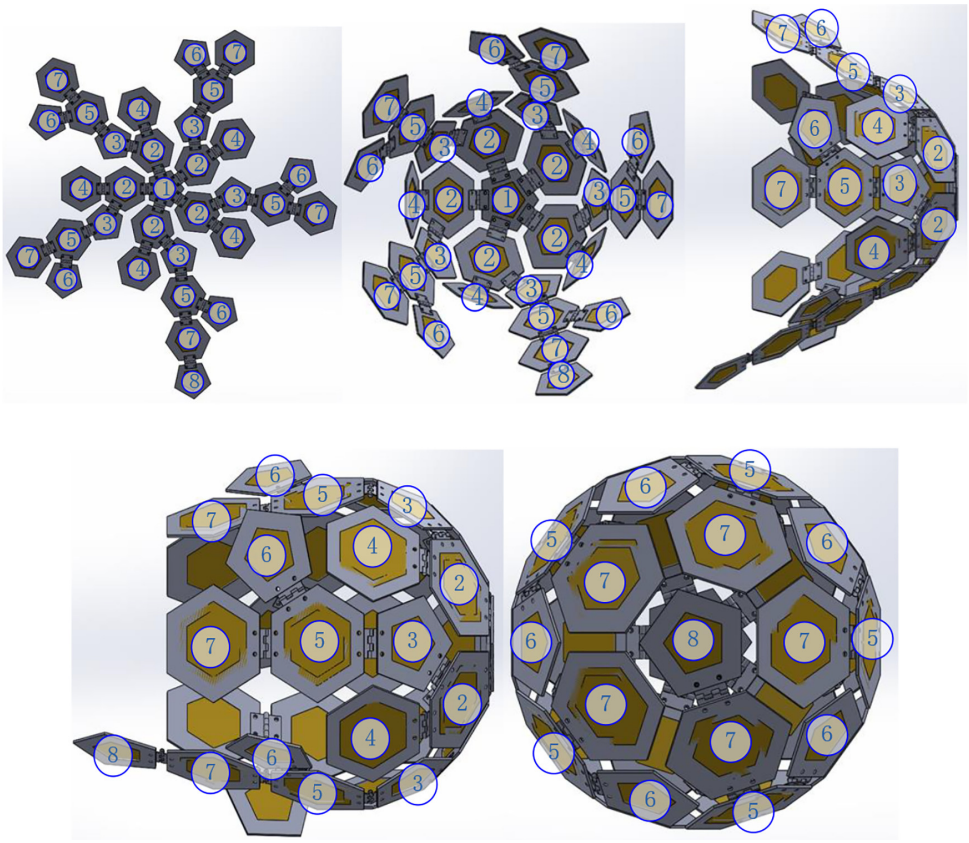


Figure 2.2.: The process of soccer-like configuration forming.

Credits: [14].

In the case of medium-size space debris, whose dimensions exceed those of the fully closed spherical configuration, a *three-finger caging configuration* is adopted for target capture. The unfolding planar structure of the capture mechanism can be divided into five distinct parts, referred to as *Finger 1* to *Finger 5*. During three-finger caging, two adjacent fingers together with a third, spatially separated finger form a semi-enclosed configuration, which enables the effective containment and stabilization of the space target.

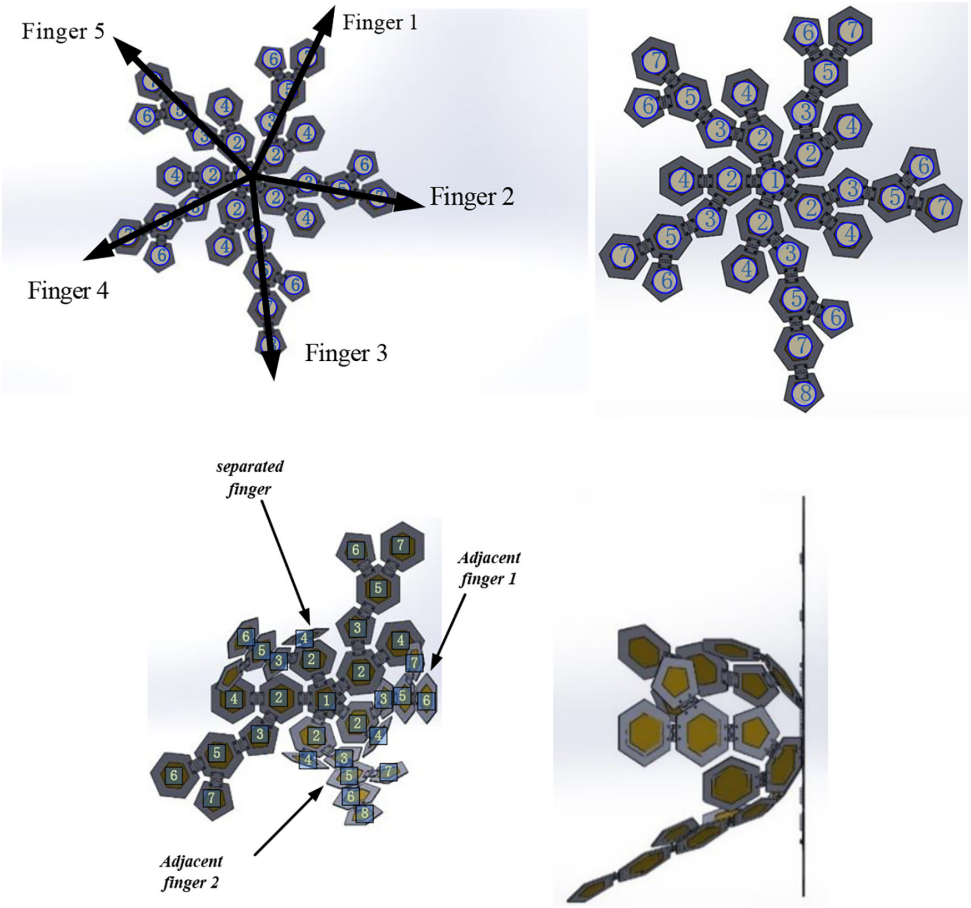


Figure 2.3.: The process of three-finger configuration forming.

Credits: [14].

2.2. From Previous Studies to Parametric Model Design

In this section, results obtained results from previous studies [15] and [16] conducted on this model are presented and discussed, as they represent the starting point of the work conducted. These results provide the theoretical and methodological foundations required to carry out a more extensive parametric study, aimed at extending the applicability of the mechanism to a wide range of operational scenarios. Therefore, in order to investigate the system's response to changes in design configurations, the research focused on the creation of a Multibody Dynamics (MBD) simulation model. The need to identify an optimized configuration of the origami mechanism, suitable for different possible applications, made it necessary to implement a parametric model. Unlike a static or purely deterministic simulation, the parametric approach allows the geometry to be independent from its nominal values, treating the most critical dimensions as design variables. In this context, Adams software (whose technical specifications and numerical integration algorithms will be discussed in detail in Chapter 4) was used not only as a solver, but also as an integrated geometric parametric modeling platform. To make the model adaptive, Design Variables (DV) were introduced within the Adams environment and self-executing command files were created in order

to import values using the same algorithm regardless of the characteristic measurements of the model under analysis. These variables allow the definition of the main geometric points of the tiles that form the mechanism (such as vertices and midpoints), along with the thickness of the components and their inertial properties, through interconnected mathematical expressions, as well as the analysis and identification of particular points in the graphs derived from the laws of motion imposed on the hinges. As shown in the Dialog Box below, each geometric parameter has been associated with a design variable, thus allowing the software to automatically regenerate the system topology during iteration cycles.

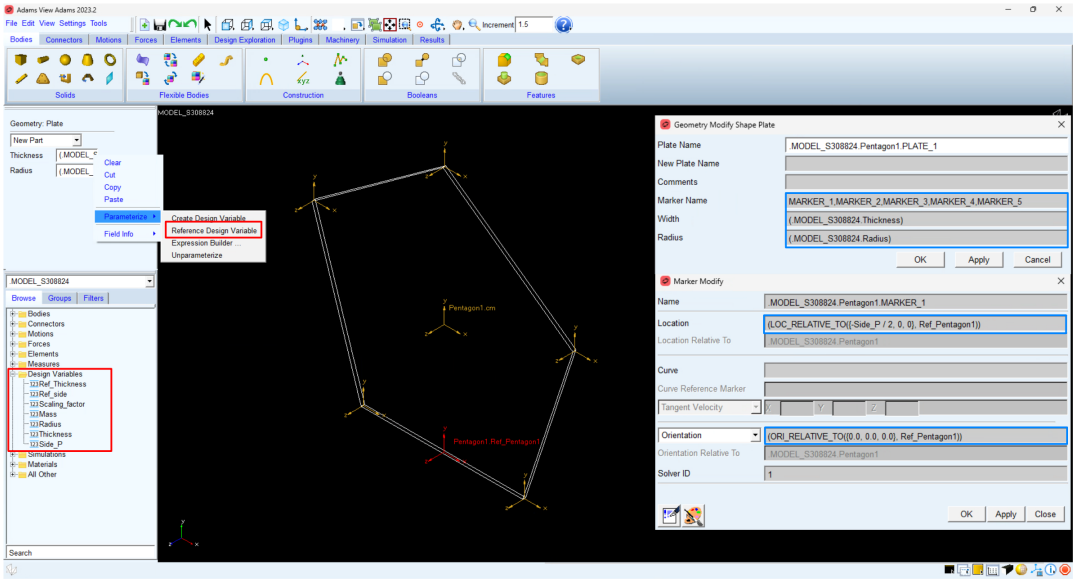


Figure 2.4.: Adams' main dialog boxes

Although the MBD model is currently undergoing preliminary validation and refinement, this automated capability already significantly reduces the time required for design iterations and minimizes the risk of manual remodeling errors. Ultimately, this robust parametric framework lays the groundwork for a future systematic exploration of the design space: once fully developed, it will enable rigorous optimization routines to identify the most performing kinematic and dynamic configurations for the targeted operational scenarios.

2.2.1. Configuration of the model

The initial configuration of the model descends from the study [16], which was considered a useful reference prototype for this parametric study. This analysis also provided the initial data used to define the friction forces acting on the joints, as well as the initial characteristics of the target. From the most recent analysis [15], all data and information concerning the `MOTIONS`, `SFORCES` acting on the joints, and related sensors were derived. All these elements were then adapted in order to be applicable to a different scale of the parametric model. The milestones that led to this work are presented in subsection below.

Chaser geometry

As discussed in Subsection 2.1.1 the origami mechanism consists of thirty-two tiles of equal length size and thickness, all connected by joints to produce the desired configuration. In the first analysis mentioned above, the measures are established as follows and, with regard to the length of the side, remain virtually unchanged in the second analysis.

- Side = 0.2 m
- Thickness = 0.01 m

Based on the modal analysis performed in the second study with a flexible plate model, the panel thickness was reduced to 4 mm through the use of laminated plates consisting of 32 CFRP plies.[15] Therefore, the following reference measures were chosen to be set in Adams to configure the geometry of the model:

- `Ref_side`
- `Ref_thickness`
- `Scaling_factor`

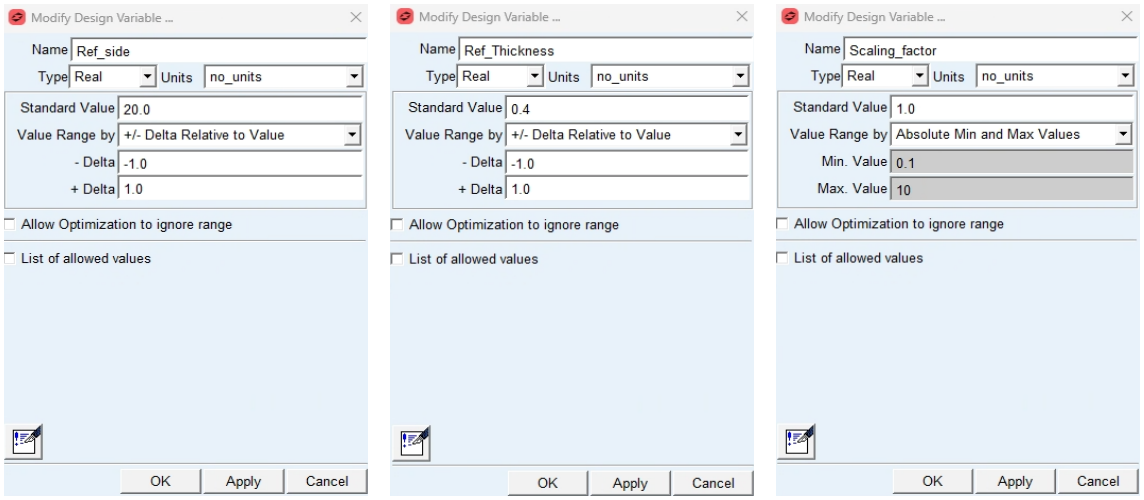


Figure 2.5.: Core design variables

As mentioned above, the previous models are intended as prototypes; therefore, the corresponding design variables were created as dimensionless quantities in order to be used as reference parameters for the parameterization process driven by a `scaling factor`. This scaling parameter represents the ratio between the side length of the parametric origami mechanism under investigation and the one used in the previous analyses. A range of values was selected to allow the creation of models up to ten times smaller and up to ten times larger than the reference configuration.
Once this parameter has been defined, it is possible to define the surface area and volume occupied by the configuration of the mechanism analyzed: the soccer ball configuration. Looking at the shape and the geometric composition, it is possible to define this configuration as a truncated icosahedron, and therefore the geometric relationships for the parametric model are defined as follows [17]:

- *Characteristic Edge Length* (a)
  The primary scale of the solid, defined as:

$$a = \text{scaling\_factor} \cdot \text{Ref\_side}$$

- *Circumradius* (R)
  The radius of the sphere passing through all 60 vertices:

$$R = \frac{a}{4}\sqrt{58 + 18\sqrt{5}} \approx 2.478\, a$$

- *Midradius* (ρ)
  The radius of the sphere tangent to all 90 edges:

$$\rho = \frac{3}{4}a(1 + \sqrt{5}) \approx 2.427\, a$$

- *Surface Area* (A)
  The total area of the 12 pentagonal and 20 hexagonal faces:

$$A = 3\left(10\sqrt{3} + \sqrt{5(5 + 2\sqrt{5})}\right) a^2 \approx 72.607\, a^2$$

- *Volume* (V)
  The total space enclosed by the truncated icosahedron:

$$V = \frac{1}{4}(125 + 43\sqrt{5})a^3 \approx 55.288\, a^3$$

Target geometry

The same approach expressed in Subsection 2.2.1 was applied to the target, so each dimentional variable expressed for model definition was taken and made dimensionless in order to enable its use in the parametrized ones. A cubic body with the following geometric and inertial characteristics was firstly considered: [1]:

[1] Derived from typical values for nanosatellites with similar dimensions

Table 2.1.: Geometric parameters and initial conditions of TARGET

Quantity	x	y	z
Side length L $[cm]$		[25]	
Moment of inertia I $[kg\,m^2]$		[0.216]	
Initial position r_0 $[m]$	[-1.000]	[1.125]	[20.125]
Initial velocity v_0 $[m/s]$	[-0.20]	[-0.10]	[0.50]
Initial angular velocity ω_0 $[deg/s]$	[10]	[10]	[10]

where I_i denote the moments of inertia, x_0, y_0, z_0 the initial position components, $v_{i,0}$ the initial translational velocity components, and $\omega_{i,0}$ the initial angular velocity components, all defined along the axes of the model $i = x, y, z$.

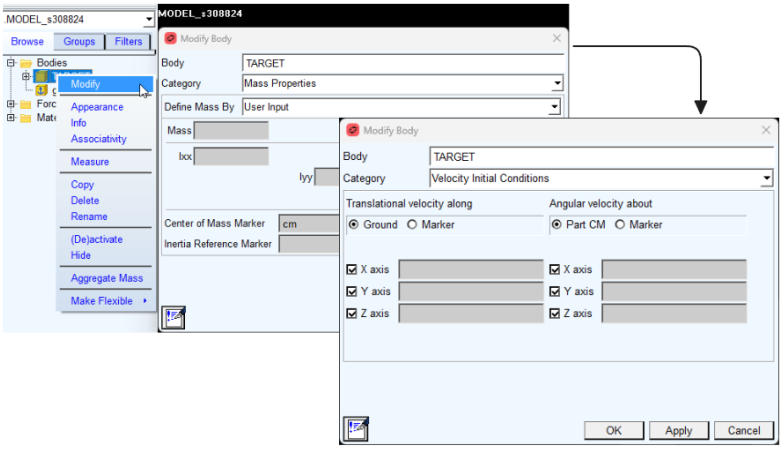


Figure 2.6.: TARGET's initial conditions

To enhance the model's versatility, an additional parameterization level has been implemented. Users can define the simulation target directly via the general `.cmd` file, selecting between cube and sphere target's shape automatically triggering a topological reconfiguration.

```
! ========================================
! TARGET GEOMETRY SELECTION
! ========================================

! Uncomment the desired target configuration:
file command read file="Commands/TARGET_CUBE.cmd"

! file command read file="Commands/TARGET_SPHERE.cmd"
! ========================================
```

Listing 2.1: Target change commands in the model command file (.cmd)

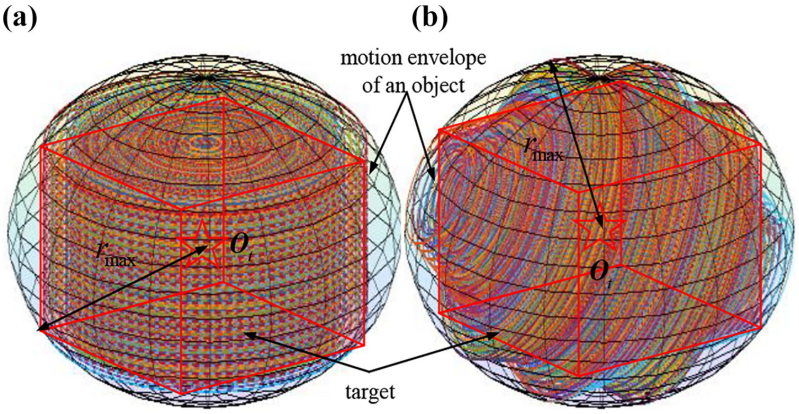


Figure 2.7.: The motion envelope for different measured initial angular velocities

Credits: [14].

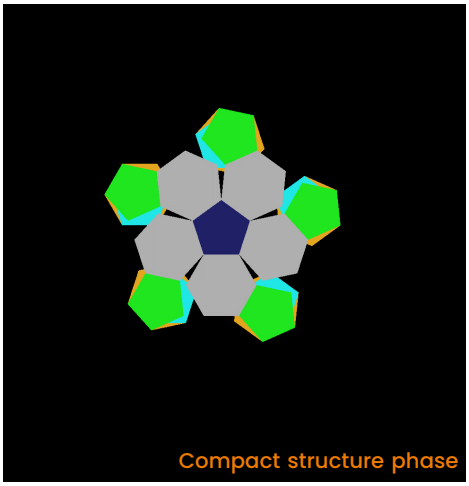
Integration of Validated Sensors into the Parametric Model

The deployment and capture system together with of adopted SENSORS required to fulfill them[15] has been identified as the most feasible solution for the origami-based model under investigation, assuming the soccer-like configuration for capture phase 2.2. In this subsection, the changes were minimal and above all concerned the replacement of "raw" numerical values with design variables. Below, the deployment and capture phases with the most relevant sensors are illustrated.

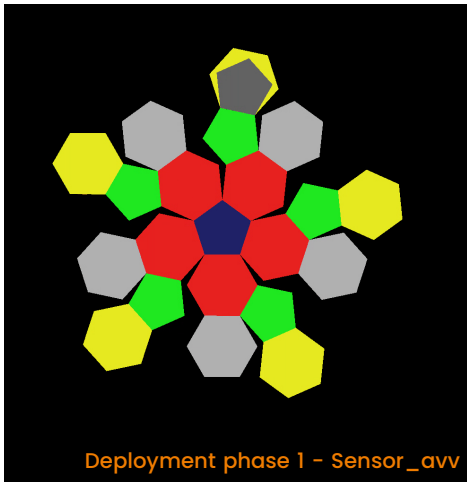
Deployment phase

Deploying consist of three phases :

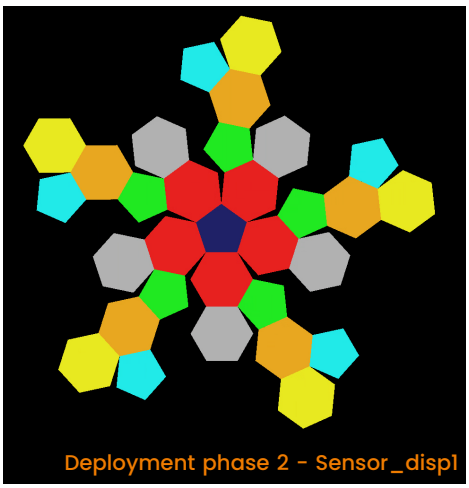
- Deploying phase one: from the compact configuration, the hexagons connected to the fixed ones of first level and to the fixed pentagons of the second level are deployed. They are identified by:
  - `Hexagon2`
  - `Hexagon3`
- Deploying phase two: From deployed phase one, the hexagons and pentagons connected to level two `Hexagon3` unfold:
  - `Pentagon3`
  - `Hexagon4`
- Deploying phase three: Deployment of the single pentagon present on the `ne` arm:
  - `Pentagon4`



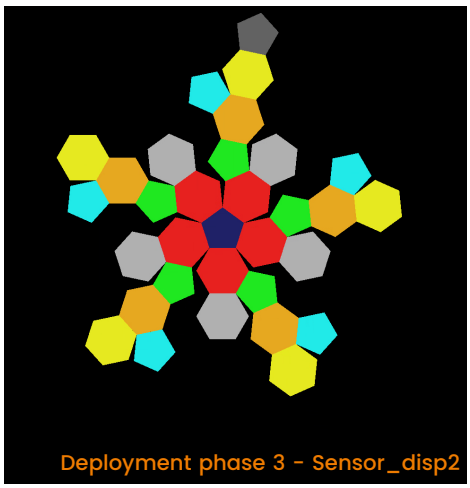
(a) Compact phase 0



(b) Deployment phase 1



(c) Deployment phase 2



(d) Deployment phase 3

Figure 2.8.: Deployment phases

The motion laws set for the deployment phases are the same for each tile and level. In fact, a rotation of a straight angle around the hinge axis is required in a specific time set as a design variable.

$$\theta(t) = \text{STEP5}\,(t,\, 0,\, \theta_{start},\, \textit{time to deploy},\, \theta_{straight}) \tag{2.1}$$

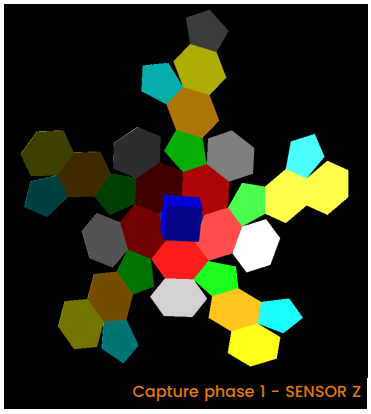
where θ_{start} and $\theta_{straight}$ represent the angle in compact phase and the final angle in deployed phase.

Capture phase

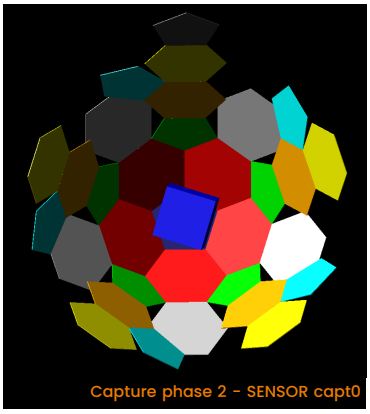
As [15], the capture phase is structured into five distinct stages:

- Capture phase one: from the fully deployed configuration, the hexagons connected to the main one (first level) start to close. They are identified by:
  - `Hexagon 1` - Angle: 37.38 deg

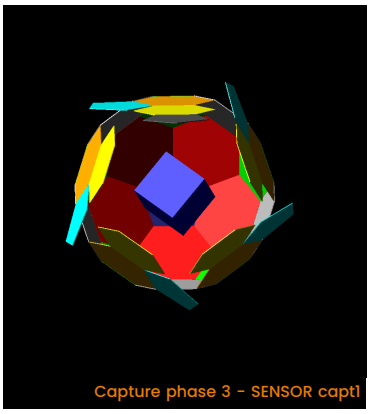
- Capture phase two: From capture phase one, the pentagons connected to level one `Hexagon1` fold:
  - `Pentagon2` - Angle 37.38 deg
- Capture phase three: Folding of the hexagons connected to the `Hexagon1` and `Pentagon2` (Third level):
  - `Hexagon2` - Angle 41.81 deg
  - `Hexagon3` - Angle 37.38 deg
- Capture phase four: Folding of the pentagons and hexagons connected to `Hexagon3`:
  - `Pentagon3` - Angle 37.38 deg
  - `Hexagon4` - Angle 41.81 deg
- Capture phase five: Folding of the single pentagon present on the `ne` arm:
  - `Pentagon4` - Angle 37.38 deg



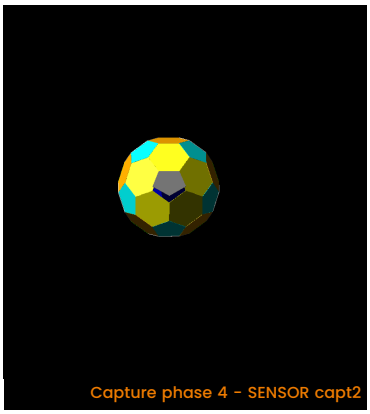
(a) Capture phase 1



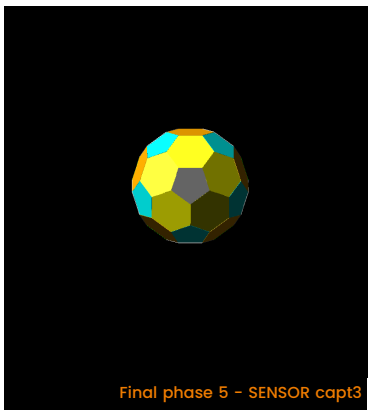
(b) Capture phase 2



(c) Capture phase 3



(d) Capture phase 4



(e) Final capture - Phase 5

Figure 2.9.: Capture phases

For the capture phases the laws of motion follow the same structure as Eq. 2.1, but with variations depending on the pair of polygons involved in the capture phase.

- `Hexagon - Pentagon`

$$\theta(t) = \text{STEP5}\,(t,\ 0,\ \theta_{straight},\ \textit{time to capture},\ \texttt{Angle\_btw\_p\_e}) \tag{2.2}$$

- `Hexagon - Hexagon`

$$\theta(t) = \text{STEP5}\,(t,\ 0,\ \theta_{straight},\ \textit{time to capture},\ \texttt{Angle\_btw\_e\_e}) \tag{2.3}$$

Laws of `MOTION`

Based on the most recent study [15] the laws of motion applied to joints were analyzed. A law of motion can be described as a mathematical function that imposes the desired law on a moving part. For the parametric analysis ongoing, since motion laws are independent of geometric or inertial factor it was decided to maintain mostly the same approach in terms of both functional formulation and sensor configuration. Only minor modifications were introduced in order to make every raw value inserted before referred to a design variable and in standardizing the behavior of each plate by adopting uniform conventions for measures.

```
IF(SENVAL(SENSOR_capt1):
    0,
    IF(SENVAL(SENSOR_avv):
        0,
        0,
        STEP5(time, SENVAL(SENSOR_avv), 0.0d →Start_angle,
        SENVAL(SENSOR_avv)+25 →Time_Disp,180.0d →Straight_Angle)),
    STEP5(time, SENVAL(SENSOR_capt1), 180.0d →Straight_Angle,
    SENVAL(SENSOR_capt1)+25→Time_Capture, 217.38d →(Straight_angle − Angle_btw_e_e)))
```

Listing 2.2: MOTION EXAMPLE

This is provided only as an illustrative example for the `e1_e2_ne` tiles; the complete relations and their trends will be investigated later in Section 4.3.5.

`SFORCES`

To accurately simulate deploying and capture phases in a dynamic model, the imposed laws of motions have to be replaced by functions capable to represent actuators behavior. Given the presence of `Revolute Joints`, it has been opted to insert a Single - Component Torque force allowing the application of a rotational force to either one part or two about a specified axis. For the starting formulation of these Single-Component Torque elements the same approach adopted in [15] was followed, so the moments produced on the hinges from kinematic analysis were analyzed and once that was done `STEP5` functions were used to replicate their temporal trends in the `SFORCES` formulation. Furthermore, as an initial approach, the use of PD controllers was also maintained in phases where the position had to be kept constant.

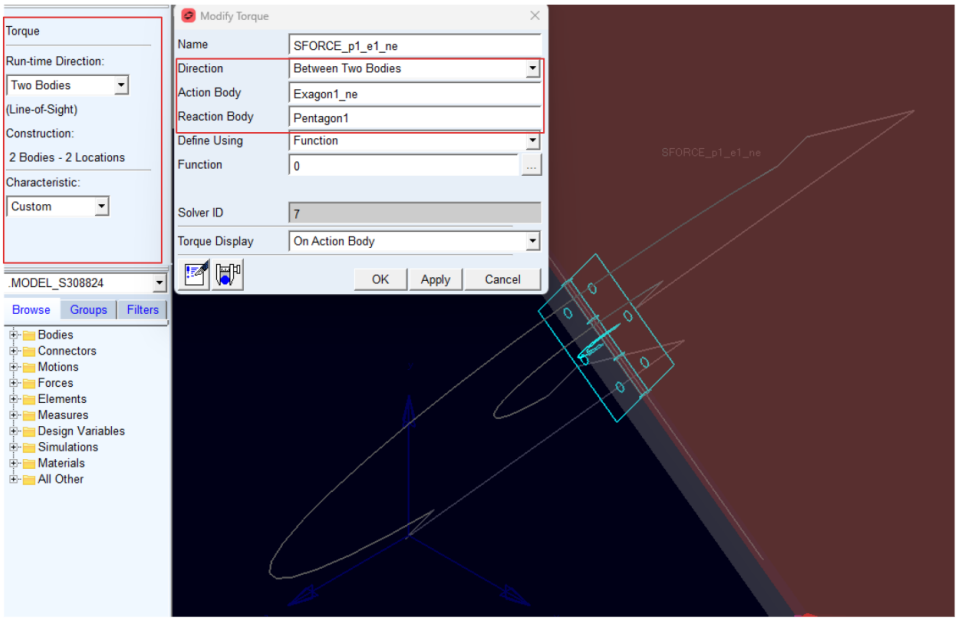


Figure 2.10.: Single - Component Torque definition in Adams

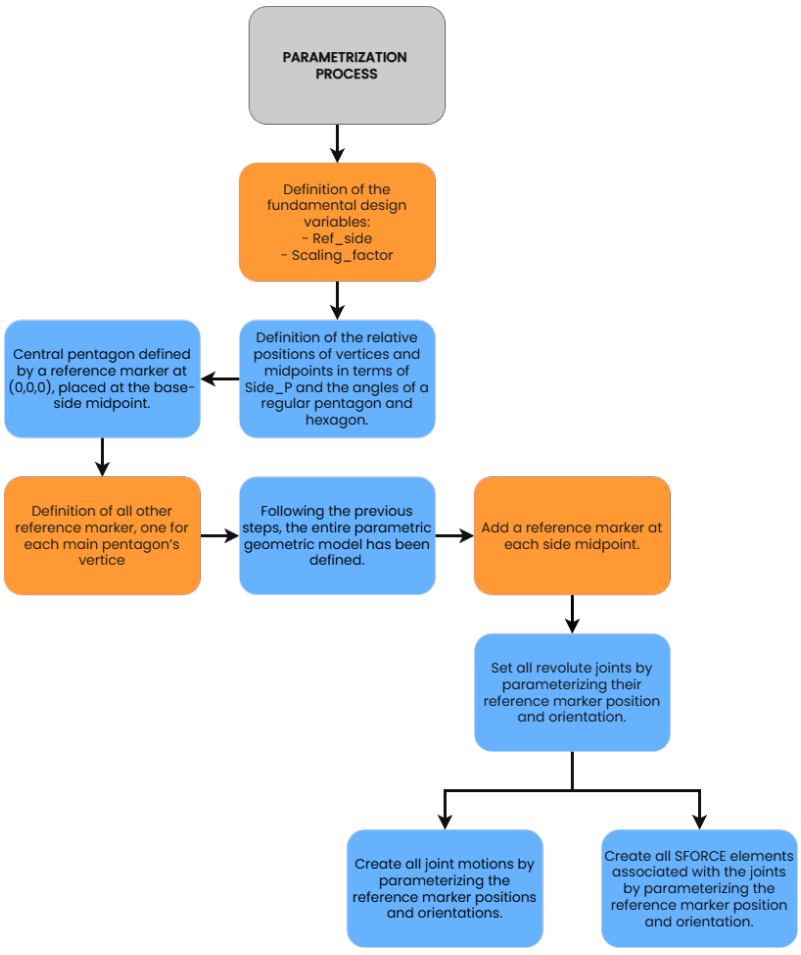
As 2.2, a preliminary illustration of the initial modifications implemented for model parameterization is reported, a detailed investigation is addressed to Chapter 4.3.6.

```
IF(.JOINT_p1_e1_az - 37.38 →Angle_btw_p_e:
              STEP5(time,SENVAL(.SENSOR_z),−0.0154 →SENVAL(.SENSOR_p1_e1_ne_c),
              SENVAL(.SENSOR_z) + 1.2 →.MAX_p1e1ne_time,0.2448 →MAX_p1e1ne_val)+
              ...,
              0,
              −0.01 →K_STEP_PD*(.JOINT_p1_e1_az-37.38 →Angle_btw_p_e)
              −0.4 →C_STEP_PD*.JOINT_p1_e1_wz),
```

Listing 2.3: SFORCE EXAMPLE

3. Preliminary setup of the parametric model

The transition from the model with fixed dimensions to a fully parametric model was driven by the necessity for scalability, kinematic robustness and design optimization. A static model, characterized by a single set of geometric values, even if optimized, is extremely sensitive to any geometric or kinematic variation, causing problems in analyses performed on it. The adoption of a parametric model addresses these limitations by using a dynamic system of geometric and kinematic relationships, ensuring consistency across design changes. parameterization is therefore fundamental to the process of optimization and final validation of the model, as it offers the possibility of generating multiple variants of the same model, allowing iterative analyses to be performed to find optimal results according to each constraint and objective to be analyzed. The following block diagram illustrates the bottom-up process adopted for the parameterization of the origami model under investigation, each block will be analyzed in the following sections.



3.1. Pentagonal and Hexagonal Tiles

To parameterize the model from a geometrical point of view, it was necessary to define a method to parameterize both hexagonal and pentagonal tiles. The approach used is based on the trigonometric properties of regular polygons, for which it is possible to express the coordinates of the vertices as functions of their characteristic angles and a reference dimension. This formulation ensures a unique definition of the tile geometry, while maintaining full control over the spatial arrangement of the elements. In addition to the vertex coordinates, the midpoints of each edge of the polygon were also parameterized. These points are essential for the definition of joints and reference markers used in the multibody model. The vertices of the regular pentagonal and hexagonal tiles are analytically defined as:

$$
\begin{aligned}
V_1 &= (0, 0, 0) \\
V_2 &= (L, 0, 0) \\
V_3 &= (L + L\cos 72^\circ,\ L\sin 72^\circ,\ 0) \\
V_4 &= (L/2,\ L(\cos 36^\circ + 1)/(2\sin 36^\circ),\ 0) \\
V_5 &= (-L\cos 72^\circ,\ L\sin 72^\circ,\ 0)
\end{aligned}
$$

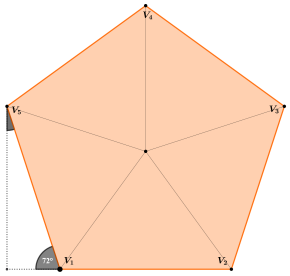


Figure 3.1.: Analytical definition of the vertices of the regular pentagonal tile

```
(LOC_RELATIVE_TO({0.0, 0.0, 0.0}, Ref_pentagon1_ne))
(LOC_RELATIVE_TO({Side_P, 0.0, 0.0}, Ref_pentagon1_ne))
(LOC_RELATIVE_TO({Side_P + Side_P*COS(72), Side_P*SIN(72), 0.0}, Ref_pentagon1_ne))
(LOC_RELATIVE_TO({Side_P/2, Side_P/(2*SIN(36))*(COS(36)+1), 0.0}, Ref_pentagon1_ne))
(LOC_RELATIVE_TO({-Side_P*COS(72), Side_P*SIN(72), 0.0}, Ref_pentagon1_ne))
```

Listing 3.1: Adams implementation of the pentagonal vertex parameterization

$$
\begin{aligned}
V_1 &= (0,\ 0,\ 0) \\
V_2 &= (L,\ 0,\ 0) \\
V_3 &= (L + L\sin(30^\circ),\ L\sin(60^\circ),\ 0) \\
V_4 &= (L,\ 2 \cdot L\sin(60^\circ),\ 0) \\
V_5 &= (0,\ 2 \cdot L\sin(60^\circ),\ 0) \\
V_6 &= (-L\sin(30^\circ),\ L\sin(60^\circ),\ 0)
\end{aligned}
$$

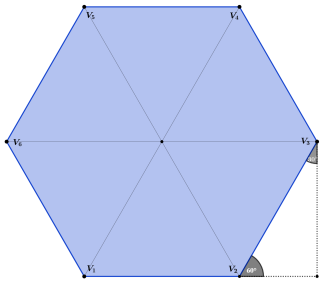


Figure 3.2.: Definizione analitica dei vertici della cella esagonale regolare

After completing the parameterization of the main pentagon, the connections with the tiles belonging to the five branches of the origami mechanism were established. Reference markers were created on the main pentagon to correctly

identify the position of the vertices of the first-level tiles of the other branches. Furthermore, a marker was positioned at the midpoint of each side of the pentagon to model the joint connecting the central structure to the first-level hexagonal tiles.

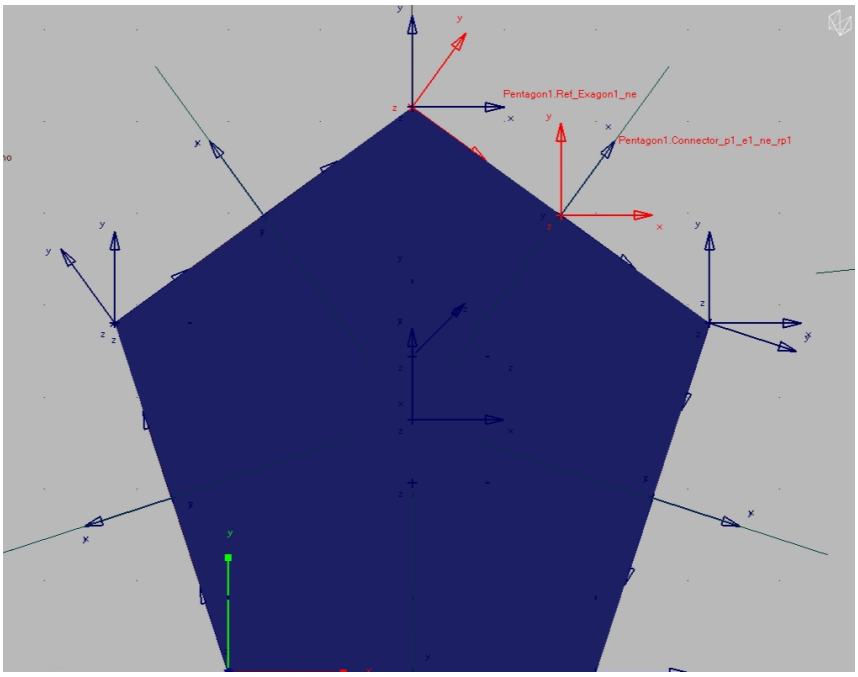


Figure 3.3.: Main pentagon with reference markers and first-level tile connections

```
Ref : (LOC_RELATIVE_TO({0.0, 0.0, 0.0}, MARKER_4))
324.0, 0.0, 0.0
```

Listing 3.2: Reference marker definition

```
Connect : (LOC_RELATIVE_TO({(Side_P * COS(72) + Side_P / 2) / 2,
(Side_P / (2 * SIN(36)) * (COS(36) + 1) + Side_P * SIN(72)) / 2, 0},
Ref_Pentagon1))
(ORI_RELATIVE_TO({0.0, 0.0, 0.0}, Ref_Pentagon1))
```

Listing 3.3: Connection marker definition

The same methodology was adopted for each tile of every arm, and, in order to complete the geometric model ensuring a proper actuator operation, a spacing (offset) between adjacent tiles was introduced.

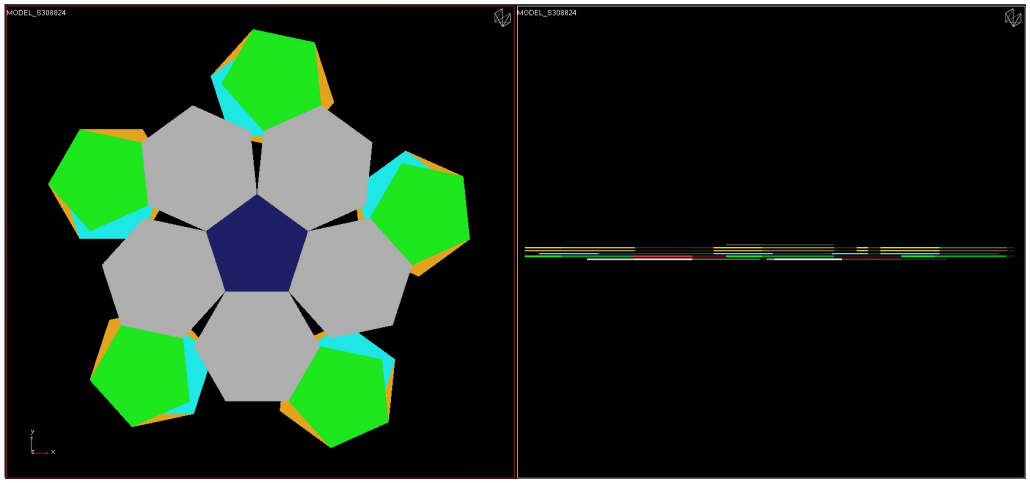


Figure 3.4.: Views of origami mechanism

For the parameterization process, it was essential to create functions whereby variations in the chaser's characteristic dimension would correspond in changes of the all other geometric dimensions.

3.2. Thickness parameterization

Once the `Scaling factor` has been defined as a function of the characteristic side length of the model, all other geometric dimensions must be consistently defined. To create the `PLATE` element in Adams, besides the edge curvature radius, the other fundamental parameter is thickness. It was not feasible to find a constant thickness suitable for each characteristic dimension, as this design choice would have led either to oversizing, with a consequent unsustainable mass, or to undersizing, causing problems in terms of flexural stiffness. Therefore, two design approaches were evaluated:

- Geometric scaling ($t \propto$ Scaling_factor).
- Scaling with flexural stiffness, to grant uniform structural behavior for each set dimension ($t \propto$ Scaling_factor^n).

Although the first hypothesis ($t \propto L$) would have simplified the model setup definition, it yields a relative deflection w/L that scales linearly with the dimension L. Consequently, scaling the model up would result in an underdeveloped thickness (leading to excessively flexible plates under their own weight), while scaling it down would result in an oversized thickness (leading to excessively heavy structures). For these reasons, the second approach was preferred.

3.2.1. Scaling with flexural stiffness: derived relationships

Hypothesis: Rectangular plate

For a rectangular cross-section with width L and thickness t, the moment of inertia J per unit width is proportional to t^3. Using the governing equations

of plate, the bending deformation w is related to the distributed load $q(x, y)$. In Kirchhoff's plate theory, the bending stiffness D is defined as:

$$D = \frac{Et^3}{12(1-\nu^2)} \tag{3.1}$$

Assuming the material properties (Young's modulus E and Poisson's ratio ν) as constant, the bending stiffness scales with the cube of the thickness ($D \propto t^3$). The governing differential equation for the plate deflection w is [18]:

$$D\left(\frac{\partial^4 w}{\partial x^4} + 2\,\frac{\partial^4 w}{\partial x^2\,\partial y^2} + \frac{\partial^4 w}{\partial y^4}\right) = D\,\nabla^4 w = q(x,y)_z \tag{3.2}$$

To understand how the parameters scale, the proportionality of the terms was analyzed. Assuming that the dominant distributed load q is the plate's self-weight, it is directly proportional to the thickness ($q = \rho \cdot g \cdot t \implies q \propto t$). Furthermore, since the operator ∇^4 involves fourth-order spatial derivatives, it dimensionally scales with $1/L^4$. This leads to the following proportionalities:

$$q \propto t, \qquad D \propto t^3, \qquad \nabla^4 w \propto \frac{q}{D}, \qquad w \propto \frac{q\,L^4}{D} \tag{3.3}$$

Substituting q and D with their thickness dependencies, the deflection scales as:

$$w \propto \frac{t\,L^4}{t^3} \propto \frac{L^4}{t^2} \tag{3.4}$$

To ensure uniform structural behavior, the principle of geometric similarity is applied to the deformation. This requires the relative deflection to remain constant ($w/L = \text{const}$), meaning the absolute deflection w must scale linearly with the characteristic length L ($w \propto L$). By imposing this condition, the relationship between the plate side length and its thickness becomes:

$$w \propto L \implies \frac{L^4}{t^2} \propto L \implies t^2 \propto L^3 \tag{3.5}$$

$$\boxed{t \propto L^{\frac{3}{2}}} \tag{3.6}$$

This derived relationship was implemented in the simulation environment to dynamically adjust the thickness.

```
variable_name = .MODEL_V3.Thickness
real_value = (.MODEL_V3.Ref_Thickness * (.MODEL_V3.Scaling_factor**(3 / 2)) cm)
```

Listing 3.4: Thickness definition in Adams

3.3. Geometric Accommodation of Finite Thickness

3.3.1. Theoretical Background and Problem Definition

Once the panel thickness is defined, one of the main challenges for a non-ideal origami structure is to properly accommodate this thickness. The ideal origami

models assume zero thickness, while real structures may suffer from deploying or capture problems and from self-interactions if thickness is not considered properly. In fact, failing to account for thickness in deployable space mechanisms could severely compromise their kinematics; thus, neglecting this parameter during the multibody modeling phase would lead to:

- geometric conflicts
  - panel interpenetration during folding
  - incomplete closure
  - solver instability

For this reason, thickness accommodation represents a fundamental design requirement, as reported in [19], the axis-shifting technique was therefore applied to the model.

3.3.2. Implementation of Axis Shift and Spacing Strategy

To overcome the geometric issues introduced above, a design variable called `Spacing` was defined in the Adams environment. This variable shifts the markers of revolute joints outward with respect to the panel edges, preventing collisions during panel rotation during deployment and capture phases. If the spacing is not sufficient the rotating corners of panels may collide resulting in kinematic locking and numerical instability when `CONTACTS` are considered between plates.

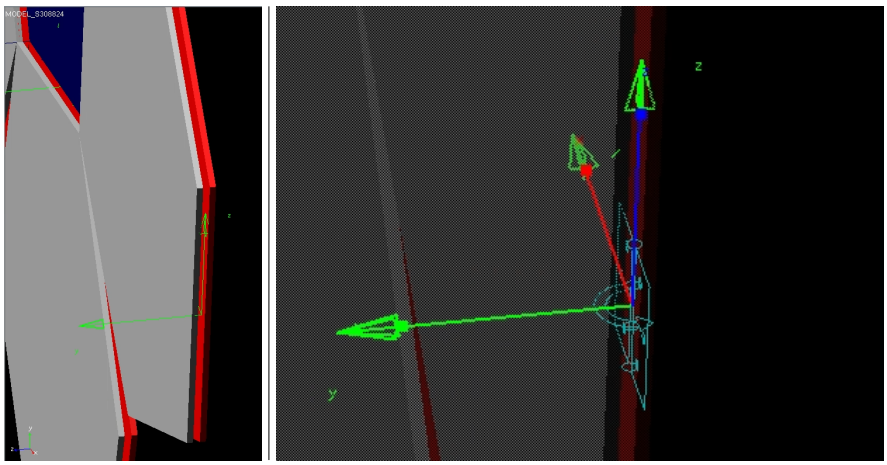


Figure 3.5.: Spacing and joint position

3.3.3. Design Choice

Given the preliminary nature of the model under investigation and the safety-critical role of the deployment mechanism, a conservative spacing strategy was adopted. Instead of determining the minimum theoretical clearance, the inter-panel gap was conservatively set equal to the panel thickness:

$$\texttt{Spacing} = \texttt{thickness}$$

This choice ensures a robust kinematic/dynamic behavior throughout the entire folding motion and provides sufficient space for the integration of mechanical hinge components in future design iterations. By prioritizing geometric robustness over maximum compactness, the model enables a reliable validation of the multibody simulation and the associated deployment sequence.

3.4. Effect of Parametrized Thickness on the Chaser Physical Quantities

3.4.1. Mass scaling relationship

Established the non-linear thickness parameterization ($t \propto L^{\frac{3}{2}}$), it is necessary to evaluate how this design choice impacts the overall mass of the model. In a standard geometric scaling approach, the volume (and consequently the mass, assuming constant density ρ) would scale with the cube of the characteristic length (L^3), but with this approach that property is not valid anymore. The mass M of the plate is proportional to its volume V, which is the product of its area A and its thickness t. Therefore, the mass proportionality can be derived as follows:

$$M = \rho \cdot V = \rho \cdot (A \cdot t) \implies M \propto L^2 \cdot t \tag{3.7}$$

Substituting the previously derived thickness relationship ($t \propto L^{\frac{3}{2}}$):

$$M \propto L^2 \cdot L^{\frac{3}{2}} = L^{2+\frac{3}{2}} = L^{\frac{7}{2}} \tag{3.8}$$

$$\boxed{M \propto L^{3.5}} \tag{3.9}$$

This result highlights that the mass grows significantly faster than a purely geometric scaling (L^3). This is the expected trade-off to maintain the required flexural stiffness at larger scales.

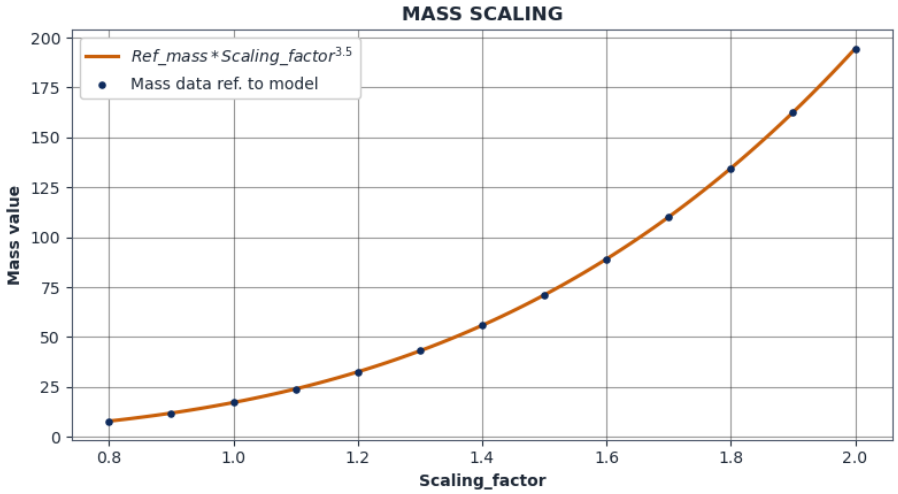


Figure 3.6.: Mass Scaling

3.4.2. Moment of inertia scaling

In multibody dynamic simulations, such as those performed in Adams, the mass moment of inertia dictates the rotational dynamic response of the components. The moment of inertia I of a body is generally proportional to its mass M multiplied by the square of a characteristic distance r from the axis of rotation. Since any planar characteristic distance r scales linearly with the characteristic length L, the basic proportionality is:

$$I \propto M \cdot r^2 \implies I \propto M \cdot L^2 \tag{3.10}$$

By substituting the mass scaling relationship ($M \propto L^{\frac{7}{2}}$) derived in the previous section, the scaling law for the mass moment of inertia becomes:

$$I \propto L^{\frac{7}{2}} \cdot L^2 = L^{\frac{7}{2}+2} = L^{\frac{11}{2}} \tag{3.11}$$

$$\boxed{I \propto L^{5.5}} \tag{3.12}$$

Compared to a standard geometric scaling (where $I \propto L^5$), the inertia here increases more rapidly with the model size. This highly non-linear relationship must be carefully accounted for in the model's parameterization, as it drastically affects the natural frequencies and the dynamic forces during the simulation.

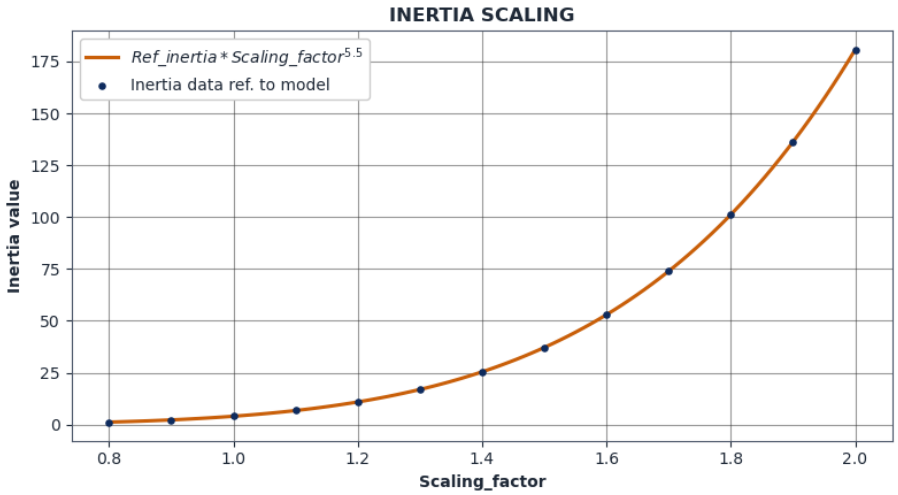


Figure 3.7.: Inertia Scaling

Once these relationships for mass and inertia have been defined, always taking into account the fundamental relationship that characterizes thickness, a range of values can be found to scale the gains of the PD controls used in the model. As a preliminary assumption, the scaling can be considered with an exponent included between the exponents obtained from the analyses of mass and inertia.

$$K, C \rightarrow K \cdot \texttt{scaling\_factor}^{\gamma},\ K \cdot \texttt{scaling\_factor}^{\gamma}, \qquad \gamma \in [3.5, 5.5]$$

3.5. Joint modeling

As defined in the previous sections, a marker was created and positioned at the midpoint of each tile that had to be mutually connected with the tile of the subsequent level. Each marker follows the same naming convention used for the tiles definition in Chapter 2:

```
Connector_(first tile)_(second tile)_relative branch_relative tile
```

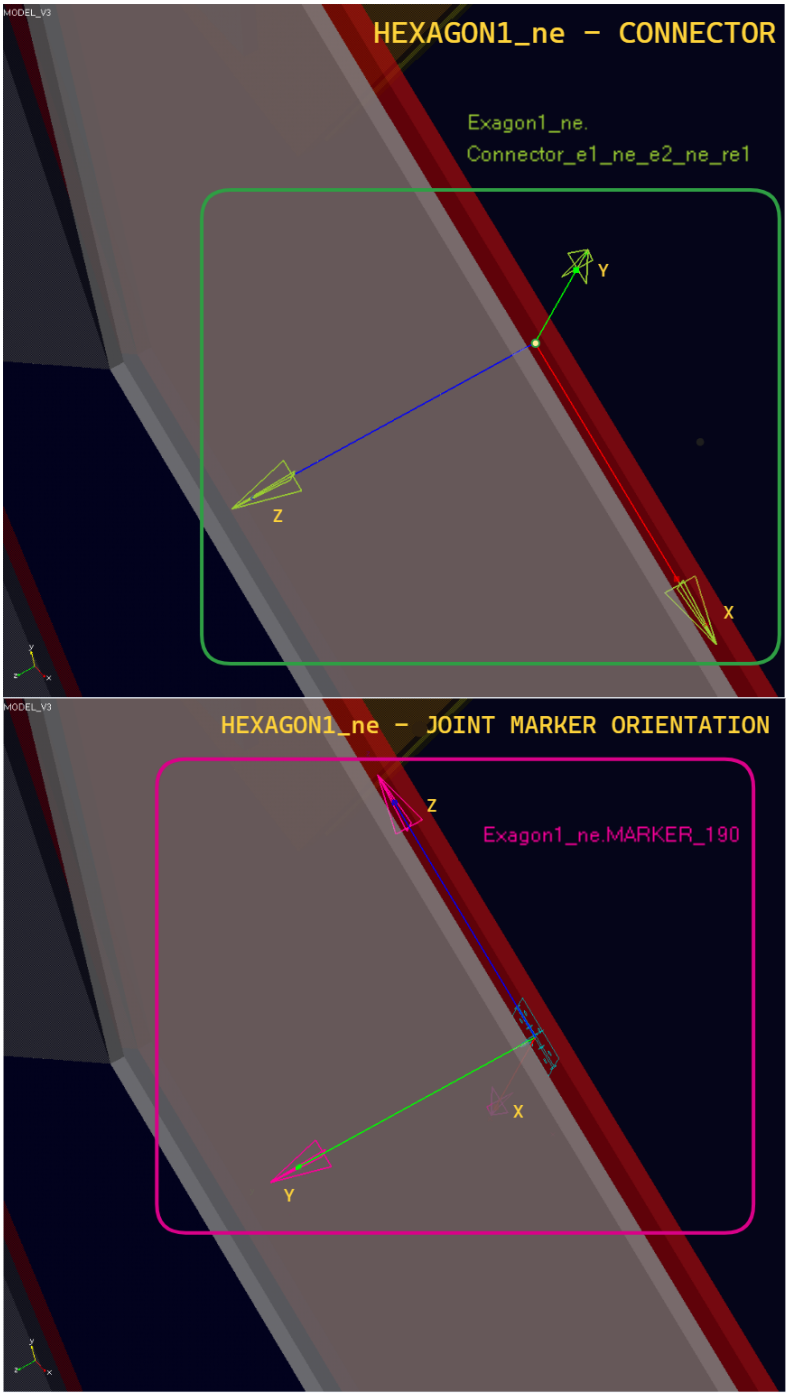


Figure 3.8.: Connector and relative joint marker orientation

Once the connectors were defined on the tiles having relative movement in the analyzed mission phases, a revolute joint was created with them as reference points. After creating the joint, the orientation of its two reference markers was set so that the z-axis coincides with the axis of relative motion (rotation). To ensure a consistent definition of the parametric model, all joints were set with the same conventions in order to have the signs of the relationships, resulting from the analysis of the forces referring to them, defined in a unique way. The main settings used in the creation of the revolute joints are shown below.

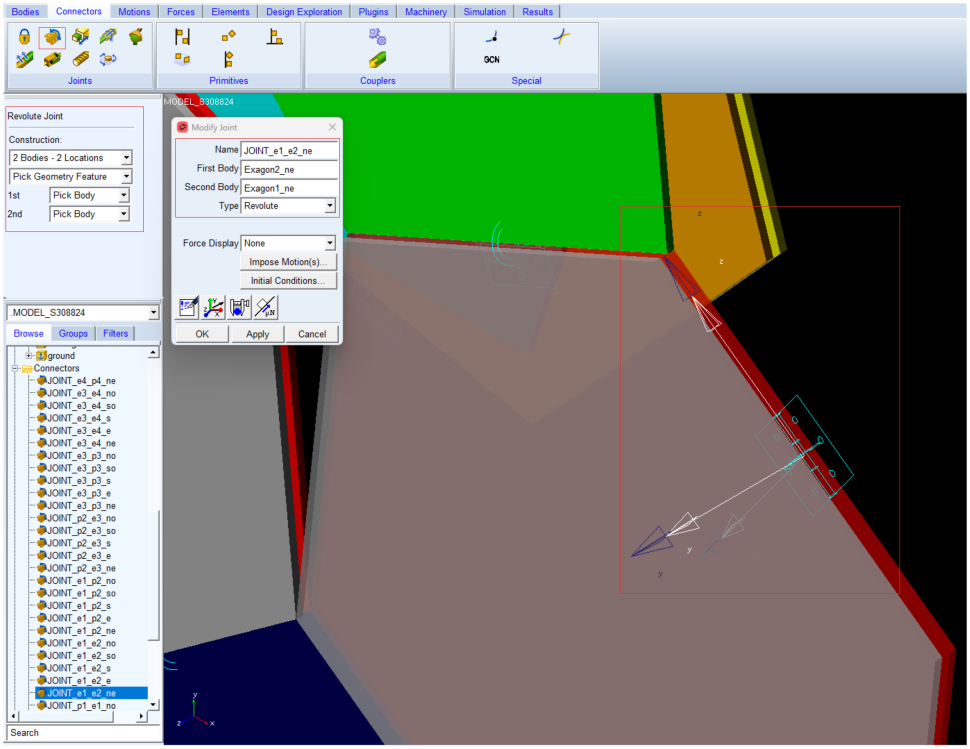


Figure 3.9.: Joint definition in Adams

4. Numerical Implementation and Analysis

4.1. The Adams Simulation Platform

As already mentioned in 2.2, the dynamic validation of the deployable mechanism under examination was performed using Adams simulation environment. This software represents the industry standard for MBD analysis that allows an accurate investigation of the kinematic and dynamic behavior of complex mechanical systems subjected to external loads, constraints, and contact conditions [20]. By using the MBD approach, the bodies are generally modeled as rigid, or as flexible, when modal analysis is employed, and are connected by joints that define their degrees of freedom. This method was especially appropriate for the analysis of the origami based mechanism, as the study mainly focused on its kinematic and dynamic behavior.

4.1.1. Solver Settings and Environment

Solver settings

For the dynamic analysis of the multibody system, the `GSTIFF` (Gear Stiff) integrator was selected [21].

Table 4.1.: Summary of Adams Solver Settings.

Parameter	Value
Integrator Type	GSTIFF
Formulation	I3
Corrector	Original
(H_{max})	(none)

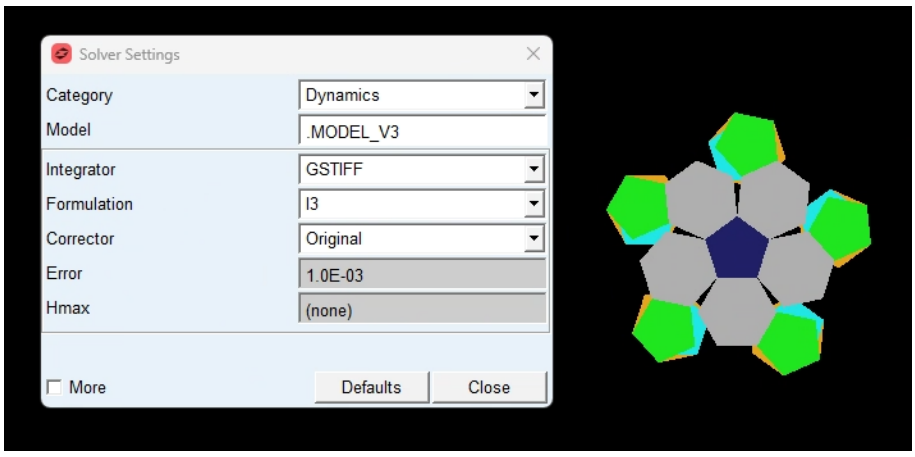


Figure 4.1.: GSTIFF settings

Simulation Environment

Considering the space application of the debris retrieval system, the gravitational environment was modified with respect to the default terrestrial settings. The gravitational acceleration g was set to zero along all axes (x, y, z), in order to simulate a microgravity condition (or orbital free-fall). This assumption allows the inertial forces and the actuation torques to be isolated as the only contributors to the system motion, excluding the self-weight of the structure, which would otherwise be dominant in ground-based tests. Regarding the unit system, the International Units (SI, or MKS) has been used.

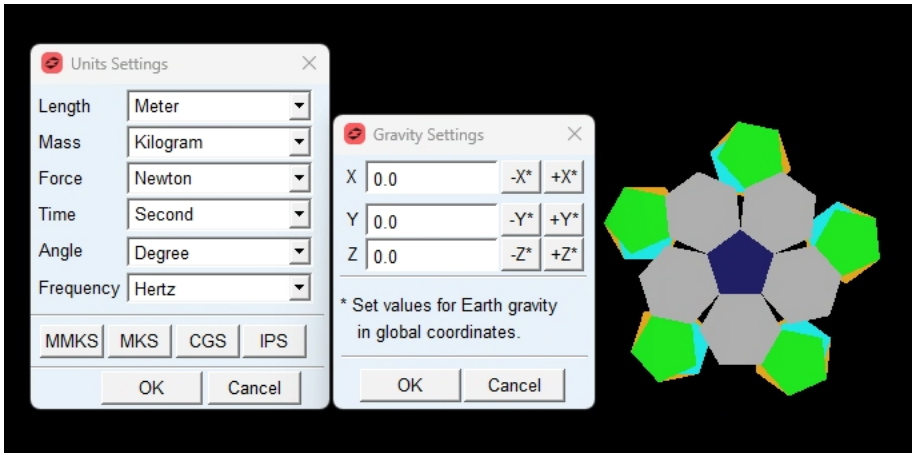


Figure 4.2.: Units and gravity in Adams environment

4.2. Attitude Control Definition

Once the parametric model was defined from a geometric point of view, the control logic of the attitude control system and the rotational forces acting on the hinges needed to be implemented. In order to understand the concepts in this chapter, it is necessary to analyze how Proportional–Integral–Derivative (PID) controllers, and in particular PD controllers, work. PID controllers are among the most common control algorithms used both in robotics and aerospace industries and, as suggested by the acronym itself, they rely on three terms:

- proportional term
- integral term
- derivative term

which are varied to obtain an optimal response. [22] Their fundamental operating principle is based on the calculated error of signal $error(t)$ defined as a difference between desired setpoint (e.g target position) and the measured state variable (e.g chaser position).

$$u(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt} \tag{4.1}$$

Where the parameters are defined as:

- $u(t)$: Control signal (controller output).
- $e(t)$: Control error
- K_p: Proportional gain.
- K_i: Integral gain.
- K_d: Derivative gain.

Proportional Response: The Present

The Proportional term represents the reaction to the current state of the system. It provides the immediate control authority required to reduce the error depending only on the difference between the setpoint and the process variable, referred to as the error term. This term defines the ratio of output response to the error signal. Increasing proportional gain results in increasing speed of the control system response. However, if the proportional gain is too large, the process variable will begin to oscillate.

Integral Response: The Past

The integral term represents the "memory" of the system, accumulating the history of the error over time. This means that even a small error causes the integral component to increase and will cause it to increase perpetually until it is canceled out, so the final effect can be summarized as driving steady state error to zero.

Derivative Response: The Future

This term represents the prediction of the system's trend. It is proportional to the rate of change of the state variable, that means that it manages the output to decrease if the process variable is increasing rapidly.

In this study the Integral action was omitted to prevent integrator wind-up, which would cause dangerous overshoot following the prolonged approach phase. Furthermore, due to the absence of static dissipative forces in the orbital environment, the PD scheme is sufficient to ensure asymptotic convergence without the risk of instability.

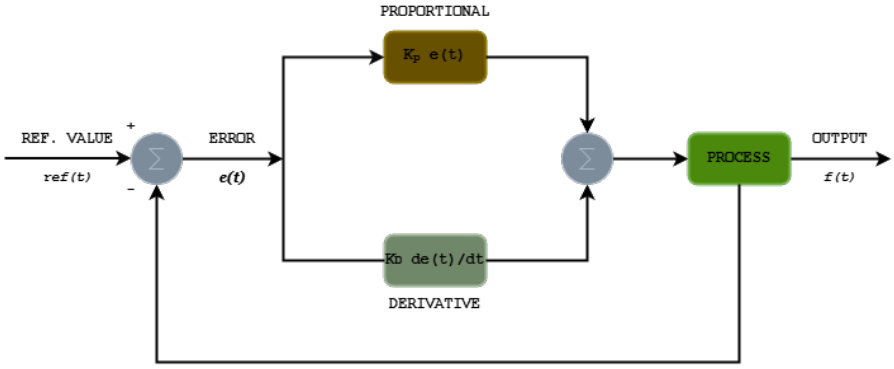


Figure 4.3.: PD controller block diagram implemented for the chaser's attitude control

4.2.1. Control Logic Implementation

This subsection describes the fundamental phases of rendezvous and the various approaches used from a theoretical point of view. To properly tune and implement the previously described PD control logic, a specific rendezvous mission profile must be considered. For this purpose, some considerations regarding the phases and the developed concepts refer to the analysis of the ClearSpace-1 (CS-1) mission, which is a demonstration mission of the rendezvous, capture and de-orbiting phases of the Vega Secondary Payload Adapter (VESPA), launched on board Advanced Generation European Vehicle (VEGA) in 2013 [23]. For the model under consideration, not all the operational phases of a typical space mission are considered, but rather the subset of phases starting from CHASER, which has already reached the orbit of TARGET, is already in phase with it, and has already performed the fly-around trajectories in the far-rendezvous phase. Therefore, the phases under consideration are:

- Close-rendezvous phase
- Capture phase

As highlighted in [23] there are two zones that are fundamental in proximity operations mission timeline:

- Approach Zone (AZ)
- Keep out zone (KOZ)

AZ implies that Chaser had on-board access to a continuous reliable estimate of the relative position of the TARGET, while KOZ implies that the CHASER had to follow the approach corridor and had to ensure a 6-Degree of freedom (DOF) relative closed loop control. Another important achievement from CS-1 is the implementation of decision points in order to progress the mission timeline only if a set of mission-dependent decision criteria is successfully met.
Following the points described above, the Guidance Navigation and Control (GNC) system and a new criterion for the start of the deployment phase were implemented for the origami mechanism under consideration. First, the navigation angles of the CHASER relative to the TARGET were computed to implement the PD logic for attitude control. The main objective, as described in 4.2 is

to eliminate the misalignment error between the reference frame fixed to the CHASER and the one associated with the TARGET. In accordance with [23] two different approach strategies were chosen depending on the distance from the TARGET.

- Far Range $\Longrightarrow$ Line of sight (LOS) pointing strategy
- Close Range $\Longrightarrow$ attitude hold strategy

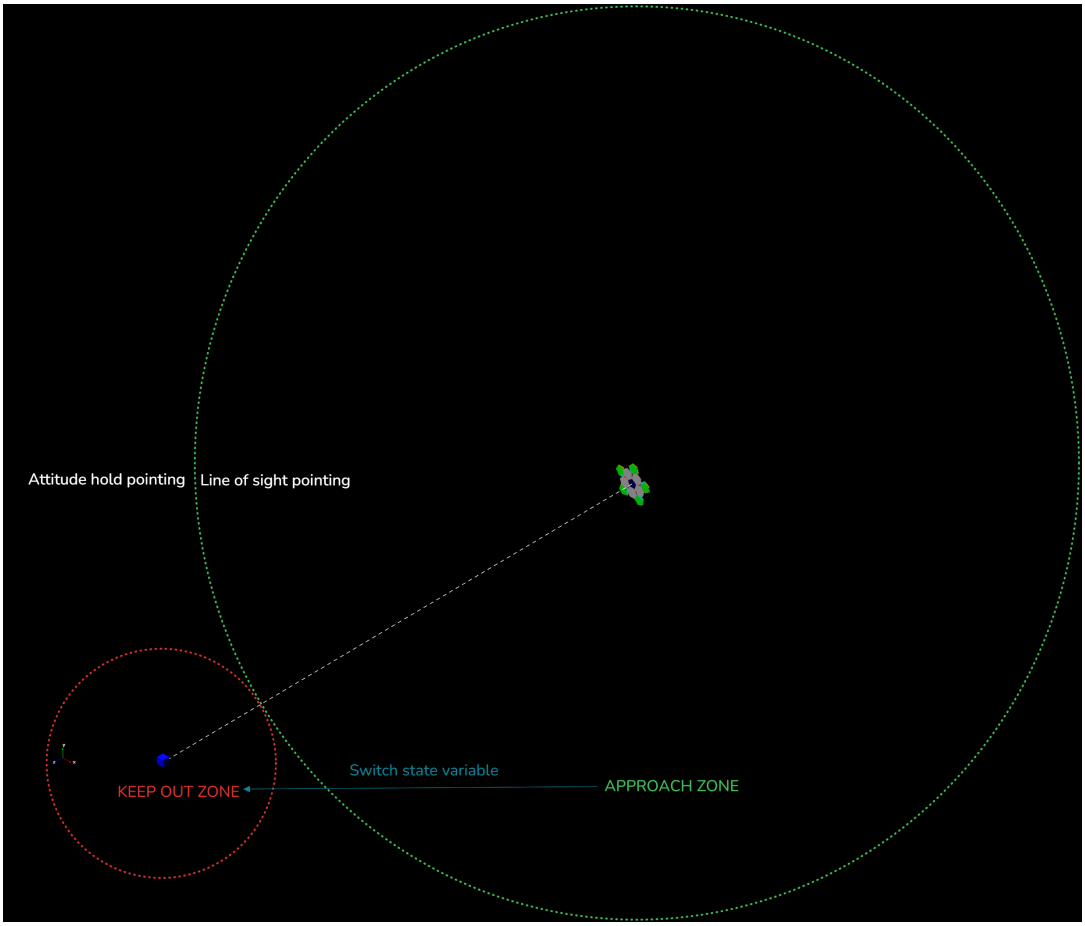


Figure 4.4.: Rendezvous zones

To define these control laws, it was necessary to determine the pitch angle and yaw angle. Since the target may rotate around its own center of mass, a method based on the relative distances between two bodies and subsequent calculation of their arctangents was chosen. To accurately derive all required quantities, a reference frame centered at the center of mass (CM) of the CHASER was defined. This reference frame is characterized by:

- Z-axis: pointing axis
- Y-axis: vertical transverse axis
- X-axis: axis completes the right-handed coordinate system

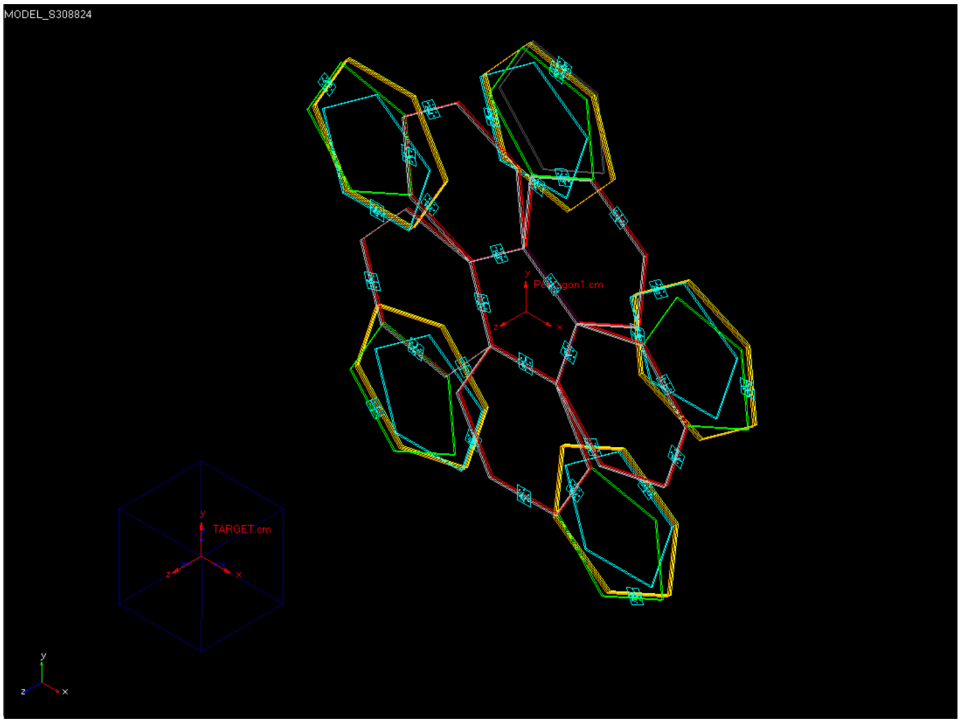


Figure 4.5.: Axis orientation

In the Adams environment, the same was done for the TARGET, and to ensure a consistent approach the same direction and orientation were chosen for the reference frame. It should be noted that, in real world operations and not only in the simulation environment a non-cooperative TARGET, as is the case for most space debris, cannot be equipped with sensors or identification markers. Therefore, the detection of the TARGET's center of mass and its subsequent characteristics must be performed by CHASER's onboard sensors.
Once these reference frames have been defined for each instant, it is possible to obtain the distances as well as the relative velocities along each of them.

$$\mathbf{r}_{rel} = \begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix}, \qquad \mathbf{v}_{rel} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \tag{4.2}$$

4.2.2. Definition of pointing errors

In order to define the pitch and yaw angles, positional data must be converted into angular information. Therefore, within the simulation environment, the ATAN2 function is employed, since it provides a correct angular definition while taking the quadrant into account. First, positional data is extracted using the `Distance along ()` function of Adams and defined as state variables. This led to the definition of the following quantities:

- DIST_X_TGT
- DIST_Y_TGT
- DIST_Z_TGT

The marker configuration used for all reference bodies was defined as:

$$\texttt{DX(From\_Marker, To\_Marker, Along\_Marker)}$$

$$\Downarrow$$

$$d_x = \texttt{DX(TARGET.cm, Pentagon1.cm, Pentagon1.cm)}$$

With this notation, `DX` computes the distance along the X-axis. Following the exact same logic, `DY` and `DZ` are used to extract the distances along the Y and Z axes (d_y and d_z), while the `DM(From_Marker, To_Marker)` function calculates the absolute distance magnitude.
These measurements simulate the data that would be collected by the `CHASER`'s onboard visual sensors (detailed in Chapter 5). Based on these values, the angular tracking errors required for attitude control were computed.

Yaw

The yaw tracking error, ψ_{err}, quantifies the lateral misalignment and drives the corresponding control actuator. Geometrically, it represents the angle between the nominal pointing axis (Z) and the relative position vector projected onto the horizontal XZ-plane:

$$\psi_{err} = \text{atan2}\left(d_x + \epsilon, d_z\right)$$

Pitch

The pitch tracking error, θ_{err}, defines the elevation angle required to compensate for vertical misalignment. It is computed as the angle between the horizontal XZ-plane and the true 3D relative position vector:

$$\theta_{err} = \text{atan2}\left(d_y + \epsilon, \sqrt{d_x^2 + d_z^2}\right)$$

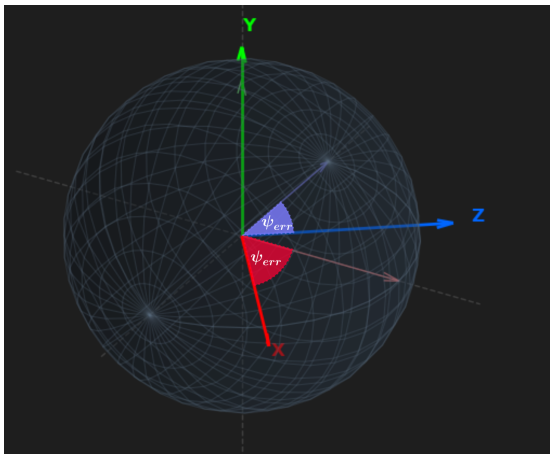


Figure 4.6.: Yaw angle

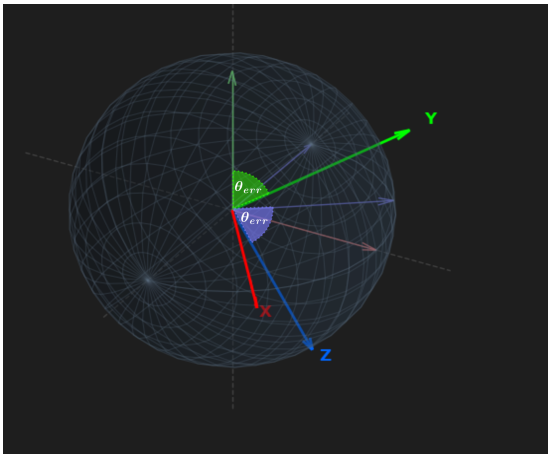


Figure 4.7.: Pitch angle

In the Adams environment, these quantities were defined as state variables according to the following formulation. It is important to explain the choice of

the two-argument arctangent function, atan2, rather than the standard single-argument atan. This choice is critical for the control logic for two reasons:

- extends the output range to cover all four quadrants $(-\pi, \pi]$, resolving directional ambiguities inherent to the standard arctangent which is limited to $(-\pi/2, \pi/2)$
- it guarantees numerical stability within the simulation by implicitly handling singularities where the denominator approaches zero, thereby preventing division by zero errors.

A numerical offset ε is introduced in order to prevent the relative distance from becoming exactly zero. This ensures a stable evaluation of the PD control law avoiding potential numerical discontinuities or warnings in the simulation.

```
! Error_zx_ay_YAW  -- Angular state (yaw)
ATAN2(
    -(VARVAL(.MODEL_V3.DIST_X_TGT) + Epsilon),
     VARVAL(.MODEL_V3.DIST_Z_TGT)
)
```

Listing 4.1: Yaw error state variable definition

```
! Error_zy_ax_PITCH  -- Angular state (pitch)
ATAN2(VARVAL(.MODEL_V3.DIST_Y_TGT)+Epsilon,
      SQRT((VARVAL(.MODEL_V3.DIST_X_TGT)**2)+(VARVAL(.MODEL_V3.DIST_Z_TGT)**2)))
```

Listing 4.2: Pitch error state variable definition

4.3. Attitude Control Implementation

With these state variables defined, the PD (Proportional-Derivative) control strategy for the attitude determination and control system was implemented. First, a VTORQUE element was selected to manage the rotational dynamics.

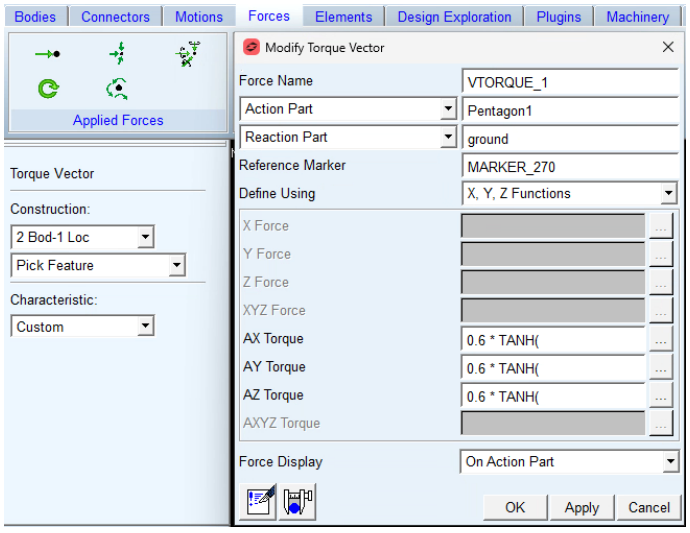


Figure 4.8.: Torque Vector setup in the simulation environment.

It was decided to decouple the control actions: the guidance torques are applied exclusively via this VTORQUE, whereas the navigation forces (translation) are applied as three single component forces ((SFORCES)) acting along the system's characteristic axes (X, Y, Z). The pitch and yaw tracking errors formulas defined previously constitute the proportional component of the PD control, while the derivative component requires angular rate measurements. Consequently, the angular velocities of the chaser relative to the ground reference frame were obtained in order to simulate the behavior of an onboard gyroscope.

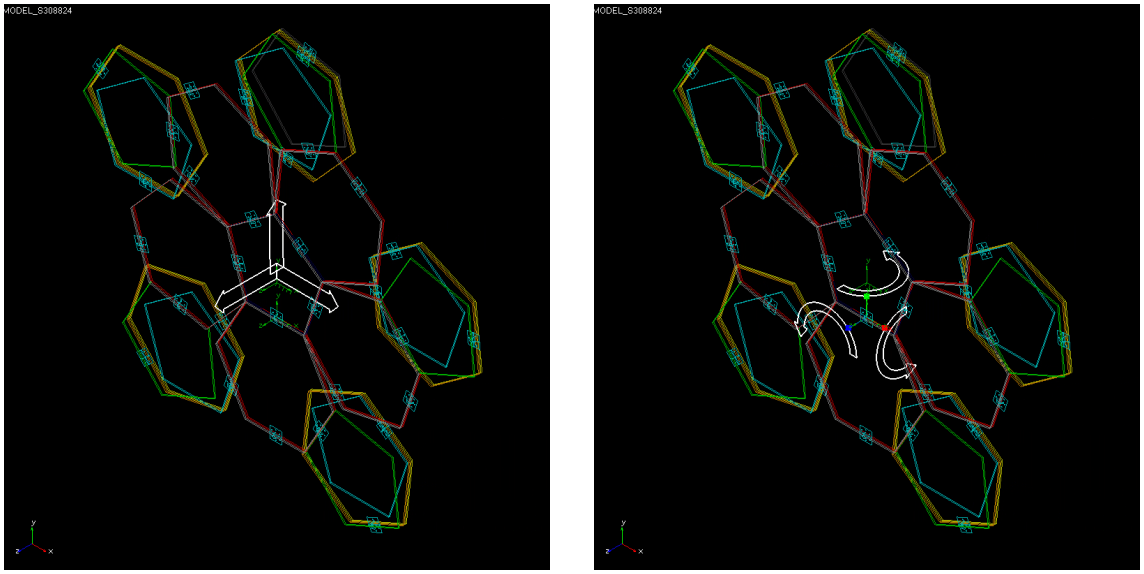


Figure 4.9.: SFORCES and VTORQUES applied to the origami mechanism

4.3.1. Control Law Formulation

The control logic for the yaw and pitch axes is implemented as a non-linear controller with the following main features:

- Saturation using a natural saturation function like TANH
- Mode switching regulated by the SWITCH state variable

The mathematical formulation of the torque command τ_{yaw} implemented in the Adams environment is described by the following equation:

$$\tau_{yaw} = T_{saturation} \cdot \tanh\left(\frac{f_{pd}}{T_{saturation}}\right) \tag{4.3}$$

Where $T_{saturation}$ is the saturation limit of torque along an axis and τ_{yaw} represent the relation expressed from PD controller:

$$f_{pd} = (1 - S) \cdot f_{att_{hold}} + S \cdot f_{LOS} \tag{4.4}$$

S represents the switching state variable ("SWITCH"), which guarantees the transition between LOS approach and attitude hold mode. In the simulation performed, this variable was defined with a function STEP5 having as its parameter

the initial distance between the chaser and the target (DISTANCE_0).

$$\text{STEP5}(distance, distance_0, 0, distance_1, 1)$$

Where:

- $distance$: the current relative distance evaluated by the STEP5 function
- $distance_0$: the threshold (expressed as a percentage of DISTANCE_0) at which the LOS control is completely turned off
- $distance_1$: the threshold (expressed as a percentage of DISTANCE_0) at which the transition from LOS to attitude hold begins

Actuator Saturation

The outermost layer of the control law utilizes a hyperbolic tangent function (tanh) as a modulator. This prevents the controller from requesting infinite torque when errors are large, ensuring that the signal is modulated by the TANH's coefficients and does not exceed the design characteristic of the actuators considered as "raw values" or design variables.[24]

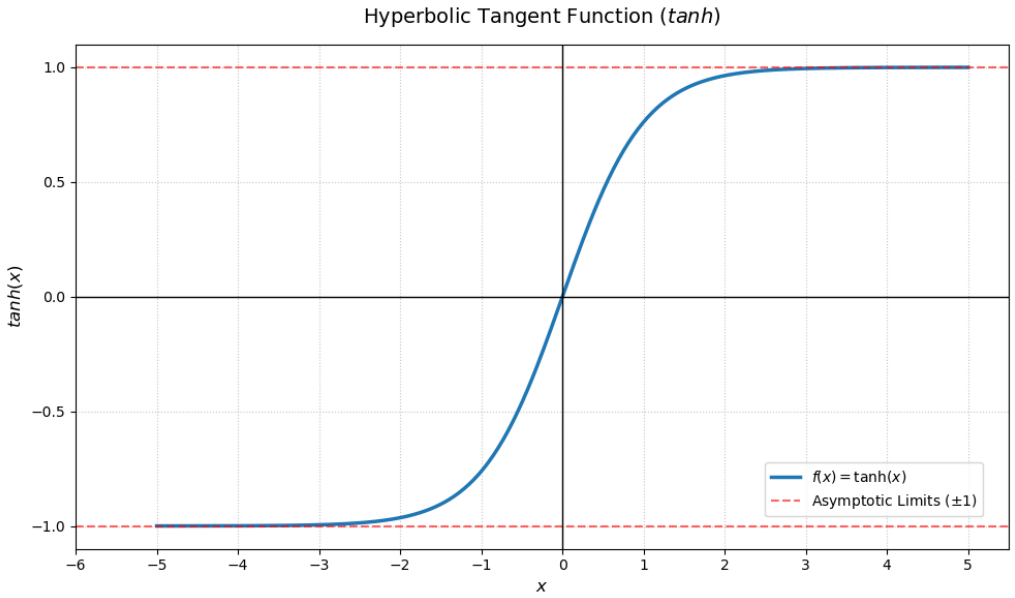


Figure 4.10.: Actuator soft-saturation via tanh function

Phase 1: Active Approach ($S \approx 1$)

During the long-range approach phase ($S \approx 1$), the controller utilizes the relative velocity of the line of sight in order to correct the derivative term [25].

$$f_{LOS} = -K \cdot \theta_{err} - C \cdot (\omega_{gyro} - \omega_{rateo_{LOS}})$$

The term $\omega_{rateo_{LOS}}$ anticipates the required rotational velocity based on the target's linear velocity (feedforward):

$$\omega_{rateo_{LOS}} = \frac{V_{along_{opp_{ax}}}}{\max(d_{safe}, ||\vec{d}||)}$$

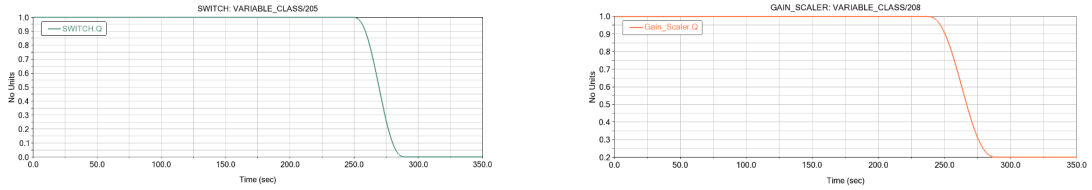
The term $||\vec{d}||$ represents the real distance ("DM"). A safety threshold d_{safe} is introduced in the denominator to prevent numerical singularities like divisions by zero or excessive control efforts when the chaser approaches `TARGET` too fast. In fact, too small values in the denominator would have caused the derivative term to explode, leading to numerical instability or actuator saturation.

Phase 2: Station Keeping and Capture $S \approx 0$

When the system is in the capture phase, the control law focuses on stability to prevent oscillations due to the high relative angular rates at close range.

$$f_{att_{hold}} = -(K \cdot \alpha) \cdot \theta_{err} - (C \cdot \alpha) \cdot \omega_{gyro}$$

A gain scheduling technique is applied via the factor α ("Gain_Scaler"). This factor reduces the effective stiffness (K) and damping (C) as the distance to the target decreases, resulting in a softening of the control response by filtering oscillations caused by geometric singularities when the chaser is close to the target.



(a) (b)

Figure 4.11.: STEP 5 functions: SWITCH and GAINSCALING

a) Strategy transition: the `STEP5` function manages the switch between the two attitude control philosophies.

b) Parameter adjustment: another different `STEP5` function that modulates stiffness and damping coefficients to ensure smoother reactions when close to the `TARGET`.

4.3.2. Z-Axis Approach and Lateral Compensation via PD Control

Once the pure attitude control has been theoretically ensured, it must be integrated with the approach force logic between the `CHASER` and the `TARGET`, in order to guarantee an accurate analysis of the final rendezvous phase under investigation. The main force that guarantees the approach is along the z-axis, which is responsible for managing the approach speed and the maintenance of the docking state once the desired distance from the target (`DISTANCE_START_CAPTURE`) is reached.

In this physical context, the Proportional (P) term acts as a virtual spring pulling the CHASER toward the TARGET generating a force based on the positional error, driving the spacecraft forward when the relative distance is large. Conversely, the Derivative (D) term acts as a virtual damper based on the relative velocity actively resisting the motion as the CHASER accelerates, dissipating kinetic energy to ensure a smooth, asymptotic approach and completely preventing a high-velocity collision at the docking interface.

$$F_z(t) = \underbrace{\sigma(t)}_{\text{Time Ramp}} \cdot \overbrace{(m \cdot G_m)}^{\text{Mass scaling}} \cdot \left[\underbrace{a_{max} \tanh\left(k_{sf}(\Delta z - d_{start})\right)}_{\text{Non-linear proportional action}} + \underbrace{c_d \Delta v_z}_{\text{Damping derivative action}} \right] \tag{4.5}$$

Where the parameters are defined as:

- $\sigma(t)$: A time-dependent step function (representing the *Ramp_Time*) that activates the thrust, preventing instantaneous shocks to the spacecraft structure.
- m - G_m: The spacecraft mass and the corresponding mass gain factor, used to scale the force.
- a_{max}: The maximum allowable approach acceleration, which defines the upper limit of the thruster performance.
- tanh$(\cdot)$: The hyperbolic tangent function acts as a soft-saturation operator. It modulates the positional error relative to the capture distance ($d_{start\_capture}$), ensuring a high response when far from the target and a smooth, asymptotic deceleration as the chaser nears the docking interface.
- $c_d \Delta v_z$: The derivative term, where c_d is the damping gain and Δv_z is the relative velocity.

Simultaneously, by applying the same logic, the controller generates specific forces along the X and Y axes. These lateral thrusts are fundamental to counteract the misalignments of the Z-vector induced by the activation of pitch (θ) and yaw (ψ) moments.
Since the thrusters used for attitude maneuvers can also generate unwanted translations, the PD controller computes the forces F_x and F_y to cancel these disturbances.

4.3.3. System Response - Comments and Graphics

Analysis of the YAW and PITCH error trends confirms the control philosophy adopted. The zoomed portions of the graphs show the initial action of the PD control system, the most crucial phase for the success of the mission. In this phase, starting from an initial error, the PD logic has the aim of canceling the misalignment and subsequently keeping it at or close to zero.

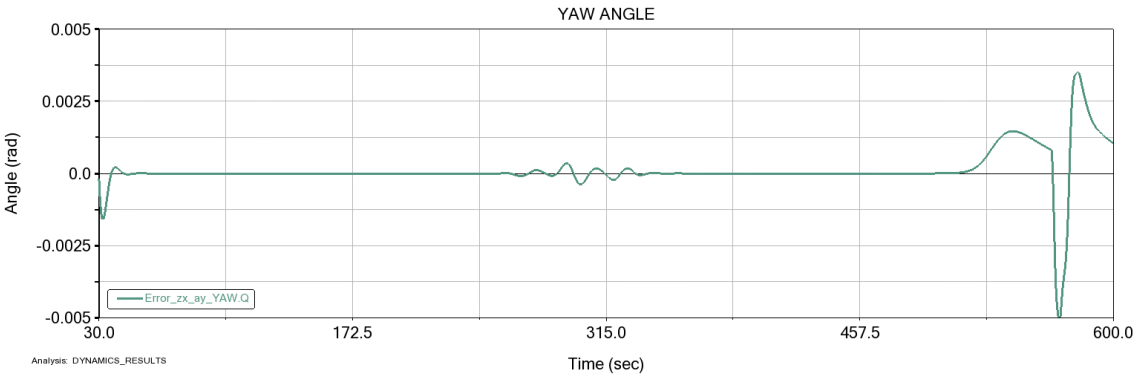


Figure 4.12.: Zoomed-in view of the region of interest - Yaw

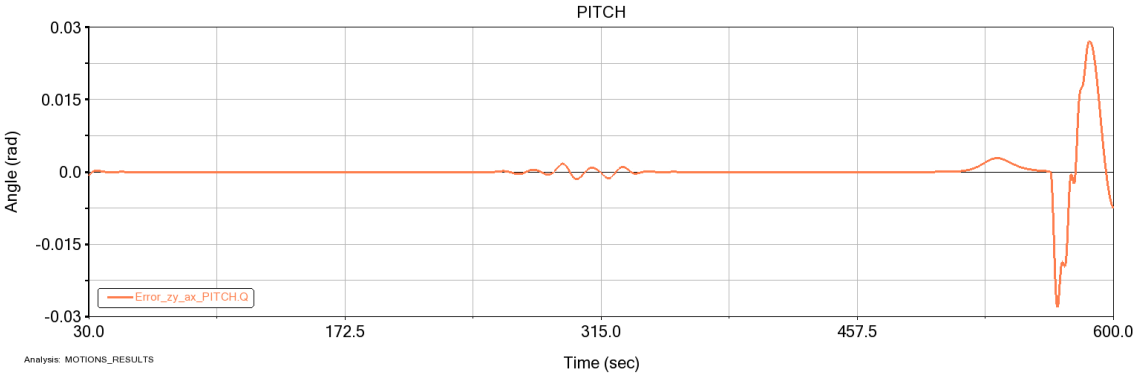


Figure 4.13.: Zoomed-in view of the region of interest - Pitch

The first oscillations occur during deployment phases, during which the forces on the hinges and the variation of geometry induce a dynamic response on the chaser. Even under these conditions the controller maintains the error to values close to zero. The most critical phase, characterized by highly non-linear behavior starts when `SWITCH` function is activated and proceeds in capture phase. In this phase the decision to partially sacrifice pointing along the LOS is intentional, in favor of an attitude hold strategy in order to ensure adequate parallelism between the surfaces of the two bodies. From the trends it can be seen that the controller is strongly influenced by both the capture phase and the progressive reduction of the measurements, which leads to an amplification of the terms based on the arctangent functions.

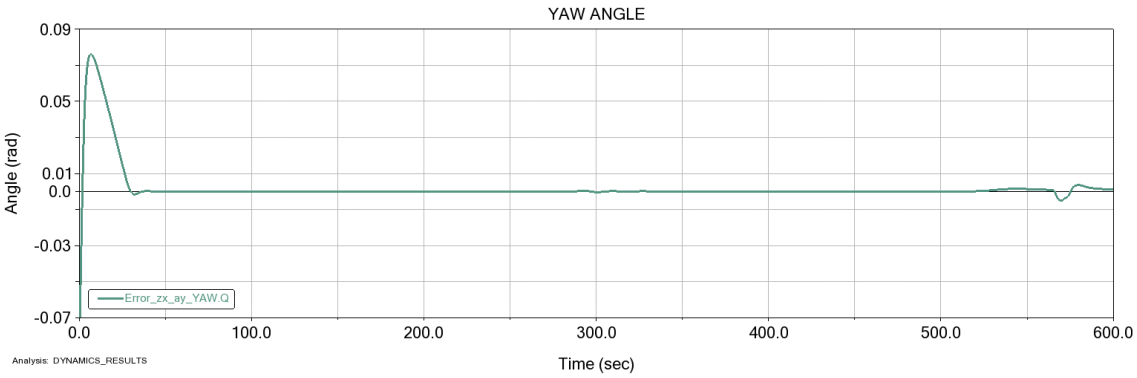


Figure 4.14.: Full-scale response of the system - Yaw

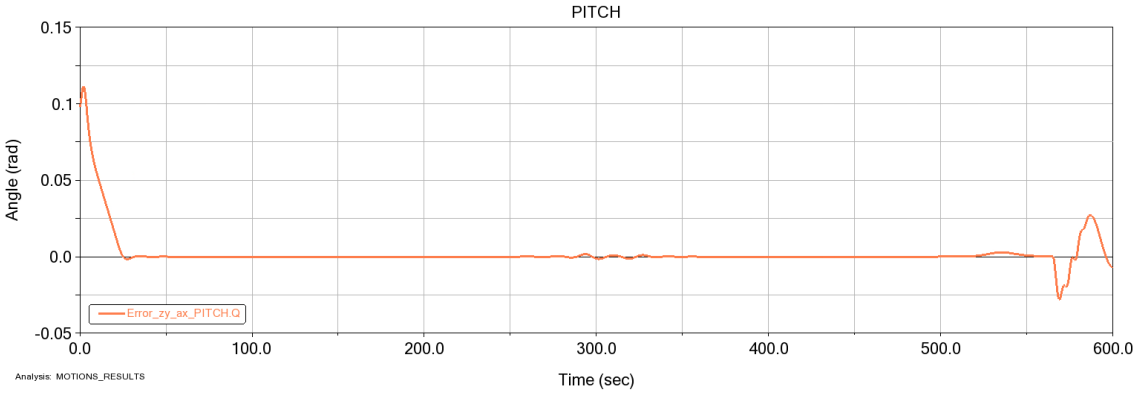


Figure 4.15.: Full-scale response of the system - Pitch

Despite these critical issues, the errors remain within acceptable limits, as confirmed by the graph showing the overlap of the trajectories of the two bodies, which indicates an overall satisfactory system behavior. Nevertheless, these results also suggest that there is still significant room for improvement, particularly in terms of robustness during highly non-linear phases.

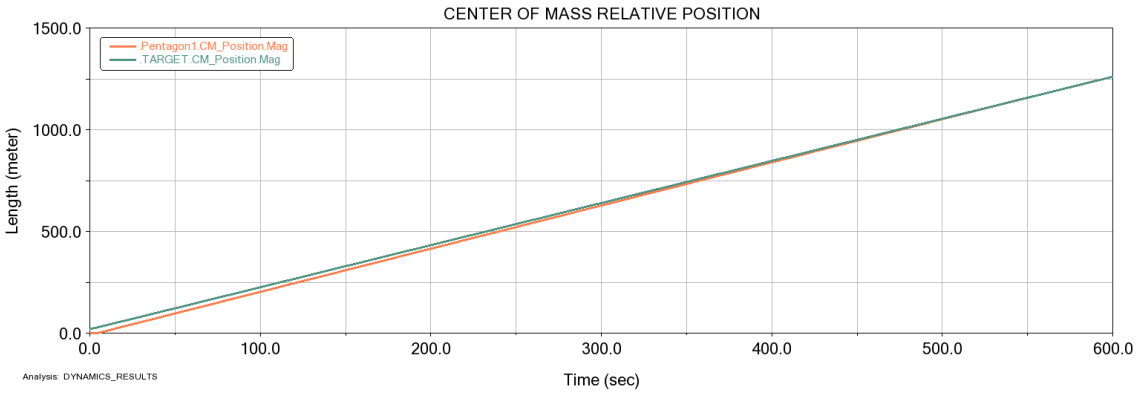


Figure 4.16.: Relative position of chaser and target's centers of mass

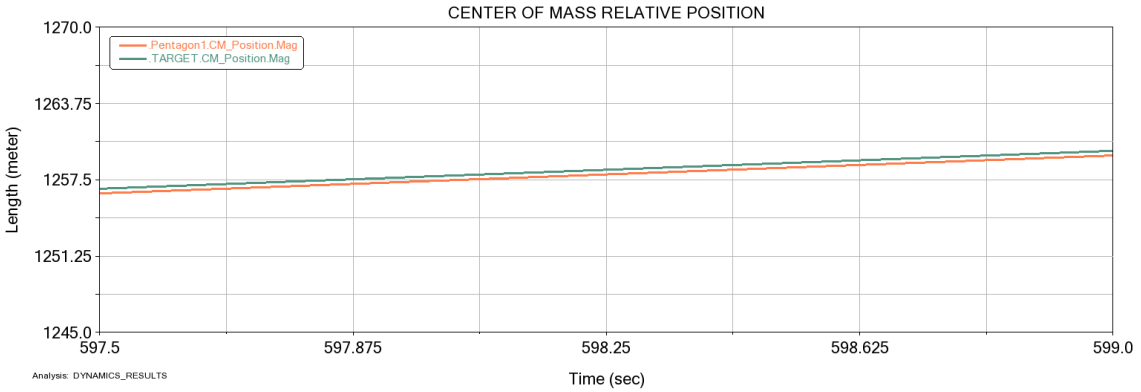


Figure 4.17.: Relative position of chaser and target's centers of mass zoomed view

From the trends related to the relative positions of the CHASER and TARGET's centers of mass it is possible to validate both the attitude control system and the force system used to manage the approach and subsequent station keeping. From the two views, it clearly appears that CHASER progressively attempts to recover speed on TARGET until the relative distance between their centers of mass is almost zero. That "gap-distance" is useful to ensure a safe capture without excessive contact between surfaces. In the simulation framework, the minimum allowable separation distance is parametrized through a dedicated design variable that defines the threshold condition which activates the sensor logic to govern the start of the capture phase.

- DISTANCE_START_CAPTURE → SENSOR_Z definition

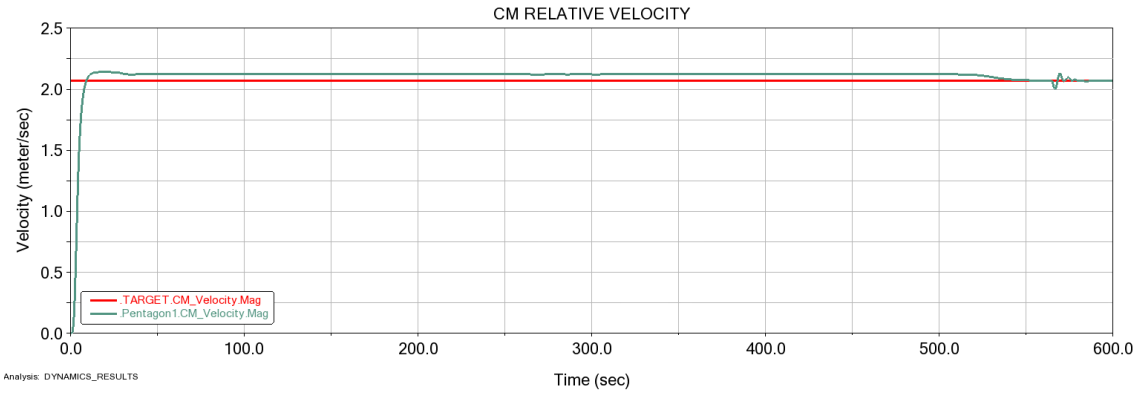


Figure 4.18.: Relative velocity between chaser and target's center of mass

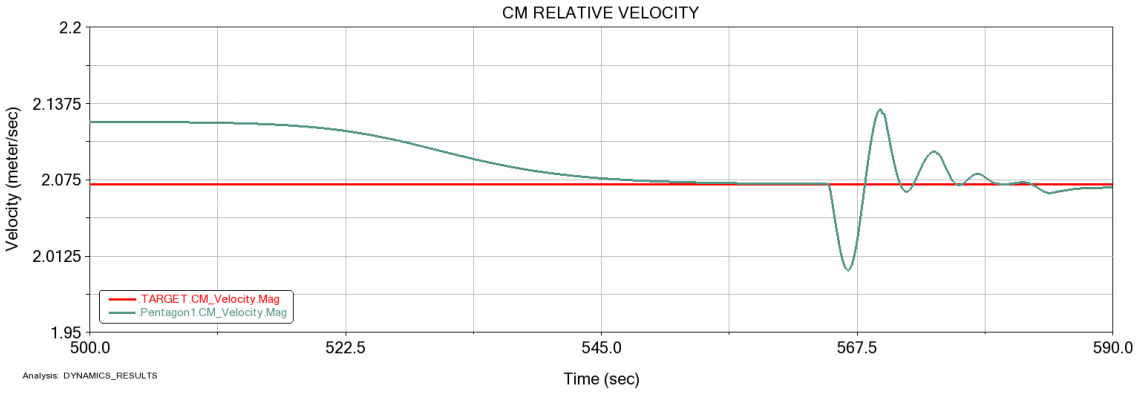


Figure 4.19.: Relative velocity between chaser and target's center of mass zoomed view

The speed trends also clearly identify the logic used for the approach phase and subsequent station keeping. In this simulation, the chaser observes the target moving away with a relative velocity of about $2m/s$, which is typical of the final stages of a rendezvous mission. To reduce this distance, the chaser accelerates until it temporarily exceeds the target's velocity in order to catch up, and then performs a deceleration phase until their velocities become the same. During the capture phase, the largest disturbances are highlighted due to the reaction of the actuators that regulate the tiles' closure. However, these perturbations are quickly compensated by the derivative component of the PD controller, which governs the approach logic. Once the capture phase is complete, the `CHASER` continues at the same speed as the `TARGET`, and for the purposes of analysis, they can be considered as a single body.

4.3.4. New deploying logic

Once the global attitude and approach maneuver of the chaser are stabilized, the focus shifts to the internal kinematics. Specificallt the activation logic for the origami mechanism had to be refined. It was decided to regulate the triggering of the origami opening mechanism no longer as a pure function of distance but as a control function with two distinct parameters:

- Angular errors: pitch and yaw
- Distance at the start of the "final" rendezvous phase `D_0`

Once these parameters have been defined, the overall function that determines its activation phase is computed within the sensor (`SENSOR_avv`), denoted as: $F_{sensor_{avv}}$

$$F_{sensor} = \underbrace{STEP5(E_{rms}, \phi_{min}, 1, \phi_{max}, 0)}_{f_{orientation}} \cdot \underbrace{STEP5(D_0, d_{min}, 1, d_{max}, 0)}_{f_{distance}} \tag{4.6}$$

The first term, $f_{orientation}$, evaluates the combined angular misalignment between the system and the target. This quantity is computed as the mean square error of the Pitch and Yaw angles:

$$E_{rms} = \sqrt{\theta_{pitch}^2 + \theta_{yaw}^2} \tag{4.7}$$

The second term, $f_{distance}$, controls the spatial proximity along the Z axis, measuring the relative distance with respect to the nominal reference value `D_0`. The sensor is configured to trigger when the value of the composite function exceeds the critical threshold of 0.5; exceeding this threshold ensures that the mechanism is simultaneously within the "core" of the permissible tolerances for the two parameters.

4.3.5. Applied Motions and Torque Profiles

For the `MOTION` explained in this section, reference is made to the `pentagon2` and `hexagon3` tiles referring to arm `ne` as they are representative of both phases: deployment and capture. As already highlighted in subsection 2.2.1, the laws of motion are governed by the following relationship:

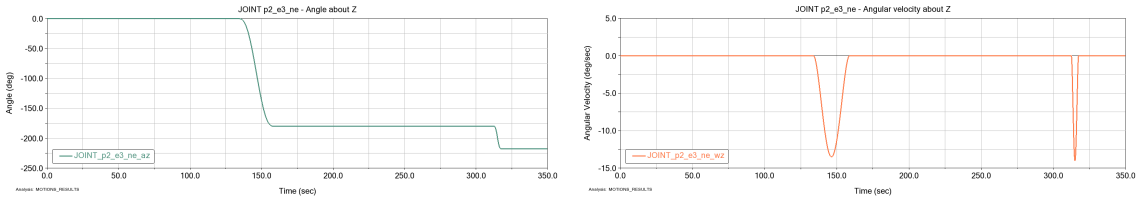
```
IF(SENVAL(SENSOR_capt1):
    0,
    IF(SENVAL(SENSOR_avv):
        0,
        0,
        STEP5(time, SENVAL(SENSOR_avv), 0.0d, SENVAL(SENSOR_avv)+Time_Disp,
        Straight_angle)),
STEP5(time, SENVAL(SENSOR_capt1)+Delay_Capture01, Straight_angle,
SENVAL(SENSOR_capt1)+Time_Capture+Delay_Capture01, Straight_angle+Angle_btw_p_e))
```

Listing 4.3: MOTIONp2e3ne

Once all the `MOTIONS` have been set it was necessary to identify all the measures related to the joint. In particular:

- Projected rotation along Z
- Relative angular velocity along Z
- Torque along Z

Therefore, the trends for these variables were calculated:



(a) Angle about Z

(b) Angular velocity About z

Figure 4.20.: AZ, WZ of p2e3ne Joint

From the trends above we can clearly see the different phases highlighted in the `MOTION` relations.

Phase 1 - Holding
In this phase, the actuator have to maintain the joint (the tiles) in a fixed position and it can be seen:

- Time: $0 \leq t \leq t_{start_{deployment}}$

- Angle: $\theta = constant \rightarrow \theta(t) = \theta_{start}$
- Angular velocity ω: Zero

Phase 2 - Deployment
In this phase the actuators have to drive the joint toward the extension imposed by motion law:

- Time: $t_{start_{deployment}} \leq t \leq t_{end_{deployment}}$
- Angle: $\theta = increasing/decreasing \rightarrow \theta(t) positive/negative slope$
- Angular velocity ω: Positive/negative

For the angle and angular velocity, the sign depends on the reference adopted in measures definition, which, as described above, follow the same convention for every joint.

Phase 3 - Capture

- Time: $t \geq t_{end_{deployment}}$
- Angle: $\theta = increasing/decreasing \rightarrow \theta(t) positive/negative slope$
- Angular velocity ω: Positive/negative

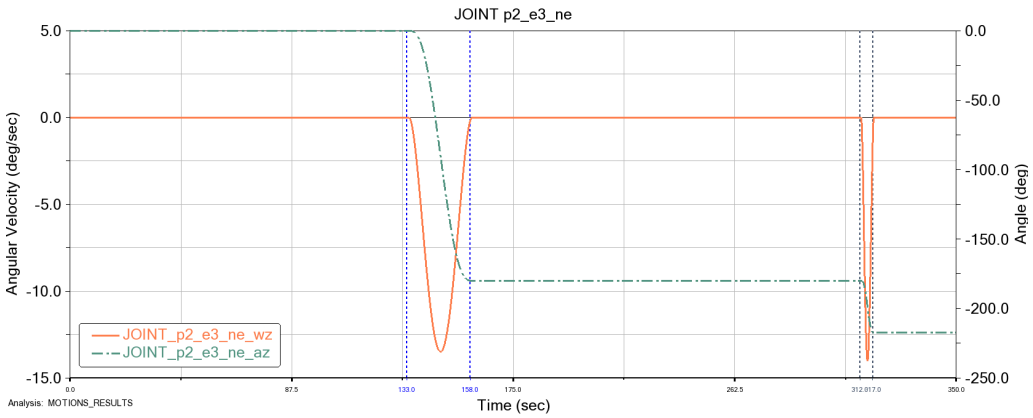


Figure 4.21.: Phases during simulation

To manage these trends, the motion applied to the joint will produce a moment, depending mainly on angular acceleration and moment of inertia.

$$\tau = I \cdot \alpha$$

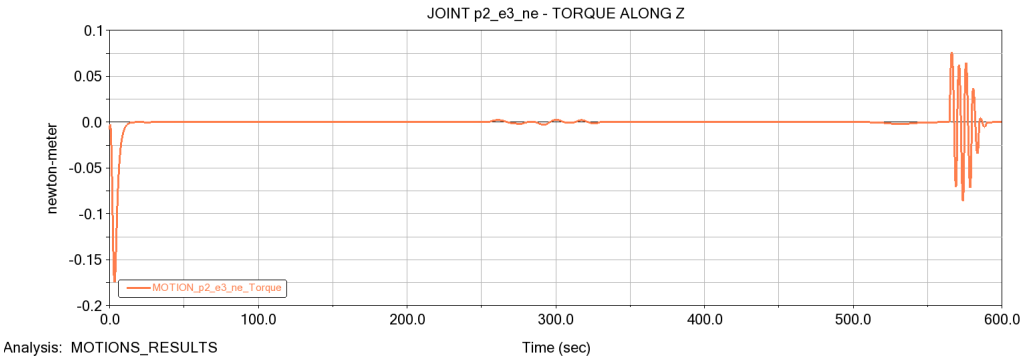


Figure 4.22.: Torque derived from only MOTION analysis

Also in the torque trend, the three distinct phases described above can be clearly identified. Moreover, by considering the complete graph, it is possible to distinguish each phase described in section 2.2.1. These relationships play a fundamental role in the development of the study, since they form the basis for all subsequent considerations related to the dynamic analysis.

4.3.6. Applied Forces and Torque Profiles

This section was identified as one of the most relevant for the parametric analysis. The previous study, as described in section 2.2.1, was based on a construction of SFORCES with a PD controller only in the holding phases and a control governed by STEP5 during the unfolding phases. This approach proved to be effective for a model with fixed dimensions and only after optimization of the STEP5 function characteristics. In fact, in the investigation for STEP5's construction, the torque-related functions produced from kinematic analysis were analyzed, and their maximum and minimum values, together with the relative time intervals were introduced in order to approximate the torque trend. This process, besides requiring a very demanding optimization even for the single model under investigation, would require extremely high computational costs in the case of parametric model. Therefore, while maintaining the same general approach it was necessary to identify modifications in order to carry out the parametric analysis with a general method and relatively low computational costs. Two substantial changes were made:

- Automatic analysis of maximum and minimum values with the corresponding duration of the intervals
- Addition of a PD control of the actuator during the motion phases

In order to implement these features in Adams, external command files were used instead of adding all instructions to the main model file. This approach helped reduce the overall project size and increased the level of automation of the model. Before analyzing the setup of a single SFORCES, here is a brief description of the command file that manages the model when an analysis is requested. Initially, a set of safety checks is executed in order to ensure a correct setup of the simulations. First, all the MOTION elements are activated, and subsequently all the SFORCES are set to zero. A "kinematic" analysis is then performed to compute the quantities described in the previous section. Finally, the

corresponding SPLINE functions are generated based on the obtained results. By doing so, it is possible to obtain the different trends produced by the MOTIONS independently of the model parameterization (e.g model size, characteristic hinge opening times, etc.). Once the SPLINE data have been imported, the trigger times of the sensors modulating the various phases were saved into design variables in order to correctly set the start times for deployment and capture maximum and minimum extraction algorithm which is explained in the following paragraph. To ensure proper validation of the algorithm, the following Adams functions were employed, all documented in [26]

- MAX/MIN: return the maximum or minimum value of an expression or matrix.
- MAXI/MINI: return the index of the maximum or minimum value of an expression or matrix.
- CLIP: extracts a subset of elements from an $M \times N$ matrix and returns them as a reduced $M \times$ N' matrix.

It is important to note that only the search for the first maximum or minimum of deployment and capture is imported using trigger sensors times; all others follow a sequential adaptive approach. Instead of using fixed time intervals, the search window for the minimum value is dynamically updated based on the result of the maximum value search. The process consists of two logical steps:

1. Peak Identification: The algorithm first identifies the time instant t_{peak} where the maximum torque occurs.
2. Adaptive Search: The value t_{max} becomes the starting point for the new search interval. This ensures that the minimum value is extracted specifically from the phase following the peak.

The maximum τ_{max} and its associated time instant t_{max} are identified within a search window bounded by the reference time t_{ref} and the time t_{end_1} defined by the time of a single capture phase:

$$\tau_{max} = \max\{\tau(t) : t \in [t_{ref}, t_{end_1}]\} \tag{4.8}$$

$$t_{max} = t_{ref} + \Delta t \cdot \text{index}\left(\max\{\tau(t)\}_{t\in[t_{ref},t_{end_1}]}\right) \tag{4.9}$$

where Δt is the integration time step and index$(\cdot)$ denotes the index-finding operator (equivalent to the MAXI/MINI function).
To find the nearest following minimum the search domain is dynamically truncated : $\Omega_{post} = [t_{max}, t_{end_2}]$.

$$\tau_{min} = \min\{\tau(t) : t \in \Omega_{post}\} \tag{4.10}$$

$$t_{min} = t_{max} + \Delta t \cdot \text{index}\left(\min\{T(t)\}_{t\in\Omega_{post}}\right) \tag{4.11}$$

This methodology ensures that $t_{min} > t_{max}$, avoiding errors due to disturbances or changes in the hinge moment trend by analyzing models with different parameters. In A.2 an example of this adaptive method can be seen, the maximum of the relative phase of motion (p2e3ne)is based on the minimum relative

to previous phase `e1p2ne`, and the script searches for the maximum only in a time interval equal to the design variable: `Time_Capture`, in order to avoid errors related to unwanted peaks caused by the forces of the attitude and approach control mechanism. The following plot illustrates all the phases of the capture sequence and the associated sensors, highlighting in particular the points referenced in the previous example.

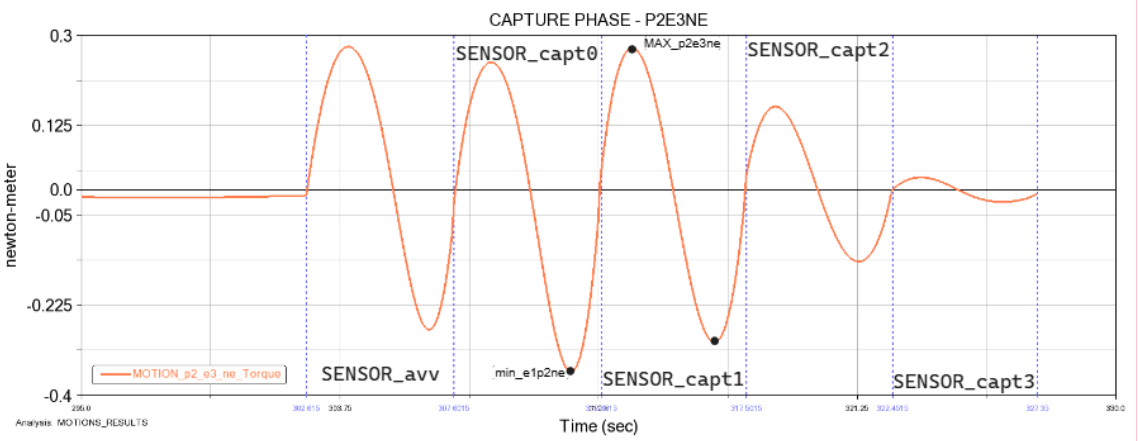


Figure 4.23.: Capture phase - Sensor highlighted

Once all maximum and minimum values have been imported into the model, the `SFORCES` are introduced and to accurately describe their behavior, their general formulation can be divided into two distinct phases that utilize two different control logic styles.

- Holding phase: mechanism remains in a stowed, compact configuration facing only external perturbations
- Unfolding phase: mechanism deploys according to the trajectory imposed by motion laws

Phase 1: Holding

During that phase of the simulation, the primary objective is to maintain the origami mechanism in its compact configuration. In this state, the joints must resist external perturbations, such as initial numerical noise caused by GNC system. To achieve this, a pure PD control law is implemented. In this context, the controller functions as a "virtual spring-damper" system, locking the hinge to a fixed reference angle θ_{hold}. The control torque τ_{hold} is defined as:

$$\tau_{holding}(t) = -\left[K \cdot (\theta_{instant} - \theta_{desired}) + C \cdot \omega_{instant}\right] \tag{4.12}$$

Where:

- $\theta_{instant}$ is the instantaneous joint angle measured by the solver.
- $\theta_{desired}$ is the constant target angle for the closed state.
- K and C are the proportional and derivative gains set as design variables.

In the derivative term, the subtractive term is not present because the desired angular velocity for the phase is zero, since the plate must remain stationary. In the Adams Function Builder, this is implemented by referencing the joint angle `AZ` and angular velocity `WZ` directly, ensuring a stiff response to any deviation from the stowed position before the deployment sequence begins.

Phase 2: Unfolding

Upon reaching the activation time by triggering `SENSOR_avv` (for deploying phase 1), the control strategy switches to the deployment phase. Unlike the standard controller, this phase employs a hybrid feedforward plus feedback architecture. This approach is selected to enhance tracking accuracy by decoupling the bulk torque requirement from the error correction mechanism. The total torque applied to the joint, τ_{total}, is the additive superposition of a pre-calculated moment profile and a corrective feedback term:

$$\tau_{total}(t) = \tau_{st5}(t) + \tau_{pd}(t) \tag{4.13}$$

- Feedforward Term (τ_{st5}): STEP5 Function This component, τ_{st5}, provides the estimated nominal torque required to move the tile against gravity and internal friction. This profile is modeled using the `STEP5` function. The `STEP5` function generates a specific force curve that approximates the theoretically required moment obtained from kinematic analysis

$$\tau_{st5}(t) = \texttt{STEP5}(time, t_{start}, T_{initial}, t_{end}, T_{final}) \tag{4.14}$$

- Feedback Term (τ_{pd}): PD Control The previous term is an approximation, a feedback loop is added to correct any deviations. This PD controller reads the `AZ` and `WZ` trends from imported splines using the `CUBSPL` function [1].

$$\tau_{pd}(t) = -\left[K \cdot (\theta_{instant} - \text{CUBSPL}(\theta_{spline})) + C \cdot (\omega_{instant} - \text{CUBSPL}(\omega_{spline}))\right] \tag{4.15}$$

  This additive configuration ensures that the mechanism is driven by a smooth, globally continuous force profile (τ_{st5}), while the stiffness of the PD loop (τ_{pd}) ensures the system strictly adheres to the intended origami kinematics defined by the spline.

Building on these definitions and integrating them into the logic already validated in [15], the relationship that characterizes the single-force component related to the tiles (`Pentagon2` and `Hexagon3`) is established as follows.
All the phases are essentially modulated by three sensors through three nested `IF` functions. In Adams the `IF` function, given a control expression, is handled through three different expressions depending on specific conditions:

- $expression2 \rightarrow expression1 \leq 0$
- $expression3 \rightarrow expression1 = 0$
- $expression4 \rightarrow expression1 \geq 0$

[1] `CUBSPL` returns an interpolated point or derivative from a curve or surface, fitting the discrete data points exactly via a standard cubic spline method [26].

In the case of a sensor, the SENVAL function returns the value of its state, which can only be binary: “0” when the state is inactive (OFF) and “1” when the state is active (ON).
Figure 4.23 shows the time history of a sensor state obtained from the Adams post-processor.

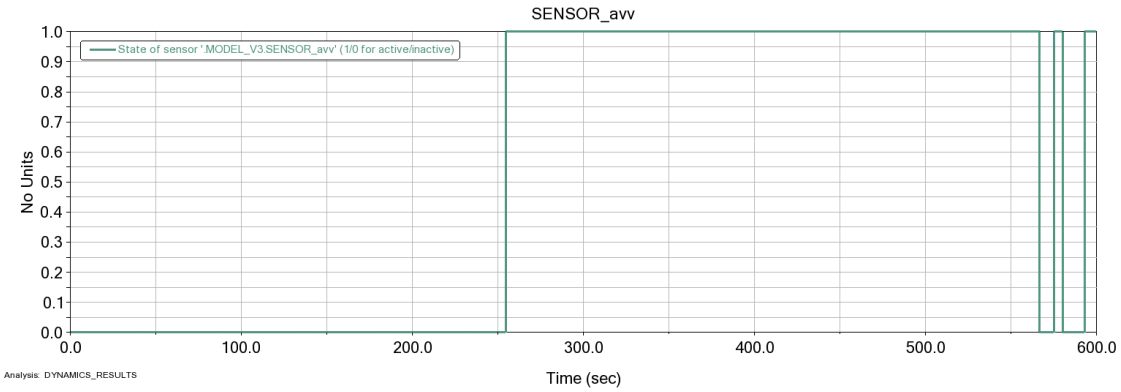


Figure 4.24.: State of SENSORavv

In this case, the solver stores only the first trigger of the sensor to initiate the deployment phase and subsequently deactivates it via a command in the simulation script.

Control logic definition

Once the operating logic of the IF and SENSORS switch state has been defined, it is possible to provide an abbreviated diagram of the control imposed on the hinges; the example in the figure below, although showing generic variables, refers to the joint analyzed previously:

- JOINT_p2_e3_ne

```
SFORCE CONTROL LOGIC
* Note: SENVAL returns 0 (Off) or 1 (Triggered). The <0 branch is required by IF
syntax but is logically impossible (returns 0).

IF( SENVAL(SENSOR_Z) ):
    < 0 → 0
    = 0 → (Off) → IF( SENVAL(SENSOR_avv) ):
        < 0 → 0
        = 0 → (Off) → [Phase 1]
        > 0 → (Triggered) → IF( SENVAL(SENSOR_disp1) ):
            < 0 → 0
            = 0 → (Off) → [Phase 2]
            > 0 → (Triggered) → IF( SENVAL(SENSOR_capt0) ):
                < 0 → 0
                = 0 → (Off) → [Phase 3]
                > 0 → (Triggered) → IF( SENVAL(SENSOR_capt1) ):
                    < 0 → 0
                    = 0 → (Off) → [Phase 4]
                    > 0 → (Triggered) → IF( SENVAL(SENSOR_capt2) ):
                        < 0 → 0
                        = 0 → (Off) → [Phase 5]
                        > 0 → (Triggered) → [Phase 6]
    > 0 → (Triggered) → [Phase 7]

---

- PHASES LEGEND -
[Phase 1]: PD control (holding)
[Phase 2]: STEP5 + PD control (deployment)
[Phase 3]: PD control (holding the deployed phase after first step of capture)
[Phase 4]: PD control (holding the deployed phase after second step of capture)
[Phase 5]: PD control (holding before trigger of the sensor that regulate the capture
phase under exam)
[Phase 6]: STEP5 + PD control (capture phase)
[Phase 7]: PD control (holding after capture phase finished)
```

Figure 4.25.: Hierarchical logic of the SFORCE command.

Figure 4.25 highlight only the scheme of control logics adopted for the definition of the SFORCE. A comprehensive and detailed description of the logic and sensors, are reported into Appendices A.3 and A.5.

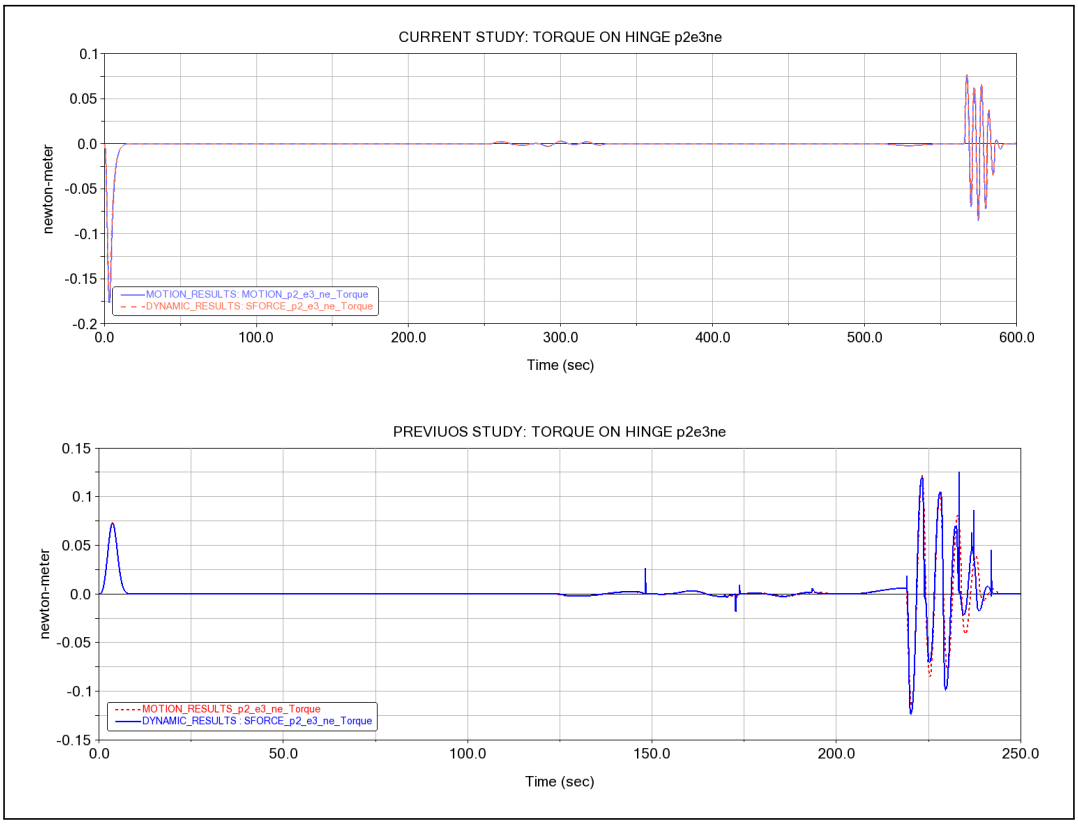


Figure 4.26.: Comparison with previous study - full view

Figures 4.26 and 4.27 highlight the effects of the new formulations included in the parametric model. From an analysis of the overall trend, without focusing on the various operational phases of the rendezvous, a general decrease in spikes is noted. In particular, when analyzing the capture phase alone, which proved to be the most sensitive to the SFORCES framework, the results show a trend much more similar to that of the kinematic analysis, with a substantial reduction in the magnitude of the spikes. A reversal in the sign of the graph can also be seen, this is due to the unique choice of reference system used for every joint, in order to have unique, standardized formulations for each phase The following section will provide a more in-depth analysis of this phenomenon.

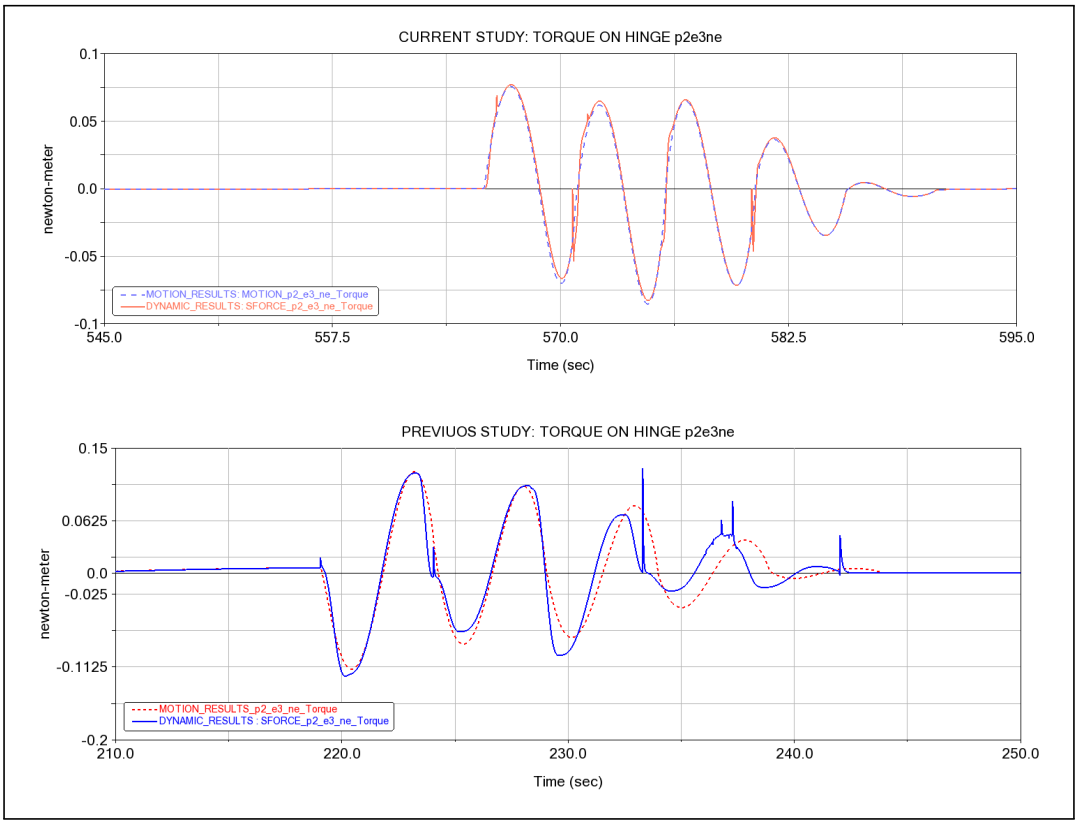


Figure 4.27.: Comparison with previous study - Capture phase.

4.4. Analysis of dynamic simulation Data

This section presents the graphical data obtained by varying the model's `scaling_factor`, highlighting the main similarities and differences and, above all, the marked sensitivity to changes in characteristic dimensions. The aim is to analyze the influence of geometric scaling on the kinematic and, consequently, dynamic behavior of the system, highlighting how changes in dimensional parameters affect the quantities of interest. Through direct comparison of the different cases studied, it will be possible to identify any recurring trends, variations, and possible operational criticalities, thus providing a solid basis for design considerations and subsequent assessments of the reliability and robustness of the model.

- `scaling_factor = 1.2`

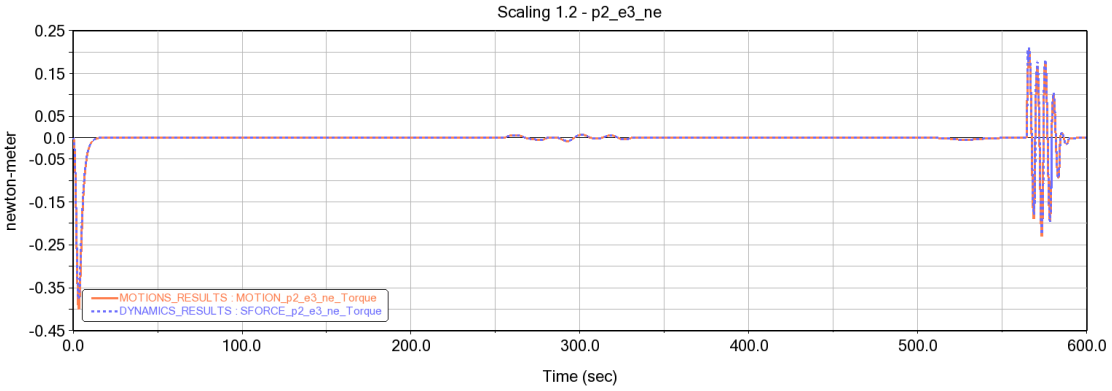


Figure 4.28.: Comparison between two scaling_factor (1.2, 1)

The plot shown in Figure 4.22 represents the trend of the angular moment about the z-axis acting on the joint `p2e3ne`. Since the `MOTION` laws were defined identically for all scaling factors, a similar overall behaviour is expected, at least in terms at least for the shape of the graph and for the mission phases considered. In this case, a comparison between the two analyses conducted is also presented, and even for the different scaling, it can be seen the almost total elimination of peaks or discontinuities between the "kinematic" and dynamic analyses. To enhance readability and allow a clearer interpretation of the results, two zoomed views of the full plot shown in Figure 4.28 are provided below.

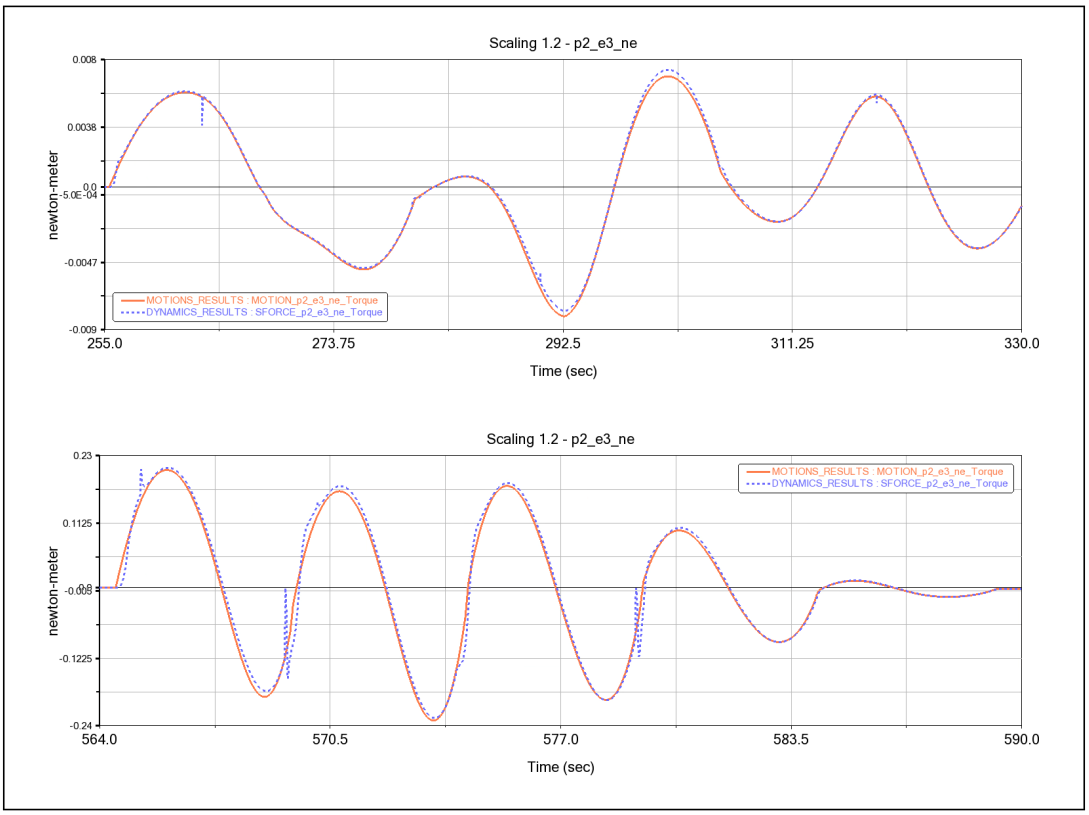


Figure 4.29.: Comparison between the two configurations.

Before moving on to the numerical comparison with the configuration with a

unitary `scaling_factor`, discussed in section 4.3.6, another interesting feature is illustrated, which fully highlights the behavior of the control logic that SFORCES impose on joints. The first graph below shows the evolution of the angle of the tile `hexagon3_ne` over time. The second, on the other hand, presents an extreme zoom of the capture phase in order to highlight the behavior of the PD control. The derivative effect of the control is clearly visible: once the desired angle is reached (considering the error interval set in the sensor), it attempts to maintain the desired position by dampening any oscillations, which, as can be seen, are reduced over time. The vertical lines (green in the graph) represent the status of the sensor that evaluates the actual closure of the tile `hexagon3_ne`; for each vertical line there is a sensor trigger. To avoid discontinuities and various successive activations, the simulation script opted to deactivate the sensor once the first trigger was reached, deciding to ignore the micro oscillations typical of the phase under examination.

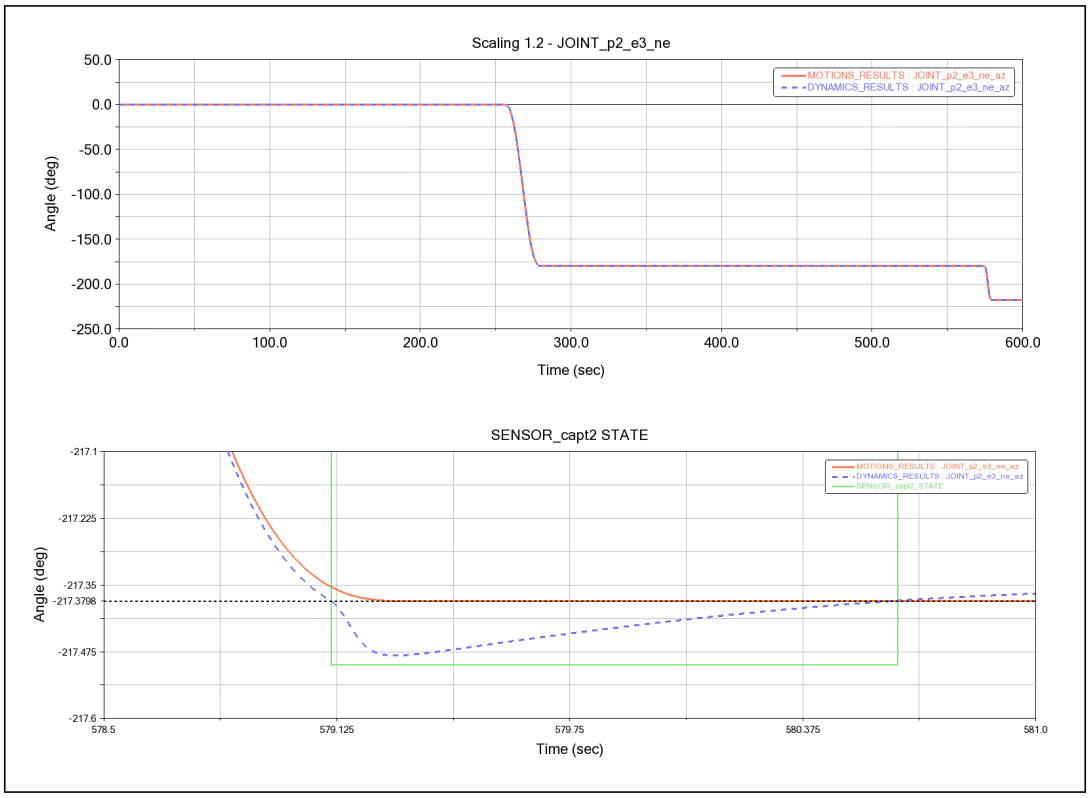


Figure 4.30.: Joint p2_e3_ne angle along Z - Full and detailed capture view

In the following paragraph, firstly two `scaling_factor` will be compared (1, 1.2). Then, the analysis is extended to a higher scaling factor (1.4) to better highlight the main critical issues of the system. In order to quantify the effect of the scaling factor on the system properties, the corresponding geometrical and inertial parameters are reported in Table 4.2.

Table 4.2.: Geometrical and inertial properties for different scaling_factors

Scaling factor	Side L [cm]	Thickness t [cm]	Mass m [kg]	Inertia I_{zz} [kg m^2]
1.00	20.00	0.40	17.00	4.00
1.20	24.00	0.52	33.00	10.90
1.40	28.00	0.62	55.00	25.42

As already mentioned in Chapter 3, the table shows the proportionalities adopted for the model. The thickness was dimensioned according to a conservative criterion, scaling more than linearly with respect to the characteristic dimension of the side. In particular, a 40% increase in length results in a significant increase in thickness, greater than that predicted by simple geometric similarity; similarly, a 20% increase still produces a significant increase when compared to the characteristic measurements of the system. These characteristics modify and highlight the differences that can be seen in the graphs below.

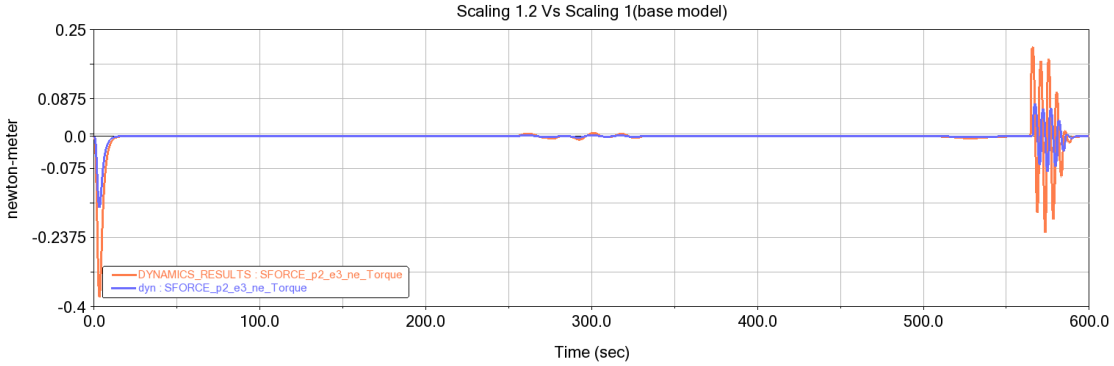


Figure 4.31.: Scaling factor 1.2 vs. scaling factor 1

Figures 4.31 and 4.34 clearly show the effects of a 20% variation in the side length with the consequent effects highlighted on the quantities in table 4.2 on the moment acting on the hinge, which, alongside attitude control logic, is a key factor for the success of the mission. The presented data refer to the dynamic analysis carried out with all SFORCES active on the joints. It is clear that this increase in the scaling factor leads to peaks approximately three times higher during the operational phases managed by the STEP5 and PD control logic.

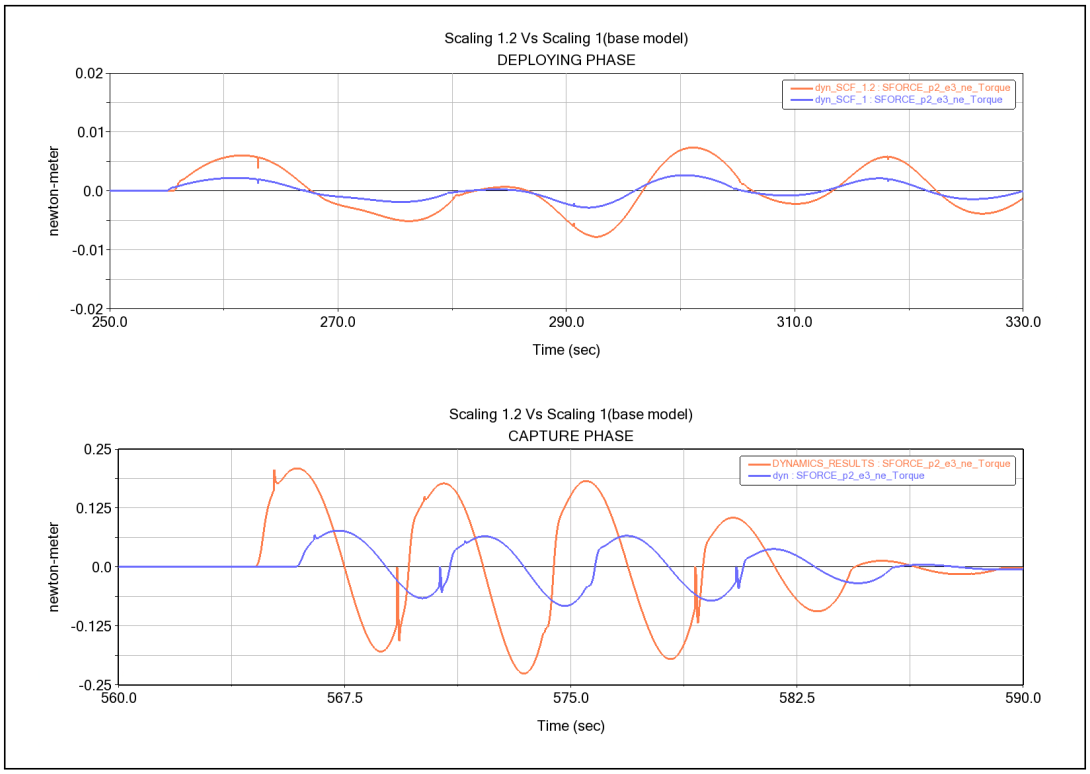


Figure 4.32.: Detailed view of the deployment and capture phases Sf 1.2 vs Sf 1

The parametric model, as described in the previous sections, is capable of autonomously managing changes to the system's characteristic dimensions, with the exception of the proportional and derivative gains of the PD control on the hinges. To determine their adjustment, even when based on proportional relationships derived from mass and inertia, noting that a rotational moment is strictly linked to the latter, manual optimization by the user is required to successfully complete the analysis. Generally, the problem consists in micro-oscillations produced by the hinge controls which induce a temporal shift in sensor triggering compared to the "kinematic" analysis. This phenomenon is significantly aggravated by the hinges' control logic, which relies on a discrete control function such as `IF`. Because this logic forces an instantaneous transition between states, it generates force spikes that the system cannot smoothly absorb. Once this issue is identified, the two values have been manually optimized to compensate for the deviations caused by this phenomenon. In this regard, a larger scale comparison clearly highlights the appearance of force peaks that, if are not properly attenuated, can lead to the interruption of the simulation in Adams. However, a manual calibration process is not ideal for a parametric model, indicating that the current control formulation, based on `IF` control logic, could be upgraded to obtain better results. This suggests that moving beyond simple conditional logic towards more robust control strategies could yield more stable results and eliminate the need for manual tuning, as discussed in the Chapter 6.

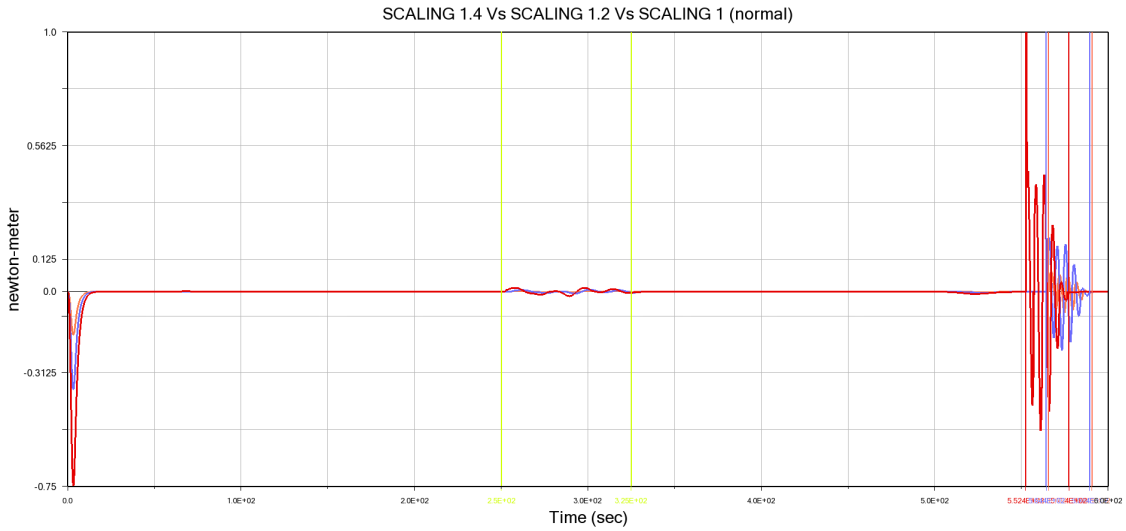


Figure 4.33.: Comparison between three different scaling_factor (1.4, 1.2, 1)

The graph shown above, although not particularly illustrative from a visual perspective, has been included to highlight the opening and capture strategies adopted from a timing perspective and to show that the most problematic phase is capture, particularly the first-level tiles and their control sensors:

- `Pentagon1_hexagon1_()`
- `SENSOR_avv`
- `SENSOR_capt0`

Regarding the different operational phases, it is clear that the strategy adopted was to keep the characteristic times of each phase constant and to adapt the logic of the system to these times. However, since the model is designed to be parametric, future studies could also consider totally opposite approaches, such as adapting the characteristic times to a very narrow range of controls in order to make a trade-off analysis aimed at identifying an overall optimized configuration.

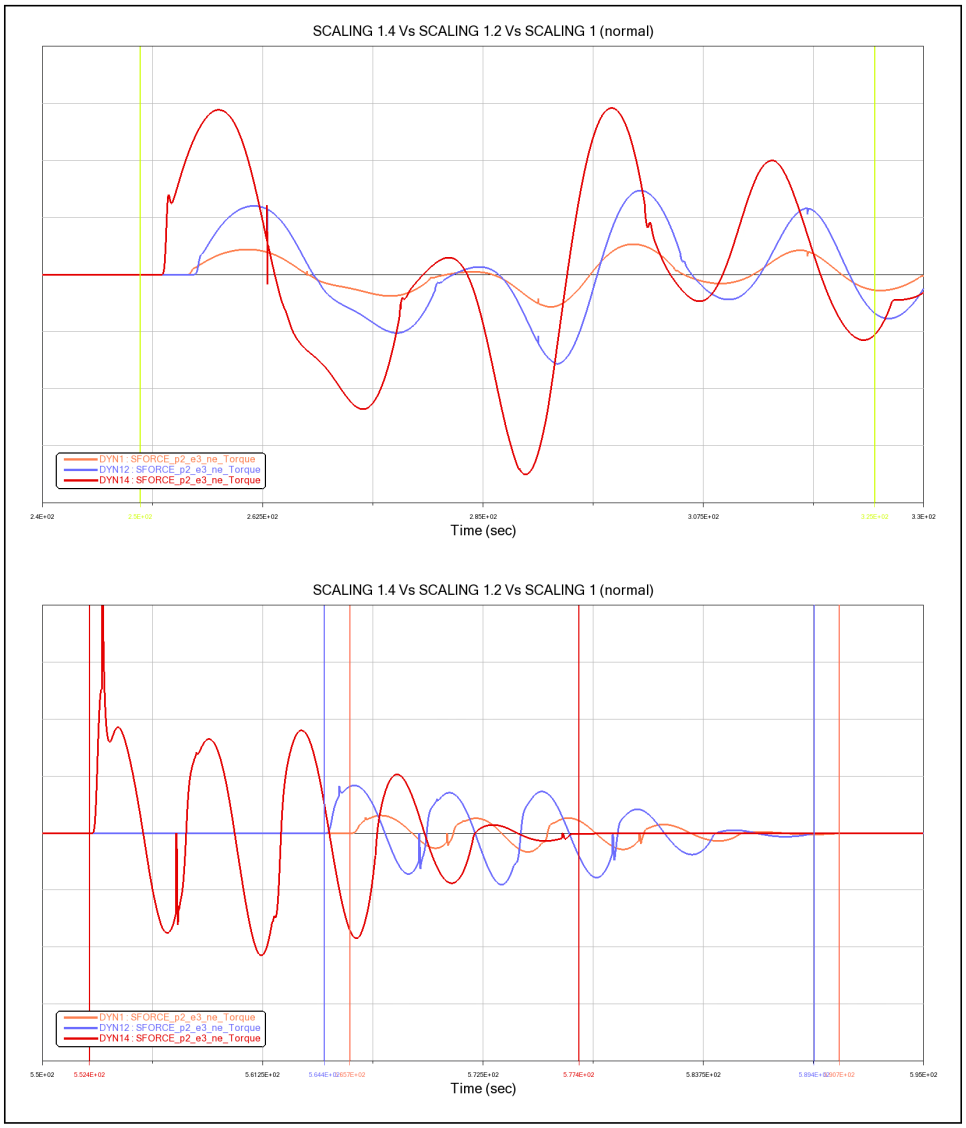


Figure 4.34.: Detailed view of the capture phase Sf 1.4 vs Sf 1.2 vs Sf 1

5. Sensor Architecture with LiDAR-based Relative Navigation

5.1. GNC Sensors

In order to consider a real application, the different mission phases and the sensors that could support them are analyzed. For the proposed origami-based mechanism a fully on-board space debris detection system can be chosen, given the following constraints:

- Non-cooperative TARGET
- Rapid dynamics (tumbling motions)
- High precision capture requirements

Based on the mission characteristics and on the expected behaviour of the target, the rendezvous operation can be divided into the following operational phases:

Table 5.1.: Mission profile and distances for satellite docking

Phase	Range	Objective
Far	> 10 km	Origami release
Mid	10 km – 10 m	Target acquisition & approach
Short	< 10 m	Alignment & docking

The following analysis focuses only on the final meters of the rendezvous, which correspond to the phases simulated in the Adams model. For these phases, the appropriate sensors for guidance and navigation will be identified, paying particular attention to their Technology Readiness Level (TRL). As highlighted in [27], it is possible to identify different optimal sensors for every phase mentioned in the table above.

5.1.1. Mid and short range sensors

At this stage, the attitude control system uses a LOS pointing strategy, and therefore highly reliable and accurate instruments are needed for angular and distance measurements. The primary purpose of this stage is to reconstruct the 3D model of the space debris in order to obtain the following characteristics:

- Relative distance reported in a reference system relative to the chaser
- Estimated rotation speed and tumbling dynamics
- Evaluation of the hypothetical target's center of mass
- Estimation of a possible optimal point for final contact or berthing

One sensor that can be analyzed is widely used in the space sector for these purposes: the Light Detection and Ranging (LiDAR).[27]

Pose estimation challenges and LiDAR

As already mentioned, a reliable estimation of the target's pose is essential for the successful execution of the approach and docking phases in a space debris removal mission. Over the years, many technologies with high TRL in the space sector have been evaluated, such as optical cameras and stereo cameras with all real-time data processing algorithms integrated, but with the latest advances, LiDAR is becoming of great interest. LiDAR is an active remote sensing system that uses light emitted by a rapid-pulse laser to reconstruct the three-dimensional topography of an object. Essentially, the light emitted by the source travels until it hits the object and is then reflected back to the system receiver, where a high-precision chronometer measures the time interval Δt between the emission of the signal and its reception, then calculates the relative distance and maps the points. [28] Given the intrinsic physical and methodological characteristics described above, LiDAR is attracting considerable interest compared to other technologies for the following reasons:[29]

- Insensitivity to changes in lighting: All traditional cameras (except for infrared ones) extremely depend on external light, the LiDAR, on the other hand, being an active sensor (emitting its own light source), is extremely robust to the changes in lighting typical of the space environment.
- Native depth information (compared to monocular systems): Monocular cameras have difficulty estimating distance, as they only refer to two-dimensional frames. LiDAR directly measures the three-dimensional distance of objects with high precision, providing immediate metric data.
- Hardware independence (compared to stereo systems): Although stereo systems are widely used, they can only estimate depth within a range limited by the physical distance (baseline) between the two optics.
- No distortion when there is movement (advantage of Flash LiDAR): While traditional scanning LiDARs acquire one frame (point) at a time, generating possible distortions if the target is rotating rapidly, flash LiDAR simultaneously acquires the entire point cloud, making it particularly suitable for non-cooperative rotating targets.
- Hardware efficiency: Modern flash LiDAR sensors feature a high frame rate, a structure with no moving mechanical parts, and a high level of integration, resulting in advantages in terms of reliability and compactness.

Building upon these advantages, the processing of the data acquired by a Flash LiDAR allows for the instantaneous generation of a high-density, distortion-free "point cloud," which accurately represents the 3D model of the scanned surface.

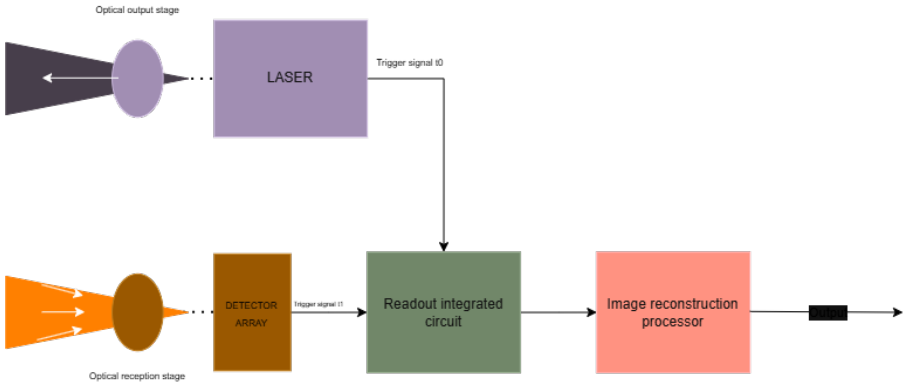


Figure 5.1.: LiDAR point cloud processing

Due to its mechanical configuration, the origami structure imposes strict constraints on the mass and volume of the sensors that can be integrated. Therefore, the use of flash lidar sensors is entirely preferable to complex redundant optical camera systems. The implementation of these sensors directly on board is attracting increasing interest, driving the search for lower-cost alternatives suitable for micro and nano-satellites, in fact a recent study evaluates the application of commercial LiDAR sensors, originally developed for automotive autonomous driving, for use on CubeSats. Their analysis confirms that the physical dimensions and operating ranges of these commercial sensors are fully compatible with the requirements evaluated in the simulations; making their integration into the proposed model a highly feasible solution [30]. Further supporting these hypotheses is the study [31], which identifies flash LiDAR as the optimal technology for 3D mapping of space debris using nanosatellites. Here, the authors demonstrate that the use of a single laser pulse combined with a photodiode array for instantaneous acquisition of the entire scene overcomes the limitations of delicate mechanical scanning systems, offering the stability and efficiency necessary to track targets even within the rigid constraints of a CubeSat.

LiDAR integration in the proposed mechanical system

In the mechanism under analysis, the integration of on-board sensors represents one of the main engineering challenges. Several critical aspects must be considered:

- the minimum thickness of the tiles
- the management of wiring in very limited spaces
- the variation of the field of view (Field of view (FOV)) during the space debris capture phase

To address these issues, different strategies can be adopted:

- embedding the sensors into the surface and protecting them with materials that do not affect their "visual" capabilities
- investigating the use of metamaterials in order to make the material itself act as a sensor, thus completely avoiding the problem of direct impacts on the tiles

From this perspective, the first solution already have existing studies and practical applications, whereas research on metamaterials is still at an early stage and can therefore be considered mainly as a future development.

5.1.2. TRL Assessment and Support cameras

Although Flash LiDAR seems to be a promising primary navigation sensor, its use in a space debris removal mission must still be supported by an adequate TRL. Considering the standardized scale in the figure and referring to all the studies mentioned above, a TRL level of 5/6 can certainly be considered if only the use of Flash LiDAR is taken into account. Therefore, for the purposes of the analysis currently being carried out on the origami mechanism, it would be necessary to define the total architecture of the LiDAR sensors, consider any redundancies and take into account the limitation that the origami mechanism closes into a sphere, thus obscuring all internal sensors once capture has taken place. The situation is different if, while considering lidar as the main sensor, we also consider the addition of traditional cameras that have been widely used and tested in operating environments, even considering the limitations described above.

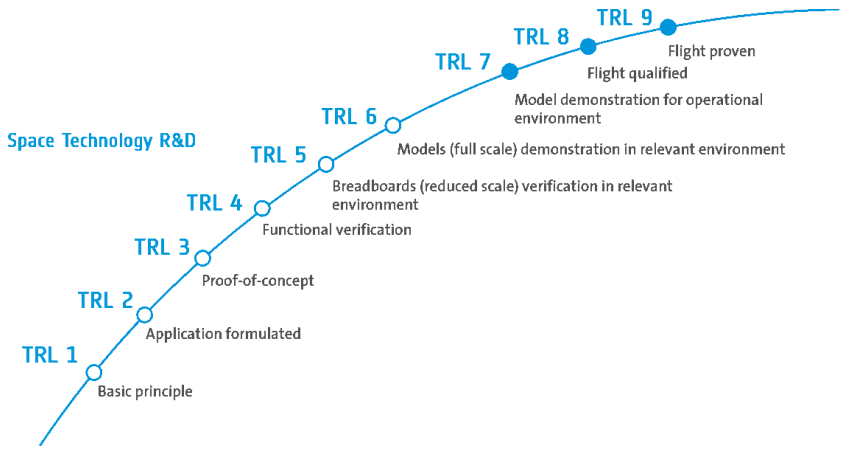


Figure 5.2.: Technology Readiness Levels Scale (TRL)

Credits: © ESA.

The integration of traditional sensors to support the LiDAR system would further increase the technological maturity of the proposed solution. These sensors are based on technologies that have already been extensively validated in the space sector, and combining them with LiDAR would result in a more robust and reliable navigation system, reducing operational uncertainty during the most

delicate phases of the mission. One relevant example is the RemoveDebris mission, a technology demonstration mission for active space debris removal. In addition to testing capture technologies such as a net and a harpoon, the mission also evaluated a Vision based navigation (VBN) system, demonstrating the feasibility of optical sensing techniques for rendezvous and target characterization [32].

Technological Benchmark: The VBN System as Mission Reference

The RemoveDebris mission aimed to demonstrate the effectiveness of a vision-based navigation system using real data acquired in space with the following objectives:

- To demonstrate the effectiveness of advanced navigation and image processing algorithms using real flight data collected from a set of complementary sensors: a standard camera and a flash-imaging LiDAR developed by CSEM.
- To validate the flash LiDAR acquisition system in flight, achieving a Technology Readiness Level (TRL) of at least 7/8.

The payload package has very compact dimensions (15 × 10 × 10 cm), which are compatible with the origami based mechanism considered in this study, and a mass of $1.8kg$, which is also consistent with the characteristic constraints of the model under investigation. In addition, the use of another fundamental mechanism is presented; the integration of which must be considered for the success of a space mission: a dedicated onboard data processing unit responsible for image acquisition, processing, and navigation data extraction from the main sensors of the payload:

- a flash LiDAR
- a standard camera

Another aspect that supports the compatibility of the RemoveDebris' payload with the origami based mission under consideration is the trajectory profile and the corresponding image acquisition phases. As described in the previous section 4.2.1, the objective of each phase is consistent with the assumptions adopted in this work as reported in [23]:

- Far range: the target appears very small (less than 20 pixels) and only the line-of-sight (LOS) can be measured.
- Mid range: the target becomes larger (between 20 and 50 pixels), allowing a more accurate LOS estimation.
- Short range: detailed 3D algorithms are used for pose estimation.
- Blind zone: the target is too close to be detected by the sensor (in this case an additional infrared camera with a high FOV could be considered).

In conclusion, LiDAR represents a solid basis for the navigation system and could be effectively integrated into the mission under consideration. However,

there are some points that have not been analyzed in this discussion but which must be declared and defined before the sensor system for the GNC system is finalized:

- A mission design defined with clear orbital planning
- Strategy for ejecting the chaser from a mother ship
- Chaser architecture: the capture method has been defined, but a bus must be integrated to the origami "net" in order to manage propulsion, navigation after capture, and data analysis.

Some of these aspects, together with other critical issues that emerged during the analysis, will be examined in the chapter 6.

6. Remarks and Future Perspectives

The analyses conducted have made it possible to define and validate, within the simulation environment, the effectiveness of the origami mechanism for capturing space debris during the last critical meters of the approach phase, at least in the preliminary stage. On the other hand, the parameterization of the model with respect to a scale factor, the relative distance between the chaser and the target, combined with the definition of a new attitude control logic and approach forces governed by PD controllers, demonstrated the theoretical feasibility of the terminal rendezvous maneuver. However, the transition from an idealized dynamic model to a real system operating in space requires the consideration of increasingly precise and complex engineering hypotheses and challenges. In fact, the simplification of certain physical dynamics, necessary in a preliminary phase of the study, leaves open several problems related to the interaction between chaser and target during the capture phase and the subsequent return to the mother ship. In this chapter, the main open issues of the model will be analyzed. For each of these identified limitations, a specific future development will be proposed to increase the TRL of the project. This approach aims to trace a roadmap that will lead the current numerical simulation towards hardware validations and control models that are increasingly closer to the actual operating conditions in orbit.

6.1. From Discrete to Continuous Control Logic

A primary area for future enhancement concerns the transition from the current discrete control architecture to a more robust, continuous formulation. As identified is Section 4.4, the reliance on `IF` logic for hinge activation introduces significant spikes in torque trends derived from dynamic analysis on the parametric model. These transitions characterized by instantaneous switching between states are the root cause of the force spikes that currently necessitate a manual tuning of the `SFORCES`' PD gains to avoid problems during analyses. To address these limitations, a key development will involve the implementation of smooth transition functions available within the Adams environment. By replacing discrete conditional logic with a continuous approximation, the control torque can be ramped up over a finite time interval. This approach would effectively filter the "shocks", allowing the derivative component of the controller to dissipate energy more effectively without triggering numerical instability. Furthermore, the adoption of a continuous control strategy would pave the way for a truly autonomous parametric model, allowing the relation for optimal control gains (K, C) becoming more predictable, removing the need for user driven manual optimization. A practical implementation could include continuous functions already utilized in this model to substitute the `IF` logic, such as:

- `STEP5`

- TANH

The figure below illustrates the proposed control logic whit a STEP5 function that could be integrated to attenuate or completely eliminate the spikes.

```
- CURRENT: Discrete IF logic (causes force spikes) -
IF( SENVAL(SENSOR_disp1) : 0, 0,
        K_CAPT * ( JOINT_az - CUBSPL(time,0,S_AZ)) +
        C_CAPT * ( JOINT_wz - CUBSPL(time,0,S_WZ)))

- DEVELOPMENT: Continuous transition replacing the IF -
STEP5( time,  SENVAL(SENSOR_disp1), 0,
        SENVAL(SENSOR_disp1) + Delta_time,
        K_CAPT * ( JOINT_az - CUBSPL(time,0,S_AZ)) +
        C_CAPT * ( JOINT_wz - CUBSPL(time,0,S_WZ)))
```

Figure 6.1.: Comparison between the discrete conditional logic (IF) and its continuous formulation (STEP5) for torque definition.

6.2. Contact and collision dynamics

The transition from the rendezvous phase to the actual capture of space debris introduces critical issues related to contact dynamics. Unlike traditional rigid robotic systems, an origami mechanism has an intrinsic flexibility that is advantageous for the passive dissipation of kinetic energy during impact. However, these features also introduce a significant complexity if the stiffness of the hinges between the tiles is not modeled and controlled properly, as post-impact oscillations can compromise the stability of the capture. In previous work [15], a hybrid method for the actuation mechanism is proposed:

- An active base unit: consisting of five electric actuators that control the main degrees of freedom
- A tendon-driven system: dedicated to managing the upper levels, ensuring a high strength-to-weight ratio, minimizing masses.

This system is an excellent starting point given its great adaptability and lightness, which fits perfectly with the main principles of an origami-based mechanism. However, by including the capture phase in the study, certain precautions become necessary in order to avoid compromising the integrity of the mission or the mechanism itself during first impact with the target. The capture phase is indeed the most critical stage from a structural point of view, particularly for the actuators that in the previous study [15] are modelled by:

- **A motor**: provides the primary rotational input to the mechanism's joints.
- **A gearbox**: necessary to step down the motor's speed and deliver the high torque required to move the origami tiles and resist impact loads.

A high energy capture event, characterized by high impacts on the tiles, could not only damage the panels themselves but also cause the complete failure of the gear teeth inside the actuation system. This would compromise the proper

closure of the mechanism, leading to capture failure and, consequently, to mission loss. To mitigate this risk and ensure completely safe capture, a control architecture that allows active modulation of the stiffness (K) and damping (D) parameters of the PD control logic that governs the movements of the tiles is needed. To this purpose, two approaches have been identified which, depending on the mission TRL, can be adopted either separately or jointly.

- Hardware solution (mechanical protection): The integration of a clutch interposed between the gearbox output shaft and the actuator [33].
- Software solution (impedance control): the implementation of a software that control the PD (*Proportional-Derivative*) behavior depending on the scenario. [1]

Alongside these joint-level protection strategies, the physical interaction between the mechanism's tiles and the space debris must be carefully managed to ensure a stable grasp. Once the origami mechanism has fully enveloped the target, the final phase relies on a gradual deceleration rather than a rigid clamping action. To achieve this, the system exploits distributed micro-contacts along the inner surfaces of the tiles. By integrating dry adhesives, such as gecko inspired materials [34], the chaser can safely stabilize the target's tumbling motion, smoothly dissipating residual kinetic energy without applying excessive concentrated loads.

6.3. Target capture and mission phases

After validating the correct functioning of the model, a precise definition of the mission architecture is required, with particular attention to the mechanisms for transporting the chaser to the desired operational orbit and the right choice of strategy for approaching the target. Although the phasing operations of the chaser have already been described in [16], it is necessary to clearly define the previous phases, for two fundamental reasons:

- To size the chaser to ensure its resistance to the vibrational loads of the most critical phase: the launch.
- To define the return strategy to the mothership and the management of the combined system after the capture phase.

In the preliminary phase, it was also appropriate to consider the space shuttle as a mothership, but after the withdrawal of the program by NASA in 2011 [35], it is necessary to evaluate other options. One advantage of the mechanism under consideration is its high compactness during stowage, so, for launch mechanisms, some of the most commercially available options could be considered, such as SpaceX, Vega-C, or Ariane 6 [36]. Once the launch vehicle has been defined, the chaser release strategy must be defined, it can consist of two different philosophies, which in turn provide two different ways of optimally approaching the target. To provide a basis for further investigation, the following elements are briefly described below:

[1] [34] This paper proposes a robotic system combining hardware and software compliance to safely capture uncooperative space debris, which, if properly integrated, can be applied to the mechanism under consideration.

- Release into a lower orbit than that of space debris
- Release into a higher orbit than that of space debris

Both strategies are valid from a technical feasibility point of view, but as [37] explains, leaving the chaser in a lower orbit is much safer because it does not cross the orbit of the TARGET in the event of propulsion system failure. Therefore, being released into a lower orbit, the chaser will have a higher angular velocity than the target. Consequently, the phasing phase will require the chaser to begin operations from a rear position (behind the target), allowing the natural orbital kinematics to reduce the time gap. Only once it has reached the correct position relative to the debris, the chaser will perform orbital raising maneuvers to initiate proximity operations (Autonomous Rendezvous and Capture (AR&C)).

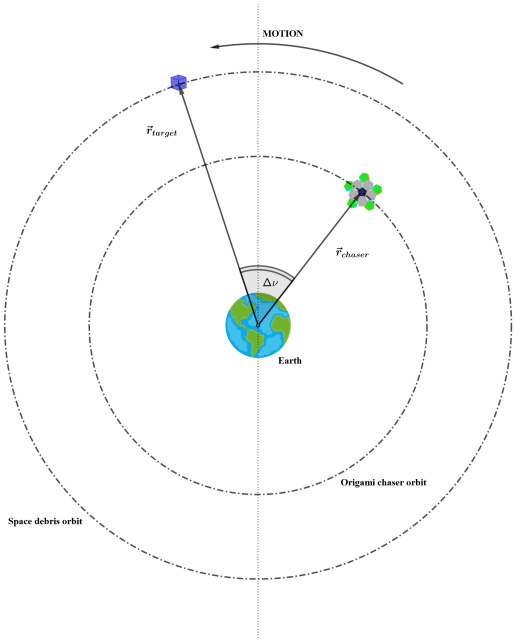


Figure 6.2.: Release orbit

As already mentioned in the paragraph, another topic of discussion arises: the contact side for satellite capture (front or back). The choice between a frontal approach (*V-bar forward*) or a rear approach (*V-bar backward*) represents a crucial compromise between the structural integrity of the target and fluid dynamic safety. On the one hand, the rear approach is often the preferred choice from a structural point of view and in terms of mission reliability, as even with a system failure, the risk of collision with space debris would be lower, avoiding the formation of further debris. On the other hand, the front approach would mitigate the risk associated with exhaust gases, but would increase the risk in the event of impact. However, although it might seem the most suitable option, the latter strategy has a huge disadvantage: a fly-around maneuver is required to allow the chaser (if arriving from a lower orbit) to position itself in front of the target, increasing propellant consumption (Δv). The choice could fall entirely on the first approach, integrating the effect of exhaust gases into the capture phase, which, although it may seem negative, can turn into an advantage, as demonstrated in [38].

7. Conclusions

Analyzing the obtained results, it can be concluded that the preliminary parameterization of the model from a geometric and kinematic point of view was extremely useful in overcoming the intrinsic limitations of a single case study, making the model flexible, scalable, and adaptable to different geometries of the `TARGET`.
The performed simulations further confirmed the validity of the origami concept, providing positive and consistent results when the `scaling_factor` of the model was varied. In fact, by analyzing the trends obtained for the quantities under examination, it can be stated that geometric and kinematic parameterization proved to be a fundamental tool for exploring the design space, allowing the strengths of the mechanism to be highlighted, validating its macroscopic behavior, and at the same time identifying critical issues.
This work has therefore provided a substantial contribution to the validation of the virtual prototype, establishing solid foundations for subsequent engineering developments and for the creation of a physical demonstrator, although not yet definitive. However, the transition from a purely kinematic-geometric model to a dynamic system operating in a realistic environment has brought to light several aspects to be explored in order to finalize a real space mission. In parallel with the final validation of the origami mechanism itself, a proper mission architecture should be defined and validated in order to determine the optimal allocation of onboard sensors, the optical payload for target reception and navigation, and to obtain accurate estimates of mass and volume budgets.
In summary, the parameterization and simulations conducted in this work confirm the strong potential of origami structures in space, while also outlining an initial multidisciplinary roadmap for transforming a mechanical concept into a safe and fully feasible mission.
As already discussed in the chapter on future developments, the calibration of the coefficients K and C for the PD control on the hinges in the parametric model with the resulting parametric analyses integrated into the Adams software and a clear definition of the space mission, represent the natural continuation of this study.

Appendices

A. Appendix A

A.1. Master.cmd

```
1
2  !****************************************************************
3  !                    COMMANDS DESCRIPTION
4  !****************************************************************
5
6  ! Macro Name:     [Master.cmd]
7  ! Last update:    [23/01/2026]
8
9  !****************************************************************
10 ! READS THE COMMANDS IN ORDER TO START DYNAMIC SIMULATION
11 !****************************************************************
12
13 !Command to enable all motion and disable all forces
14 file command read file="Commands/ACT_DEACT.cmd"
15
16 !Command to set all forces to zero.
17 file command read file="Commands/Forces_ZERO.cmd"
18
19 !Command to start simulation with MOTION activated.
20 file command read file="Commands/Simulation_Script_KIN.cmd"
21
22 !Command to generate SPLINES of joint position and angular velocity
23 file command read file="Commands/Spline.cmd"
24
25 !Command to create design variables containing sensor trigger
      timestamps
26 file command read file="Commands/Macro_Trig_SENSORS.cmd"
27
28 !Commands to create design variables containing maximun and minimum af
      MOTION results
29 file command read file="Commands/Macro1_P1E1.cmd"
30
31 [...]
32
33 ! Command to import all SFORCES functions
34 file command read file="Commands/Forces_IN.cmd"
35
36 !Command to enable all forces and disable all motion
37 file command read file="Commands/DEACT_ACT.cmd"
38
39 !Command to start dynamic simulation.
40 file command read file="Commands/Simulation_Script_DIN.cmd"
```

Listing A.1: Command file description

A.2. Findings of the maximum and minimum time and value

```
!CAPTURE - P2E3NE

! RICERCA TERZO MASSIMO
variable create &
    variable_name = MAX_p2e3ne_val_CAPT &
        real_value    = (MAX(CLIP(.MODEL_V3.MOTIONS_RESULTS.
            MOTION_p2_e3_ne_Torque.Q.values,.MODEL_V3.min_e1p2ne_time/(
            1E-02),.MODEL_V3.Time_Capture/(1E-02))))

variable create &
    variable_name = MAX_p2e3ne_time_CAPT &
        real_value    = (.MODEL_V3.min_e1p2ne_time+(1E-02*(MAXI(CLIP(.
            MODEL_V3.MOTIONS_RESULTS.MOTION_p2_e3_ne_Torque.Q.values,.
            MODEL_V3.min_e1p2ne_time/(1E-02),.MODEL_V3.Time_Capture/(
            1E-02))))))

! RICERCA TERZO MINIMO

variable create &
    variable_name = min_p2e3ne_val_CAPT &
        real_value    = (MIN(CLIP(.MODEL_V3.MOTIONS_RESULTS.
            MOTION_p2_e3_ne_Torque.Q.values,(.MODEL_V3.
            MAX_p2e3ne_time_CAPT/(1E-02)),.MODEL_V3.Time_Capture/(1E-02
            ))))

variable create &
    variable_name = min_p2e3ne_time_CAPT &
        real_value    = (.MODEL_V3.MAX_p2e3ne_time_CAPT+(1E-02)*MINI(
            CLIP(.MODEL_V3.MOTIONS_RESULTS.MOTION_p2_e3_ne_Torque.Q.
            values,(.MODEL_V3.MAX_p2e3ne_time_CAPT/(1E-02)),.MODEL_V3.
            Time_Capture/(1E-02))))
```

Listing A.2: Capture - p2e3ne

A.3. SFORCES formulation - Phase 1

Phase 1: Holding - Deployment

```
IF( SENVAL(.MODEL_V3.SENSOR_Z) : 0,
   ! Sensor that starts capture

   IF( SENVAL(.MODEL_V3.SENSOR_avv) :
   ! Sensor that starts deployment (SENSOR_Z = OFF)
    0,


    IF( SENVAL(.MODEL_V3.SENSOR_disp1) :
    ! DEPLOYMENT PHASE (Sensor Disp1 = OFF)
        0,

        ! 1. Transition (STEP5)
         STEP5(time, .MODEL_V3.MAX_time, 0, .MODEL_V3.min_time, ... ) +

        ! 2.  PD + CUBSPL
         - ( .MODEL_V3.K_DEPL * ( .MODEL_V3.JOINT_az - CUBSPL(time, 0, .MODEL_V3.S_AZ) ) +
             .MODEL_V3.C_DEPL * ( .MODEL_V3.JOINT_wz - CUBSPL(time, 0, .MODEL_V3.S_WZ) )
    ) +
         ... ,

         STEP5(time,
            SENVAL(.MODEL_V3.SENSOR_disp1), 0,         ! Start: Trigger time SENSOR_disp1
            SENVAL(.MODEL_V3.SENSOR_disp1) + Delta\_time,      ! End: Trigger +
    Delta\_Time

      ! Holding phase 1 - Deployment completed
      .MODEL_V3.K_CAPT * ( .MODEL_V3.JOINT_az - CUBSPL(time, ... ) ) +
      .MODEL_V3.C_CAPT * ( .MODEL_V3.JOINT_wz - CUBSPL(time, ... ) )
           )
      )
      ...,
   )
   ...,
)
```

Listing A.3: SFORCE DEPLOYMENT PHASE

A.4. SFORCES formulation - Phase 2

Phase 2: Holding - Capture - Holding

```
IF( SENVAL(.MODEL_V3.SENSOR_capt0) : 0,
   ! Capture not started → smooth engagement after deployment

   STEP5(time, SENVAL(.MODEL_V3.SENSOR_disp1), 0,
         SENVAL(.MODEL_V3.SENSOR_disp1) + Delta_time,

      .MODEL_V3.K_CAPT * ( .MODEL_V3.JOINT_az - CUBSPL(time,0,.MODEL_V3.S_AZ) ) +
      .MODEL_V3.C_CAPT * ( .MODEL_V3.JOINT_wz - CUBSPL(time,0,.MODEL_V3.S_WZ) )
   ),

   IF( SENVAL(.MODEL_V3.SENSOR_capt1) : 0,
      ! Capture Stage 1 → holding with CAPT PD

      STEP5(time, SENVAL(.MODEL_V3.SENSOR_capt0), 0,
            SENVAL(.MODEL_V3.SENSOR_capt0) + Delta_time,

         .MODEL_V3.K_CAPT * ( .MODEL_V3.JOINT_az - CUBSPL(time,0,.MODEL_V3.S_AZ) ) +
         .MODEL_V3.C_CAPT * ( .MODEL_V3.JOINT_wz - CUBSPL(time,0,.MODEL_V3.S_WZ) )
      ),

      IF( SENVAL(.MODEL_V3.SENSOR_capt2) : 0,
         ! Capture transient phase

         STEP5(time, SENVAL(.MODEL_V3.SENSOR_capt1), ..., .MODEL_V3.MAX_time_CAPT, ...) +
         STEP5(time, .MODEL_V3.MAX_time_CAPT, ..., .MODEL_V3.min_time_CAPT, ...) +

         ! PD tracking during transient
         STEP5(time, ..., ..., ...,
            .MODEL_V3.K_CAPT * ( .MODEL_V3.JOINT_az - CUBSPL(time,0,.MODEL_V3.S_AZ) ) +
            .MODEL_V3.C_CAPT * ( .MODEL_V3.JOINT_wz - CUBSPL(time,0,.MODEL_V3.S_WZ) )
         ) + ... ,

         ! Final holding (capture completed)

         STEP5(time, SENVAL(.MODEL_V3.SENSOR_capt2), 0,
               SENVAL(.MODEL_V3.SENSOR_capt2) + Delta_time,

            .MODEL_V3.K_CAPT * ( .MODEL_V3.JOINT_az - CUBSPL(time,0,.MODEL_V3.S_AZ) ) +
            .MODEL_V3.C_CAPT * ( .MODEL_V3.JOINT_wz - CUBSPL(time,0,.MODEL_V3.S_WZ) )
         )
      )
   )
)
```

Listing A.4: SFORCE CAPTURE PHASE

A.5. Simulation script

Dynamic simulation script

```
! Insert ACF commands here:
!START DISPLACEMENT
SIMULATE/DYNAMIC, END=600, DTOUT=1.0E-02
DEACTIVATE/SENSOR, ID=2
!START DISPLACEMENT LEVEL 2
SIMULATE/DYNAMIC, END=600, DTOUT=1.0E-02
DEACTIVATE/SENSOR, ID=15
!START DISPLACEMENT LEVEL 3
SIMULATE/DYNAMIC, END=600, DTOUT=1.0E-02
DEACTIVATE/SENSOR, ID=6,10,11,28,29,40,42,44,46,48,50
!START DISPLACEMENT LEVEL 4
SIMULATE/DYNAMIC, END=600, DTOUT=1.0E-02
DEACTIVATE/SENSOR, ID=7,12
!START DISPLACEMENT LAST PENTAGON
SIMULATE/DYNAMIC, END=600, DTOUT=1.0E-02
DEACTIVATE/SENSOR, ID=16
! START CAPTURE
SIMULATE/DYNAMIC, END=600, DTOUT=1.0E-02
DEACTIVATE/SENSOR, ID=1
SIMULATE/DYNAMIC, END=600, DTOUT=1.0E-02
DEACTIVATE/SENSOR, ID=52,53,54,55,56
! CAPTURE LEVEL 1
SIMULATE/DYNAMIC, END=600, DTOUT=1.0E-02
DEACTIVATE/SENSOR, ID=17,18,24,31,36,37
SIMULATE/DYNAMIC, END=600, DTOUT=1.0E-02
DEACTIVATE/SENSOR, ID=13,26,33,34,35
! CAPTURE LEVEL 2
SIMULATE/DYNAMIC, END=600, DTOUT=1.0E-02
DEACTIVATE/SENSOR, ID=8,14,25,32,38,39
! CAPTURE LEVEL 3
SIMULATE/DYNAMIC, END=600, DTOUT=1.0E-02
DEACTIVATE/SENSOR, ID=9,19,23,27,30,41,43,45,47,49,51
! CAPTURE LEVEL 4 EXAGON 2
SIMULATE/DYNAMIC, END=600, DTOUT=1.0E-02
DEACTIVATE/SENSOR, ID=20
! CAPTURE LAST LEVEL
SIMULATE/DYNAMIC, END=600, DTOUT=1.0E-02
DEACTIVATE/SENSOR, ID=21,22
SIMULATE/DYNAMIC, END=600, DTOUT=1.0E-02
```

Listing A.5: Dynamic simulation

Bibliography

[1] European Space Agency. *About space debris*. The European Space Agency. URL: https://www.esa.int/Space_Safety/Space_Debris/About_space_debris (visited on 12/31/2025).

[2] European Space Agency. *The current state of space debris*. The European Space Agency. URL: https://www.esa.int/Space_Safety/Space_Debris/The_current_state_of_space_debris (visited on 12/31/2025).

[3] BusinessCom Networks. *Starlink and Space Junk*. Accessed: 2024-05-20. 2023. URL: https://www.bcsatellite.net/blog/starlink-and-space-junk/ (visited on 01/01/2023).

[4] Heather F. Riley. *Micrometeoroids and Orbital Debris (MMOD)*. Accessed: 2026-01-01. 2016. URL: https://www.nasa.gov/centers-and-facilities/white-sands/micrometeoroids-and-orbital-debris-mmod/.

[5] European Space Agency. *The Kessler Effect and How to Stop It*. Accessed: 2026-01-01. URL: https://www.esa.int/Enabling_Support/Space_Engineering_Technology/The_Kessler_Effect_and_how_to_stop_it.

[6] European Space Agency. *ESA Space Environment Report 2025*. Report. Accessed: 2025-12-18. ESA Space Debris Office, Apr. 2025. URL: https://www.esa.int/Space_Safety/Space_Debris/ESA_Space_Environment_Report_2025.

[7] *Orbital Debris Quarterly News* 29.3 (Sept. 2025). Volume 29, Issue 3. URL: https://orbitaldebris.jsc.nasa.gov/quarterly-news/.

[8] ESA Space Debris Office. *ESA's Annual Space Environment Report*. Document Type LOG GEN-DB-LOG-00288-OPS-SD. Issue/Revision 9.1. European Space Agency (ESA), Oct. 2025. URL: https://www.sdo.esoc.esa.int/publications/Space_Environment_Report_I9R1_20251021.pdf.

[9] *Orbital Debris Quarterly News* 11.2 (Apr. 2007). Volume 11, Issue 2. URL: https://orbitaldebris.jsc.nasa.gov/quarterly-news/.

[10] Alessandro Galletti. *AESASpazio. L'arte degli origami nello spazio*. Accessed: 2026-01-03. AESA Torino. 2024. URL: https://aesatorino.altervista.org/aesaspazio-larte-degli-origami-nello-spazio/.

[11] Junhao Wu et al. "Origami principle in space deployable membrane structures: Mechanism, application and prospects". In: *Chinese Journal of Aeronautics* (2025). Article in press. DOI: 10.1016/j.cja.2025.103831.

[12] Manan Arya et al. "Demonstration of deployment repeatability of key subsystems of a furled starshade architecture". In: *Journal of Astronomical Telescopes, Instruments, and Systems* 7.2 (2021), p. 021202. DOI: 10.1117/1.JATIS.7.2.021202. URL: https://doi.org/10.1117/1.JATIS.7.2.021202.

[13] Washington State University. *Researchers use origami to solve space travel challenge*. Accessed: 04 January 2026. Dec. 2020. URL: https://news.wsu.edu/press-release/2020/12/15/researchers-use-origami-solve-space-travel-challenge/.

[14] Chong Sun, Wenya Wan, and Lisheng Deng. "Adaptive space debris capture approach based on origami principle". In: *International Journal of Advanced Robotic Systems* 16.6 (2019). DOI: 10.1177/1729881419885219.

[15] Giuseppe Balsamo. "Multibody analysis of a space debris capture mechanism based on origami principle". July 2025. URL: http://webthesis.biblio.polito.it/36768/.

[16] Gregorio Mastria. "Analisi Multi-Corpo e Progettazione di un meccanismo di cattura per detriti spaziali e IOS basato su principi origami". July 2024. URL: https://webthesis.biblio.polito.it/32263/.

[17] Eric W. Weisstein. *Truncated Icosahedron*. From MathWorld—A Wolfram Resource. Wolfram MathWorld. URL: https://mathworld.wolfram.com/TruncatedIcosahedron.html (visited on 02/27/2026).

[18] E. Yamaguchi. *Basic Theory of Plates and Elastic Stability*. Structural Engineering Handbook. Editor: Chen Wai-Fah. Chapter 1. CRC Press LLC. Boca Raton, 1999.

[19] Spencer Magleby et al. "Accommodating Thickness in Origami-Based Deployable Arrays". In: *Journal of Mechanical Design* 135 (Nov. 2013). DOI: 10.1115/1.4025372.

[20] Hexagon Manufacturing Intelligence. *Adams View User Guide*. 2023.3. MSC ADAMS Documentation. Hexagon. 2023. URL: https://nexus.hexagon.com/documentationcenter/en-US/bundle/Adams_2023.3_Adams_View_User_Guide/resource/Adams_2023.3_Adams_View_User_Guide.pdf.

[21] Hexagon Manufacturing Intelligence. *ADAMS Solver User Guide*. 2021.0.2. Pages XX–YY. Hexagon MI. 2021. URL: https://help-be.hexagonmi.com/bundle/Adams_2021.0.2_Adams_Solver_User_Guide/raw/resource/enus/Adams_2021.0.2_Adams_Solver_User_Guide.pdf.

[22] National Instruments. *The PID Controller Theory Explained*. 2026-01-28. 2025. URL: https://www.ni.com/en/shop/labview/pid-theory-explained.html.

[23] José Vasconcelos et al. "Close-Proximity Operations Design, Analysis, and Validation for Non-Cooperative Targets with an Application to the ClearSpace-1 Mission". In: *Aerospace* 12.1 (2025), p. 67. DOI: 10.3390/aerospace12010067. URL: https://www.mdpi.com/2226-4310/12/1/67.

[24] Yanmin Ren and Aijia Xing. "Finite-time attitude tracking control for spacecraft based on backstepping method with input saturation". In: *PLOS ONE* 20.6 (June 2025), pp. 1–16. DOI: 10.1371/journal.pone.0326150. URL: https://doi.org/10.1371/journal.pone.0326150.

[25] Alex Ellery. “Tutorial Review on Space Manipulators for Space Debris Mitigation”. In: *Robotics* 8.2 (2019). ISSN: 2218-6581. DOI: 10.3390/robotics8020034. URL: https://www.mdpi.com/2218-6581/8/2/34.

[26] Hexagon Manufacturing Intelligence. *Adams View Function Builder User Guide*. 2024.1. Accessed: January 2026. Hexagon. 2024. URL: https://nexus.hexagon.com/documentationcenter/en-US/bundle/Adams_2024.1_Adams_View_Function_Builder_User_Guide/resource/Adams_2024.1_Adams_View_Function_Builder_User_Guide.pdf.

[27] Harvey Gomez and Toralf Boge. “Relative Navigation Determination in the Capture of Non-Cooperative Targets”. In: 6.1 (2013). Ed. by L. Ouwehand. URL: https://conference.sdo.esoc.esa.int/proceedings/sdc6/paper/76.

[28] National Ecological Observatory Network (NEON). *The Basics of LiDAR*. Accessed:February 2026. 2024. URL: https://www.neonscience.org/resources/learning-hub/tutorials/lidar-basics.

[29] Gaopeng Zhao, Sixiong Xu, and Yuming Bo. “LiDAR-Based Non-Cooperative Tumbling Spacecraft Pose Tracking by Fusing Depth Maps and Point Clouds”. In: *Sensors* 18.10 (2018). ISSN: 1424-8220. DOI: 10.3390/s18103432. URL: https://www.mdpi.com/1424-8220/18/10/3432.

[30] I. Lopez-Calle. “Space Debris In-Orbit Detection with Commercial Automotive LiDAR Sensors”. In: *Sensors* 24.22 (2024), p. 7293. DOI: 10.3390/s24227293.

[31] A A Andryakov, D V Malygin, and A R Khabibulin. “Development of lidar system based on nanosatellite for composition of 3D space debris map”. In: *Journal of Physics: Conference Series* 1326.1 (Oct. 2019), p. 012021. DOI: 10.1088/1742-6596/1326/1/012021. URL: https://doi.org/10.1088/1742-6596/1326/1/012021.

[32] Thomas Chabot et al. “Vision-Based Navigation Experiment Onboard the RemoveDebris Mission”. In: *Proceedings of the 10th International ESA Conference on Guidance, Navigation and Control Systems (GNC 2017)* (2017). Salzburg, Austria, May 2017. HAL Id: hal-01784234, pp. 1–23. URL: https://inria.hal.science/hal-01784234v1/file/2017_gnc_chabot.pdf.

[33] D. L. Kirkpatrick. “The development and test of a long-life, high reliability solar array drive actuator”. In: *The 8th Aerospace Mechanisms Symposium, NASA Langley Research Center* (Oct. 1973). Document ID: 19740003561, Accession Number: 74N11674. URL: https://ntrs.nasa.gov/citations/19740003561.

[34] Maxime Hubert Delisle et al. “Hybrid-Compliant System for Soft Capture of Uncooperative Space Debris”. In: *Applied Sciences* 13.13 (2023). ISSN: 2076-3417. DOI: 10.3390/app13137968. URL: https://www.mdpi.com/2076-3417/13/13/7968.

[35] Astronomy.com. *Why Did NASA Retire the Space Shuttle?* 2020. URL: https://www.astronomy.com/space-exploration/why-did-nasa-retire-the-space-shuttle/ (visited on 02/28/2026).

[36] European Space Agency. *Europe's launchers*. URL: https://www.esa.int/Enabling_Support/Space_Transportation/Europe_s_launchers2 (visited on 02/28/2026).

[37] Brent William Barbee et al. "Guidance and Navigation for Rendezvous and Proximity Operations with a Non-Cooperative Spacecraft at Geosynchronous Orbit". In: (May 2010). Document ID: 20100019266.

[38] G. Borelli, G. Gaias, and C. Colombo. "Guidance and Control for Safe Contactless Plume Impingement Operations to Detumble an Uncooperative Spacecraft". In: *Aerospace* 11 (2024), p. 224. DOI: 10.3390/aerospace11030224.