Automation of Cable Routing for Harness Design in Satellites

Master thesis

Presented to the Faculty 04 - Production Engineering

Universität Bremen

In the partial fulfillment of the requirement for the degree of

Master of Science

In

SPACE ENGINEERING

By

Lillian Rodrigues

6176831

Supervisor:
PD Dr. rer. nat. Sven Herrmann
ZARM
Zentrum für angewandte Raumfahrttechnologie und Mikrogravitation (ZARM)
Bremen, Germany

Supervisor:
Dr. Marco Scharringhausen
DLR
Deutsches Zentrum für für Luft- und Raumfahrt (DLR)
Bremen, Germany

February, 2026

Declaration of Copyright

Universität Bremen

Eigenständigkeits- und Einverständniserklärung zur Überprüfung mit Plagiatssoftware sowie die Erklärung zur Veröffentlichung bei Bachelor- und Masterarbeiten

Declarations of Authorship and Consent for Checking with Plagiarism Software and the Declaration of Publication for Bachelor's and Master's Thesis

Studierenden-Angaben / Student Information:

Matrikelnr./ Student ID	6176831
Nachname / Surname	Rodrigues
Vorname / First Name	Lillian

Titel der Arbeit / Title of Thesis

Automatisierung der Kabelführung für die Kabelbaumentwicklung in Satelliten /
Automation of Cable Routing for Harness Design in Satellites

A) Eigenständigkeitserklärung / Declaration of Authorship

Ich versichere, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe. Alle Teile meiner Arbeit, die wortwörtlich oder dem Sinn nach anderen Werken entnommen sind, wurden unter Angabe der Quelle kenntlich gemacht. Gleiches gilt auch für Zeichnungen, Skizzen, bildliche Darstellungen sowie für Quellen aus dem Internet, dazu zählen auch KI-basierte Anwendungen oder Werkzeuge.
Die Arbeit wurde in gleicher oder ähnlicher Form noch nicht als Prüfungsleistung eingereicht.

I hereby affirm that I have written the present work independently and have used no sources or aids other than those indicated. All parts of my work that have been taken from other works, either verbatim or in terms of meaning, have been marked as such, indicating the source. The same applies to drawings, sketches, pictorial representations and sources from the Internet, including AI-based applications or tools. The work has not yet been submitted in the same or a similar form as a final examination paper.

☒ Ich habe KI-basierte Anwendungen und/oder Werkzeuge genutzt und diese im Anhang "Nutzung KI basierte Anwendungen" dokumentiert.

I have used AI-based applications and/or tools and documented them in the appendix "Use of AI-based applications".

B) **Erklärung zur Veröffentlichung von Bachelor- oder Masterarbeiten**
Declaration regarding the publication of bachelor's or master's thesis

Die Abschlussarbeit wird zwei Jahre nach Studienabschluss dem Archiv der Universität Bremen zur dauerhaften Archivierung angeboten. Archiviert werden:

Two years after graduation, the thesis is offered to the archive of the University of Bremen for permanent archiving. The following are archived:

1) Masterarbeiten mit lokalem oder regionalem Bezug sowie pro Studienfach und Studienjahr 10 % aller Masterarbeiten
Master's theses with a local or regional focus, as well as per subject and academic year 10% of all Master's thesis

2) Bachelorarbeiten des jeweils ersten und letzten Bachelorabschlusses pro Studienfach und Jahr.
Bachelor's thesis for the first and last bachelor's degrees per subject and year.

☒ Ich bin damit einverstanden, dass meine Abschlussarbeit im Universitätsarchiv für wissenschaftliche Zwecke von Dritten eingesehen werden darf.
I agree that my thesis may be viewed by third parties in the university archive for academic purposes.

☒ Ich bin damit einverstanden, dass meine Abschlussarbeit nach 30 Jahren (gem. §7 Abs. 2 BremArchivG) im Universitätsarchiv für wissenschaftliche Zwecke von Dritten eingesehen werden darf.
I agree that my thesis may be viewed by third parties for academic purposes in the university archive after 30 years (in accordance with §7 para. 2 BremArchivG).

☐ Ich bin **nicht** damit einverstanden, dass meine Abschlussarbeit im Universitätsarchiv für wissenschaftliche Zwecke von Dritten eingesehen werden darf.
I do not consent to my thesis being made available in the university archive for third parties to view for academic purposes.

C) Einverständniserklärung zur elektronischen Überprüfung der Arbeit auf Plagiate
Declaration of consent for electronic checking of the work for plagiarism

Eingereichte Arbeiten können nach § 18 des Allgemeinen Teil der Bachelor- bzw. der Masterprüfungsordnungen der Universität Bremen mit qualifizierter Software auf Plagiatsvorwürfe untersucht werden.
Zum Zweck der Überprüfung auf Plagiate erfolgt das Hochladen auf den Server der von der Universität Bremen aktuell genutzten Plagiatssoftware.

Submitted papers can be checked for plagiarism using qualified software in accordance with § 18 of the General Section of the Bachelor's or Master's Degree Examination Regulations of the University of Bremen. For the purpose of checking for plagiarism, the upload to the server is done using the plagiarism software currently used by the University of Bremen.

☒ Ich bin damit einverstanden, dass die von mir vorgelegte und verfasste Arbeit zum oben genannten Zweck dauerhaft auf dem externen Server der aktuell von der Universität Bremen genutzten Plagiatssoftware, in einer institutionseigenen Bibliothek (Zugriff nur durch die Universität Bremen), gespeichert wird.

I agree that the work I have submitted and written will be stored permanently on the external server of the plagiarism software currently used by the University of Bremen, in a library belonging to the institution (accessed only by the University of Bremen), for the above-mentioned purpose.

☐ Ich bin **nicht** damit einverstanden, dass die von mir vorgelegte und verfasste Arbeit zum o.g. Zweck dauerhaft auf dem externen Server der aktuell von der Universität Bremen genutzten Plagiatssoftware, in einer institutionseigenen Bibliothek (Zugriff nur durch die Universität Bremen), gespeichert wird.

I do not consent to the work I submitted and wrote being permanently stored on the external server of the plagiarism software currently used by the University of Bremen, in a library belonging to the institution (accessed only by the University of Bremen), for the above-mentioned purpose.

Das Einverständnis der dauerhaften Speicherung des Textes ist freiwillig. Die Einwilligung kann jederzeit durch Erklärung gegenüber der Universität Bremen, mit Wirkung für die Zukunft, widerrufen werden. Weitere Informationen zur Überprüfung von schriftlichen Arbeiten durch die Plagiatsoftware sind im Nutzungs- und Datenschutzkonzept enthalten. Diese finden Sie auf der Internetseite der Universität Bremen.

Consent to the permanent storage of the text is voluntary. Consent can be withdrawn at any time by making a declaration to this effect to the University of Bremen, with effect for the future. Further information on the checking of written work using plagiarism software can be found in the data protection and usage concept. This can be found on the University of Bremen website.

Mit meiner Unterschrift versichere ich, dass ich die obenstehenden Erklärungen gelesen und verstanden habe und bestätige die Richtigkeit der gemachten Angaben.

With my signature, I confirm that I have read and understood the above explanations and confirm the accuracy of the information provided.

10.02.2026

Datum / Date Unterschrit/ Signature

Acknowledgement

First and foremost, my sincere gratitude goes to my thesis supervisors, Dr. Herrmann and Dr. Scharringhausen. Without their guidance, support, and constructive feedback, this thesis would not have been possible.

Grateful acknowledgment is extended to Synera for the opportunity to work as a working student, for supporting my studies, and for encouraging the application and further development of the concepts encountered during my degree program. In particular, I thank Robert Ciomber for his support and technical guidance, which contributed to reliable progress over the past months.

I am especially grateful to Christian Strowik from the DLR Institute of Space Systems for generously sharing his incredible engineering experience. His insights, practical perspective, and depth of knowledge were invaluable and had a significant impact on the quality and relevance of this work.

Furthermore, I would also like to thank Marcos Pereira Campos for his continued support and encouragement throughout this journey.

Finally, heartfelt gratitude is reserved for my family, who, even from far away, have always been present through their encouragement. A special remembrance goes to my grandmother, whose love and guidance continue to accompany me, even after her passing.

"Look again at that dot. That's here. That's home. That's us."

– Carl Sagan, *Pale Blue Dot*

Abstract

The design of cable harnesses for satellite systems is a complex and time-consuming task, constrained by limited installation space, strict safety requirements, and increasing system complexity, and traditionally relies on manual CAD-based routing driven by expert knowledge and iterative adjustments, limiting flexibility during early design phases. This thesis addresses the Cable Harness Routing Problem (CHRP) by investigating the feasibility of an automated low-code routing workflow suitable for early-stage satellite design. A voxel-based methodology is proposed that integrates a CAD environment with graph-based path finding using the A* algorithm, sequential Design Space updates, and physics-based reconstruction through a spring–particle system with Raphos Physics within the Synera platform. The workflow incorporates connector orientations, routing restrictions near mounting regions, bundled harness handling, bending radius constraints, and category-based clearance rules to generate feasible routing paths in complex three-dimensional environments. The methodology was implemented and evaluated using a representative subset of the Eu:CROPIS spacecraft bus as a reference mission, enabling a controlled and realistic assessment under practical geometric and functional constraints. Compared to the manually designed reference harness, the automated routing demonstrated improved consistency in bending radius compliance and mounting interval distribution, at the cost of a limited increase in local intersections with the Non-Design Space. Although the generated routing exhibited a moderate increase in total length and mass, the automated workflow significantly reduced routing design time, completing all stages from geometry extraction to constraint verification in slightly over one hour, compared to approximately three and a half working days required for manual routing. These results indicate that voxel-based automated routing can provide a viable and efficient foundation for supporting harness layout decisions in early satellite design phases, while offering future extensions toward optimization and enhanced physical modeling.

Contents

List of Tables

List of Figures

List of Symbols

A	Cross-sectional area [mm²]
d	Damping factor
d_i	External diameter of each cable [mm]
E	Modulus of elasticity [MPa]
$f(n)$	Total estimated cost of the cheapest path through the node n
f_{BR}	User-defined factor for bend radius
$g(n)$	Cost from the start node to node n
$h(n)$	Estimated cost from n to the goal
k	Spring stiffness [N/mm]
L	Length [mm]
OD	Outside Diameter
R_{min}	Minimum bend radius [mm]
R_{target}	Reconstruction target bend radius [mm]
t_f	Fall time [µs]
t_r	Rise time [µs]
v_{new}	New velocity [mm/s]
v_{prev}	Previous velocity [mm/s]

Acronyms

ACO	Ant Colony Optimization
AI&T	Assembly, Installation, and Test
AIV	Assembly, Integration, and Verification
AWG	American Wire Gauge
CAD	Computer-Aided Design
CM	Center of Mass
CHRP	Cable Harness Routing Problem
DLR	Deutsches Zentrum für Luft- und Raumfahrt (German Aerospace Center)
DS	Design Space
ECSS	European Cooperation for Space Standardization
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
Eu:CROPIS	Euglena and Combined Regenerative Organic-Food Production in Space
GA	Genetic Algorithms
NASA	National Aeronautics and Space Administration
MOPSO/D	Multi-Objective Particle Swarm Optimization
MST	Minimum Spanning Tree
NDS	Non-Design Space
PRM	Probabilistic Roadmap Method
PSO	Particle Swarm Optimization
RAMIS	Radiation Measurement in Space
SCORE	Scalable On-board Computing
SHRH	Subgradient Harness Routing Heuristic
SPHRH	Shortest Paths Harness Routing Heuristic
TP	Twisted Pair
TSP	Twisted Shielded Pair
VR	Virtual Reality
ZARM	Zentrum für angewandte Raumfahrttechnologie und Mikrogravitation (Center Of Applied Space Technology And Microgravity)

1 Introduction

The design and routing of cable harnesses in satellites are critical to their structural integrity, functionality, and long-term operational success. Satellite harnesses account for a significant portion of the system's dry mass and can directly influence performance through their impact on weight, volume, and energy efficiency. Advances in this field are essential to address challenges such as minimizing cable lengths, reducing weight, mitigating electromagnetic interference, and ensuring mechanical reliability under extreme conditions of space operation [1; 2].

Cable harness design in satellites involves arranging power, signal, and data cables within the constrained environment of a spacecraft. This task is inherently multidisciplinary, incorporating principles of electrical engineering, mechanical engineering, and optimization algorithms. However, traditional manual methods remain widely used, which are labor-intensive and prone to errors, a critical concern in aerospace applications where failures can have catastrophic consequences [3].

Moreover, the growing complexity of satellite systems demands innovative and scalable solutions for automating cable routing processes, integrating virtual environments, and optimizing designs to achieve both reliability and cost efficiency [4]. These advances are necessary to meet the increasing demands of space missions while maintaining rigorous safety and performance standards.

Recent advances in this area include the development of automated tools and algorithms that reduce harness length, weight, and crosstalk between cables [5]. For example, approaches such as ant colony optimization and genetic algorithms have been developed to find near-optimal routing solutions [6; 1]. Complementary approaches, such as virtual reality environments, have also shown promise in enabling real-time design adjustments and concurrent planning of manufacturing processes [7]. Despite efforts to automate the complex task of routing cable harnesses in satellites, which involves diverse design, electrical, and manufacturing constraints, the process is not yet fully developed.

By leveraging advanced modeling techniques and automated design tools, the industry can address current challenges and pave the way for the next generation of high-performance satellite systems. Therefore, the motivation for this thesis lies in addressing the limitations of traditional cable routing methodologies and exploring how automation can streamline the design process, reduce costs, and improve the reliability of satellite systems.

To address these challenges, this work proposes an automated workflow for cable harness routing, combining algorithmic path-finding with physics-based reconstruction. In addition, this work makes use of Synera software [8], which offers capabilities for parametric Computer-Aided Design (CAD) and simulation, making it a promising tool for optimizing cable harness design. The methodology of this work involves a workflow a workflow that incorporates a path-finding approach using the A* search algorithm, which has been widely applied in the literature [9; 10; 4], to generate an initial route optimized for shortest length under environmental constraints. In a subsequent step, Raphos Physics, an integrated add-in within the Synera software environment, is employed to reconstruct the cable paths with realistic physical behavior. Raphos Phycics is based on a spring–particle system, and its solver relies on a state-of-the-art projection method [11], allowing simulation of cable bending and deformation.

In order to demonstrate the proposed methodology, the Eu:CROPIS satellite will be used as a case study. The Eu:CROPIS (Euglena and Combined Regenerative Organic-Food Production in Space) satellite is a mission by the German Aerospace Center (DLR) designed and launched to test biological life support systems in space, making it a relevant context for evaluating harness design under mission-critical constraints.

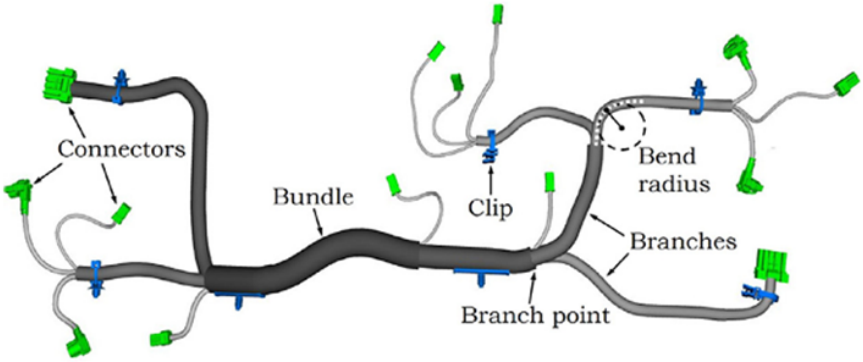
This thesis is structured as follows:

- **Chapter 2** presents the scientific background and motivation for the research. It introduces the fundamentals of spacecraft harness design, discusses key challenges, reviews the state of the art in cable routing methods, and provides an overview of the Eu:CROPIS mission, which serves as the case study.
- **Chapter 3** defines the objectives of the thesis, outlining the specific goals and scope of the research.
- **Chapter 4** details the proposed methodology for automated cable harness routing. It explains the workflow integrating the A* search algorithm for initial path finding and the Raphos Physics add-in for physics-based reconstruction within the Synera software environment.
- **Chapter 5** applies the developed methodology to the Eu:CROPIS case study, demonstrating its implementation in a real satellite context.
- **Chapter 6** presents and discusses the results of the case study, evaluating performance in terms of routing efficiency, design constraints, and practical applicability.
- **Chapter 8** concludes the thesis by summarizing the main findings, highlighting contributions, and suggesting directions for future research.

2 Scientific Background and Motivation

The harness within a spacecraft is a fundamental and critical subsystem [12; 13]. Its primary role is the distribution of electrical power and signals throughout the satellite [14]. It functions as an intricate assembly of wires, cables, and connectors, providing the necessary electrical interconnections. Serving as the carrier network for electrical and information interaction, the harness connects satellite subsystems and various sections [15]. An item of harness can be divided into bundles and branches, whereas connectors are physical elements that are used to access or connect electrical interfaces. In Figure 1, a harness item can be analyzed with its components.

Figure 1: Example of a harness item and its terminology [4].



For a whole satellite considering its multiple harness items, with all the connections and cables, it is not surprising that the harness appears very complex, as illustrated in Figure 2 with some examples. In a large satellite, the harness may require more than 50000 connections for about 1000 connectors linked by 20 km of wires [12].

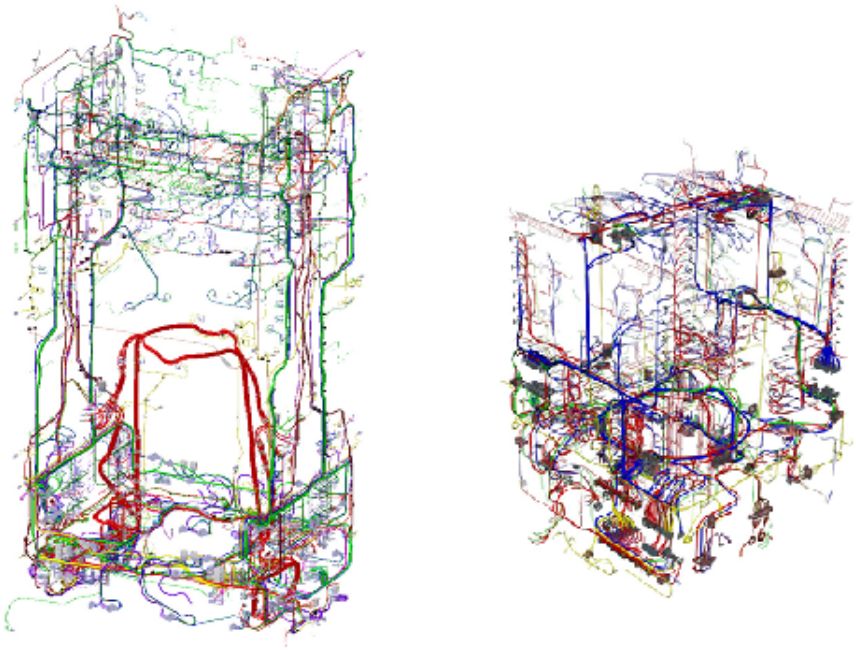
Moreover, designing cable harnesses for spacecraft is a complex, time-consuming, and costly process [6]. This complexity arises from the numerous parameters that must be considered [12]. The design must ensure the uninterrupted and undistorted flow of power and data, while also strictly adhering to volume and mass constraints [1]. As a result, the cable harness often represents a significant proportion of the overall product cost [12].

The design of the harness subsystem has largely depended on the expertise of the designers and iterative trial-and-error methods during the Assembly, Integration, and Verification (AIV) phase of the satellite. This approach can occasionally result in suboptimal configurations, leading to complications during subsystem integration, challenges associated with mass and volume limitations, and potential disruptions or signal distortions within the satellite [1].

Another challenge is that the cable harness is frequently one of the last items considered in the spacecraft configuration design. This can lead to design adaptations that are not fully optimized. For proper optimization, the involvement of the harness engineer in the early stages, particularly in locating electronic and electrical units on the spacecraft, is considered important [14; 16]. The integration of automation in the early design stages could therefore enable iterative re-optimization whenever subsystem layouts change, reducing rework and improving overall design efficiency

The harness design timeline consistently lies on the project's critical path, as it is the last subsystem to reach design maturity, because it requires all electrical interfaces and spacecraft accommodation to be finalized beforehand. On the other hand, it is among the first components

Figure 2: 3D Harness Models in Satellites: Astrium's Satcom provided by Malagoli, M., & Cosquéric, L. (2013) [12] on the left side; and a second example provided by Keskin, E. et al. (2024) [13] on the right side.



to be installed on the satellite, since electrical testing depends on the presence of the harness [12].

The challenges in spacecraft harnessing extend beyond initial design, some of them are the following:

- Mass Contribution: Cable harnesses are a significant contributor to satellite mass, accounting for approximately 10% of the dry mass of a spacecraft [13; 17]. For high-orbit satellites, the overall structure can account for about 20% to 30% of the mass [18]. The value of the mass of the harness for large satellite may exceed 100kg [12]. Therefore, mass reduction is an important way to optimize the harness, which is critical particularly for small satellites, with strict mass requirements derived from the launcher [19].
- Electromagnetic Compatibility (EMC): Cable harnesses significantly impact the electromagnetic environment of a spacecraft. Preventing electromagnetic interference (EMI) between signals is essential [14; 19; 13]. This is addressed through careful cable shielding, grounding strategies, and routing practices that minimize coupling between power and signal lines.
- Environmental Factors: Spacecraft harnesses must withstand extreme environments, including thermal variations and radiation. Specific wire insulation materials are selected based on their resistance to these conditions. Besides that, temperature requirements must be taken into consideration [14; 19].
- Mechanical Stress and Vibration: Harnesses experience mechanical stress and vibration, particularly during launch and orbital operations [13; 20]. They should be mounted accordingly. Materials must have good resistance to cold flow, cut-through, and abrasion to prevent damage from lacing, clamping, and vibration [14].
- Reliability and Failure Modes: Possible failure modes include open circuit, short circuit,

and high resistance, which can be complete or partial, permanent or intermittent [14]. Such failures are rarely caused by a single factor. They usually result from the combined effects of mechanical loads, thermal cycling, and electrical stresses acting over the operational lifetime of the system [20].

- Assembly, Installation, and Test (AI&T): Harness fabrication, installation, and testing are vulnerable phases. Damage during manufacturing is easier to address than damage to an installed harness. However, accessibility during installation is a common fault in design [14]. To mitigate the risk of damage, well-defined maintenance procedures and proper training are essential [20]. Due to the inherent flexibility of wires, robotic automation in harness assembly remains challenging [15]. Moreover, comprehensive testing is necessary to confirm the mechanical and electrical reliability of the harness [14].

Besides all these challenges, the increasing complexity and power needs of scientific and commercial spacecraft [12; 18] coupled with the demanding requirements for design and manufacturing, highlight the need to optimize satellite harness [1; 12]. Traditional design methods, often relying on physical mock-ups, are expensive, time-consuming, and test only a limited number of designs [6]. Therefore, automation of design and routing is seen as a way to reduce lead time, fulfill design constraints, and optimize the design.

The motivation for this thesis stems directly from these identified problems. The challenges of minimizing mass while ensuring reliability and facilitating manufacturing constraints, all within tight schedule and cost constraints, demand improved approaches to harness design and routing. Existing methods, whether manual, CAD-based, or early automation efforts, have limitations in addressing these intertwined requirements in an efficient and integrated manner.

While this thesis does not aim to solve all challenges, it focuses on advancing the investigation on advanced methodologies for cable harness routing in satellites. The focus lies on optimizing routing with respect to length, and, respectively, mass, while simultaneously developing a method to integrate complex constraints such as bundling, physical accessibility for mounting regions and minimum bend radius for realistic reconstruction. The aim is to contribute to overcoming the current challenges by enabling more efficient, robust, and optimized harness designs for future space missions.

A deeper understanding of the complexities involved in spacecraft harnessing requires reviewing the current state of the art in satellite cable routing. Section 2.1 therefore examines existing approaches, ranging from graph-based formulations and optimization algorithms to CAD-based design methods, as well as the standards that guide harness design in space applications. Following this, Section 2.2 provides an overview of the Eu:CROPIS mission, with particular focus on harness design and integration aspects relevant to this thesis.

2.1 State of the Art of Cable Routing in Satellites

Satellite cable harness routing involves meticulously planning bundle paths, identifying branch point locations, and adhering to numerous design and manufacturing specifications [4]. This process has evolved from labor-intensive manual methods to optimization techniques.

Historically, cable harness design has heavily relied on the experience and intuition of designers, often employing trial-and-error techniques during AIV phase [1]. This typically involves constructing physical mockups to test various designs [6]. While this allows direct, hands-on verification and assessment of designs, the method presents several disadvantages:

- It is time-consuming and costly, especially due to the need for multiple physical prototypes and revisions [4].
- It often leads to suboptimal designs, causing integration challenges, mass/volume issues, and potential signal distortions [1].
- A high rate of human error during assembly contributes to defects and delays [15].
- Debugging and reworking complex, interconnected harnesses can be difficult. This may require complete removal of the entire harness [16].
- The reliance on individual experience can lack consistency in optimization across different projects [13].

With advances in sophisticated algorithm computing and software design, significant efforts have focused on automating and optimizing harness routing to overcome the inherent limitations of manual approaches.

2.1.1 Graph-Based Approaches

Conru (1994) [6] described the Cable Harness Routing Problem (CHRP) by a cable path that can be determined by finding a route through a predefined graph. This method discretizes the 3D routing space into a network of nodes and edges (a graph) in order to find the path.

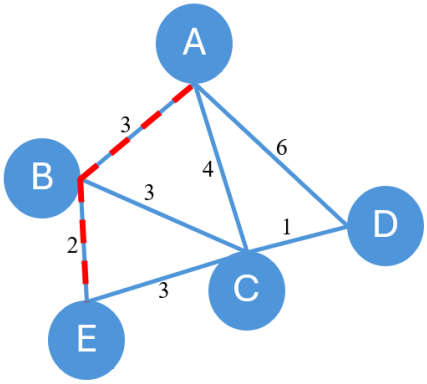
Conru (1994)'s [6] work uses Dijkstra's algorithm [21], which is a method to find the shortest path from a starting node to all other nodes in a weighted graph with non-negative edge weights. It uses a priority queue to always extend the path from the closest unvisited node. For example, in Figure 3, Dijkstra's algorithm starts from node A to reach the node E. The red path highlights the shortest path, which goes through B. The edge weights show that this path has a total cost of 3 (A-B) + 2 (B-E) = 5, which is shorter than any alternative. Thus, Dijkstra's algorithm explores neighboring nodes with the lowest cumulative cost first, updating distances until the shortest paths to all nodes are found.

A* (A-star) search is another path-finding algorithm used in the literature [9; 10; 4]. It is an extension of Dijkstra's algorithm and was introduced by Hart et al. (1968) [22] . A* achieves better performance by using heuristics to guide its search. This algorithm enhances efficiency by using a heuristic function to estimate the cost from any given node to the target destination. It is effective for identifying shortest paths [4]. Thus, the shortest path from a start node to a goal node is calculated using both the actual cost to reach a node ($g(n)$) and a heuristic estimate of the cost to reach the goal from that node ($h(n)$):

$$f(n) = g(n) + h(n) \tag{1}$$

Where ($g(n)$) is the known cost from the start node to node n, ($h(n)$) is the estimated cost from n to the goal (heuristic), and ($f(n)$) is the total estimated cost of the cheapest path through n.

Figure 3: Dijkstra's algorithm for finding the shortest path between A and E.



In the literature, related works have presented strategies for using these algorithms to incorporate additional constraints. Jin et al. (2022) [5] presented a modified Dijkstra's algorithm to incorporate specific constraints, such as limiting the "maximum overlap length" to control electromagnetic crosstalk. However, without specific modifications, basic shortest-path algorithms do not inherently address complex issues like crosstalk or bundling [5]. One strategy that has been presented is using optimization algorithms. Even though they are not used for in the development of this thesis, they are briefly described in the following subchapter.

2.1.2 Optimization Algorithms

Inspired by natural processes, these algorithms are designed to explore vast search spaces to find optimal or near-optimal solutions. Related works have explored Genetic Algorithms (GA), Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO).

Neverthless, Ng et al. (2000) [7] highlighted that the automation of harness cable routing is too open-ended [10; 7], as capturing design intent is difficult and human input is often necessary. A survey of industrial companies presented by Ng et al. (2000) [7] showed that there was a need for human expert intervention to make fine adjustments and verify solutions [3; 7]. Thus, while powerful, nature-inspired algorithms may fall short in providing reliable, reproducible, and sufficiently optimal solutions for highly complex and constrained problems like cable harness routing without extensive problem-specific adaptation and meticulous tuning [10].

Despite their capability to generate routing layouts, these algorithms generally fail to provide short-length solutions that simultaneously satisfy all design constraints, such as minimum bending radii, mounting regions, and accessibility. For this reason, they were not adopted in the present work, which prioritizes reproducibility and practical integration into CAD environments.

2.1.2.1 Genetic Algorithms (GA)

Conru (1994) [6] presented the CHRP decomposed into two sub-problems: generating the harness configuration (topology) and then routing it within the environment. Dijkstra's algorithm performs efficient, low-level path-finding and cost calculation for individual wires, which then feeds into the fitness evaluation that guides the higher-level GAs in their search for optimal harness configurations and transition placements through mutation and crossover operations. This method demonstrated a consistent performance improvement and could generate multiple routing options for designers to evaluate. However, it can be slower than purely heuristic-based methods. Besides, it is challenging to estimate how close the found solutions are to the true global optimum, as the optimal solution is often unknown (Conru, 1994).

Zhao et al. (2021)[23] showed a novel method for multi-branch cable harness layout design by uniting the Probabilistic Roadmap Method (PRM) and GA. This integrated approach unfolds in two main stages: first, the PRM constructs a non-interference roadmap attached to environmental surfaces, defining the underlying routing space. Second, an improved Genetic Algorithm is central to designing the topological branch structure and determining the positions of the branch points, utilizing specialized encoding and dynamic probabilities for enhanced efficiency. After the GA establishes the branch structure, each individual cable harness branch route is then searched using the PRM with Dijkstra's algorithm, adhering to area constraints and subsequently adjusted for minimum bending radii. Despite its advancements, this method may face scalability or complexity challenges when applied to large-scale systems. Additionally, while an improved Genetic Algorithm is proposed, there is still potential for further optimization of its parameters to enhance algorithmic efficiency.

2.1.2.2 Ant Colony Optimization (ACO)

Komninou et al. (2011) [1] formulated harness routing as a multi-objective optimization problem, with goals to minimize length and maximize bundling. Digital "ants" explore a discretized space, using heuristics based on cable length and bundling criteria. This method enables automated design of optimal harness layouts, effective for both length and bundling optimization. However, the effectiveness of bundling optimization can be sensitive to the specific application of heuristics. Requires further research into incorporating more complex constraints and testing with a larger number of cables.

2.1.2.3 Particle Swarm Optimization (PSO)

Zhang et al. (2021) [24] applied a decomposition-based Multi-Objective Particle Swarm Optimization (MOPSO/D) to optimize the layout of aircraft multi-branch cable harnesses. Each PSO particle encodes the number and positions of Steiner points (branch points) and candidate locations. Particle fitness is assessed by constructing a Minimum Spanning Tree (MST) with a modified Dijkstra's algorithm, optimizing objectives such as total cable weight, mesh unit count, and openness, while considering bending radii and fixed-surface segments. The method benefits from simplicity and fast convergence but suffers from non-reproducibility due to its stochastic nature, which is a limitation also noted by Karlsson et al. (2024) [4].

Karlsson et al. (2024) [4] introduced deterministic algorithms—the Subgradient Harness Routing Heuristic (SHRH) and -Shortest Paths Harness Routing Heuristic (-SPHRH)—to solve the CHRP as a multi-objective mixed-integer linear program. Their methods integrate environmental constraints and optimize for minimal cable length, maximal bundling, and spatial efficiency. Compared to PSO, which they implemented for benchmarking, SHRH and -SPHRH showed superior performance, faster convergence, and reproducibility.

Despite these advantages, the deterministic approaches still exhibit certain limitations.They can present notable duality gaps, limited ability to find all Pareto optimal solutions due to the weighted sum method, the -SPHRH's potential to omit local optima, and the model's current focus as a "topology router" which does not yet capture all design aspects like minimum bend radius or material properties, leaving these for subsequent design steps. While PSO offers flexibility in objective definition, its stochastic nature yielded inferior and inconsistent solutions in the CHRP context.

2.1.3 Virtual Reality (VR) and CAD-Based Design

Beyond algorithmic optimization methods, several studies have explored immersive and CAD-based approaches to improve harness design through enhanced visualization and human interaction.

Ritchie et al. (2007) [3] presented a methodology for cable harness design, assembly, and installation planning using an immersive VR environment. This approach allows engineering designers to creatively generate and edit cable harness routes in a 3D virtual space by plotting points, bending cables around obstructions, forming bundles, and adding connectors and fasteners. A key aspect of this method is its non-intrusive logging of user interactions (e.g., hand and head movements), which automatically generates downstream manufacturing planning data, such as detailed cable harness assembly and installation plans, without requiring explicit user input or subsequent manual amendments. This real-time monitoring also facilitates the categorization and analysis of design activities, providing insights into how engineers approach design problems and interact with the system.

The study demonstrated substantial productivity gains over traditional Computer-Aided Design (CAD) systems for creative cable routing design activities. The ability to non-intrusively monitor and log user activities provides a unique capability to understand creative design processes without interrupting cognitive thought processes. Besides, the method effectively supports how engineers think during the design process, offering a valuable tool for understanding design behavior and comparing different design platforms. It allows for ergonomic checking of harness access within the virtual environment, which is crucial for installation feasibility. On the other hand, full automation of wire manipulation in 3D space remains a significant challenge and an active area of research. These findings underline the potential of immersive technologies in complementing, rather than replacing, traditional CAD-based harness design workflows.

Furthermore, the literature has shown the significant role of CAD software in modern satellite cable harness routing and development [13; 19]. CAD tools like Zuken E3 series alongside SolidWorks [19], and Siemens NX CAD software [13], are employed to create 3D models of cable layouts. The process typically involves defining electrical interfaces and pin-to-pin connections, transferring this data for 3D routing, and then utilizing design features to control critical aspects such as minimum bend radius, stress relief on connectors, and adherence to EMC restrictions. These systems also support the generation of manufacturing documentation.

The advantages of using CAD for cable routing are numerous and directly contribute to optimizing satellite design. CAD can help in the identification of shortest possible routes to minimize signal loss and optimize overall harness mass, which is crucial given that harness mass can constitute a significant portion of a satellite's dry mass. It facilitates EMI/EMC mitigation by allowing separate bundling and routing of different signal types (e.g., power, digital, analog, RF, pyro signals). Furthermore, CAD permits accessibility simulations with human models to ensure ease of installation and maintenance, helps design harnesses to withstand mechanical loads, and provides a concrete model before physical implementation, aiding in early problem detection and avoiding delays or cost overruns. It also helps to ensure compliance with strict standards like NASA-STD 8739.4A [13; 19], which will be described in the next section.

Thus, CAD tools facilitate visualization, analysis, and management of complex routing, enhancing accuracy and reducing the need for physical prototypes. Despite these benefits, certain limitations of relying solely on CAD for cable routing are implied. While CAD provides a refined model, mock-up models are still developed for hands-on experience to identify gaps and practical challenges related to accessibility and rework that cannot be fully identified without physical interaction [19]. Additionally, the simulation might not always perfectly capture all physical nuances or component variations [13]. For many years, cable harness design has been

a classic design problem since many companies still need to use physical prototypes to create and verify cable routing, despite the availability of comprehensive CAD-based tools [3; 7].

Beyond design, prototypes are being developed to automate various stages of wire harness production, including wire preparation (cutting, labeling, stripping, crimping), pre-routing, routing on mock-ups, and post-routing termination. Although production is not part of this thesis, it remains an important aspect of overall harness optimization. Lei et al., (2021) [18] directly relates to harness routing by focusing on reducing the overall weight of satellite harnesses, a critical concern for high-orbit satellites. It identifies wire length, which is determined by the three-dimensional modeling direction of the satellite harness, as a key factor influencing harness weight. While the main contribution of this study lies in material-based optimization—using lightweight wires and thinner insulation layers—the resulting mass reduction directly depends on the efficiency of the routing design. Therefore, minimizing cable length through optimized three-dimensional routing is fundamental to maximizing the benefits of lightweight materials and improving satellite performance and longevity.

In this thesis, CAD-based design is combined with the A* search algorithm and a logical framework for incorporating design constraints. This integration enables both geometric optimization and adherence to practical design and integration constraints, bridging the gap between algorithmic methods and human-centered design environments.

2.1.4 Standards for Cable Routing in Satellites

Regardless of the design methodology, adherence to established physical practices and guidelines is paramount for ensuring the integrity, performance, and longevity of the harness in the space environment. Several sources highlight the importance of adhering to specific standards and guidelines for cable harness design and verification in space applications [14; 20; 16; 13]. Two foundational standardization frameworks in space projects are the European Cooperation for Space Standardization (ECSS) and the National Aeronautics and Space Administration (NASA) standards. Both establish rigorous requirements that govern the mechanical, electrical, and thermal design of cable harnesses to ensure reliability in the space environment.

2.1.4.1 ECSS Standards

ECSS provides comprehensive standards for spacecraft harness design, with an emphasis on mechanical integrity and reliability. ECSS-E-ST-20C Rev.2 [25] outlines top-level rules for harnesses. Notably, it forbids using any harness bundle as a structural support: “No piece of harness shall be used to transfer mechanical loads”. Harness restraining systems (clamps, ties, etc.) on the spacecraft structure must be designed so they do not induce stress into connectors. This prevents damage at the connector interface.

Regarding **mechanical routing and support**, ECSS standards mandate careful routing paths that avoid sharp edges, excessive bending, or free-hanging loops. Harnesses should be secured along the spacecraft structure with adequate supports to withstand launch vibrations and shock. The ECSS-Q-ST-20-30C standard [26] integrates industry guidelines (IPC/WHMA-A-620E-S) for securing and installing harnesses. According to these rules, cable bundles must be anchored at regular intervals using ties or clamps, and any interface with structure should include cushioning. For example, when a tie-wrap (cable tie) is used to attach a harness to a chassis or panel, ECSS explicitly requires a protective layer under the tie to prevent insulation damage. Clamps and lacing must be tight enough to hold the bundle but not so tight as to cut or deform the wires – compression of wire insulation is controlled to avoid degradation of electrical performance. In fact, for materials prone to “cold flow” (creep) like PTFE Teflon, ECSS calls for extra protective measures (such as sleeves or grommets under clamps) to prevent long-term

compressive damage to the insulation. By adhering to these practices, the harness remains mechanically secure without the wires themselves bearing loads or getting pinched.

Both ECSS and its adopted industry standards impose **minimum bend radii** to avoid stressing cables. Although the exact requirements vary by cable type, ECSS-Q-ST-70-61C [27] provides pre-assembly of wires rules, stating that the bending radius of the wire shall be as a minimum 2 times the diameter of the wire except for polyimide insulated wire for which the minimum bending radius is 10 times the diameter [27]. In cases where manufacturer recommendations specify more restrictive bending limits, ECSS standards do not override these values, and the highest (most conservative) bending radius requirement shall be applied. For instance, high-voltage or thermally sensitive cables are often handled per manufacturer recommendations, ensuring no sharp kinks that could compromise insulation [28]. These radius constraints preserve the mechanical and electrical integrity of the wiring over the satellite's life by preventing micro-cracks in conductors or insulation that could be caused by acute bends.

ECSS standards acknowledge that cable routing is not purely a mechanical issue, but also affects electrical performance and EMC. ECSS-E-ST-20-07C Rev.2 [29] requires harness design to segregate wiring by circuit type and sensitivity. It specifies a categorisation of harness and separate routings for wires of different categories, meaning that, for example, critical signal lines (like sensor or communication lines) should be physically separated from power lines or switching lines that could introduce noise [29]. This reduces EMI by distance and isolation. Additionally, ECSS-E-ST-20C [25] emphasizes that power lines should be twisted with their return conductor (if the structure is not used as return), specifically to minimize loop area and inductance. Twisted pairs and careful shielding are therefore standard practice.

Shielding is another key aspect. ECSS-Q-ST-20-30C [26] devotes sections to harness electrical shielding. Braided metal shields over cable bundles must provide continuous coverage and be properly terminated (e.g. pigtail to ground or 360° termination at connectors) to ensure effectiveness. The ECSS adoption of IPC-A-620 includes strict criteria for shield quality – any broken shielding strand is treated as a defect in a high-reliability harness. In other words, the braid or foil shield should remain intact and tight around the wires. ECSS standards also demand that shield terminations at connectors are robust (often using approved band clamps or solder sleeves) so that shielding integrity is maintained through connector interfaces. Proper shielding and routing (keeping sensitive lines separate from noisy lines) are part of the mechanical layout rules since they often dictate where and how cables run on the spacecraft structure (for instance, routing coax cables along a grounded surface, or grouping all "dirty" power lines away from "clean" signal bundles).

Besides, there are **structural and thermal considerations**. The ECSS harness standards acknowledge the extreme environment of space – including vacuum, extreme temperatures, and radiation – and thus incorporate related constraints into mechanical design. All materials used in wiring (conductors, insulation, tapes, ties, etc.) must be space-qualified for outgassing and thermal tolerance. For example, preferred insulation materials are PTFE, polyimide (Kapton), or ETFE, which can withstand wide temperature swings and do not emit vapors in vacuum [30]. ECSS process standards note that their normal requirements apply to assemblies used from –45 °C to +85 °C, and beyond this range special design measures are needed [31]. That means a standard harness is tested to operate in typical satellite thermal conditions, but missions expecting more extreme cold or heat must implement additional protections (such as heater tapes on harnesses, or use of specialized high-temperature wire insulations). Mechanically, differential thermal expansion between harnesses and structure is considered – slack in the routing can accommodate contraction/expansion to avoid stress. ECSS also requires derating of wires (i.e. using wires gauge sufficiently large for the current) to limit self-heating, and it coordinates with thermal control standards so that harnesses are routed away from high-heat

sources or are shielded/insulated if they must cross warm areas [30]. In summary, ECSS active standards ensure that satellite harness routing is robust mechanically (securely clamped, no sharp bends, no undue loads) while also respecting electrical separation, EMI shielding, and the thermal environment for long-term reliability.

2.1.4.2 NASA Standards

NASA has analogous stringent standards for spacecraft wiring harness design, primarily embodied in NASA-STD-8739.4A [32] ("Crimping, Interconnecting Cables, Harnesses, and Wiring", released June 30 2016) [33]. This NASA technical standard (which remains active, with updates through Change 4) provides detailed requirements to ensure that harnesses are mechanically secure and electrically reliable in the space environment.

Similar to ECSS, NASA's standard emphasizes that harnesses must be robustly supported and strain-relieved. All wire bundles should be routed along structure and clamped or tied down such that connectors and terminations are not bearing the cable load or tension. Similarly, it mandates that wiring must be supported to withstand shock and vibration, rather than left loose; harness cleats, ties, or clamps are installed at suitable intervals to prevent wires from flexing or whipping during launch. In short, NASA does not allow free-floating cables: the harness should effectively become part of the structure through proper restraint, without transmitting excessive stress to electrical connection points.

A key aspect of NASA-STD-8739.4A [32] is the enforcement of **minimum bend radii** and gentle routing practices for all cables. The standard provides quantitative guidelines presented in the Table 1 for different cable types. For instance, an overall harness bundle containing large-gauge wires or coaxial cables should not be bent tighter than about 6 × the bundle diameter. Harnesses made of only smaller wires with American Wire Gauge (AWG) #10 or smaller are slightly more flexible but still have a minimum bend radius of roughly 3 × the diameter. Cables with Kapton® (polyimide) insulation are given special care, which it is required at least 10 × the diameter as the minimum radius. This reflects the fact that polyimide, while high-temperature, is less elastic and more prone to insulation cracking if bent sharply.

Table 1: Bend Radii for Completed Interconnecting Cable or Harness. OD[1]: Outside Diameter. Source: NASA-STD-8739.4A (2016) [32]

Parameter	Minimum Bend Radius
Individual coaxial cable	6 × OD
Polyimide (Kapton) insulated	10 × OD
Overall harness (with coaxial cable or AWG size 8 or larger)	6 × OD
Overall harness (with AWG size 10 or smaller without coaxial cable)	3 × OD
Overall harness (with polyimide insulated wires included)	10 × OD

Additionally, NASA's standard specifies that wires must be routed clear of sharp or rough edges on the spacecraft. If a harness must pass over an edge or through a cutout, the edge should be rounded, grommeted, or padded – "excessive flexing or pressure on the harness over sharp or rough edges shall be prevented" and protective sleeving or cushioning added wherever abrasion could occur. These measures ensure the mechanical routing path itself does not introduce weak points in the wiring.

NASA's harness requirements also address electrical and EMI considerations as an integral part of the mechanical design. EMI refers to unwanted electromagnetic emissions or interference, while EMC denotes the overall ability of the system to function properly in its electromagnetic environment. The standard calls for twisted pair construction for wiring where applicable – it even gives a guideline that the twist pitch (lay length) of wire pairs in a harness should be on the order of 8–16 times the harness diameter. Twisting conductors (especially power and return or differential signal pairs) minimizes loop area, which reduces electromagnetic emission and susceptibility.

Furthermore, NASA-STD-8739.4A [32] also requires proper **shielding** practices for cables: it specifies that shield braids or foils be appropriately selected and applied, and that their terminations are done with approved methods. For example, the use of solder sleeves or crimped ferrules is recommended for terminating individual cable shields to connectors or ground. The standard notes that the shielding materials (braid coverage, mesh size, etc.) should be chosen to suit the application's needs – implying that high-frequency or sensitive circuits may need high-density braids or double shielding. Electrically, NASA projects also impose EMI control requirements (often via MIL-STD-461 for emissions/susceptibility), which translate to harness design rules like separating high-power lines from low-level signal wiring and bonding cable shields at designated grounding points.

In terms of **thermal and material considerations**, NASA's approach is to use only materials that are tested for space conditions: for instance, insulations and sleeving must meet NASA outgassing limits and temperature ratings (this is typically verified through references like NASA's outgassing database and materials standards). NASA-STD-8739.4A [32] itself focuses on workmanship, but it ties into other NASA standards for materials and processes – for example, potting compounds and connector accessories are specified to be suitable for the expected thermal range and vacuum of space.

Finally, both ECSS and NASA require thorough verification testing of harnesses (electrical continuity tests, insulation resistance, hi-pot tests, and sometimes vibe/thermal cycle tests for qualification) to ensure that the routing and installation meet all mechanical, electrical, and environmental requirements. In summary, Both ECSS and NASA frameworks converge on the same principles: prioritizing secure mechanical routing (adequate clamping, bend radius control, no connector stress) while also addressing the electrical integrity (EMC shielding, proper grounding) and environmental resilience throughout a spacecraft's life-cycle.

2.2 Eu:CROPIS Mission Overview

Eu:CROPIS was a small life-science satellite developed by the German Aerospace Center (DLR). It was the first mission in DLR's Compact Satellite program and was designed to investigate sustainable plant cultivation in space as a step towards regenerative life support systems [34]. This mission also serves as the case study for this thesis, providing a real spacecraft configuration to apply the proposed automated cable routing methodology.

The primary payload (Figure 6) consists of two greenhouse experiment modules where tomato seeds are germinated in a closed-loop system. A biological trickle-filter reactor (with symbiotic bacteria) processes synthetic urine into fertilizer, while an algae (Euglena gracilis) culture provides oxygen and moderates ammonia, creating a self-contained ecosystem for plant growth [35]. By rotating around its longitudinal axis, the satellite generates artificial gravity – 0.16g (Moon gravity) and 0.38g (Mars gravity) depending on spin rate – to study plant growth under those conditions [36].

In addition to the main greenhouse payload, the satellite carried several secondary technology experiments [36]:

Figure 4: The Eu:CROPIS spacecraft with deployed solar panels – the satellite is a cylindrical container (approximately 1 m diameter x 1 m height, size of the solar panels approximately 1 m x 1 m). In orbit, the top side faces the sun [35].

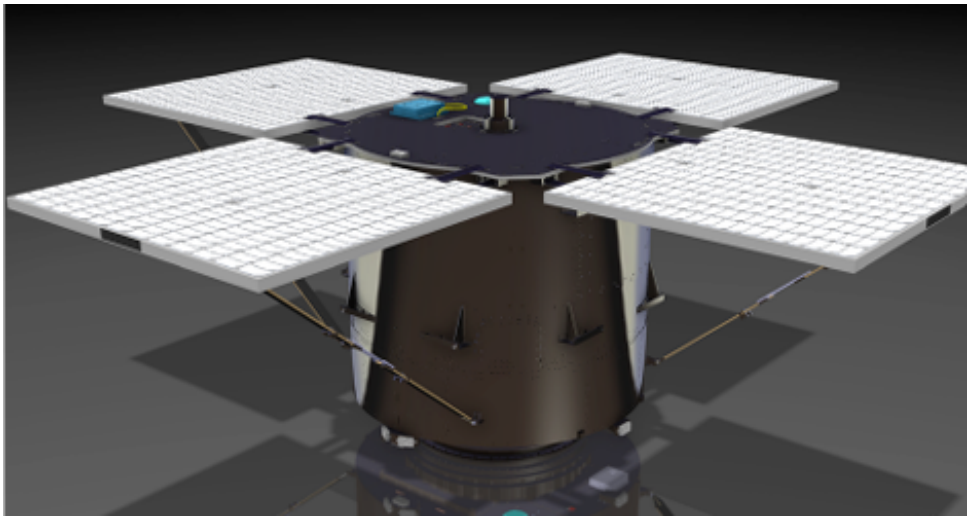


Figure 5: Schematics of the Eu:CROPIS spacecraft – with an internal view on the left and an external view on the right (solar panels not included). The lower section illustrates the satellite's bus compartment, while the upper section houses the primary payload within a pressurized vessel. The satellite bus also includes the outer hull and the top plate, which accommodate additional bus components as well as secondary payloads [35].

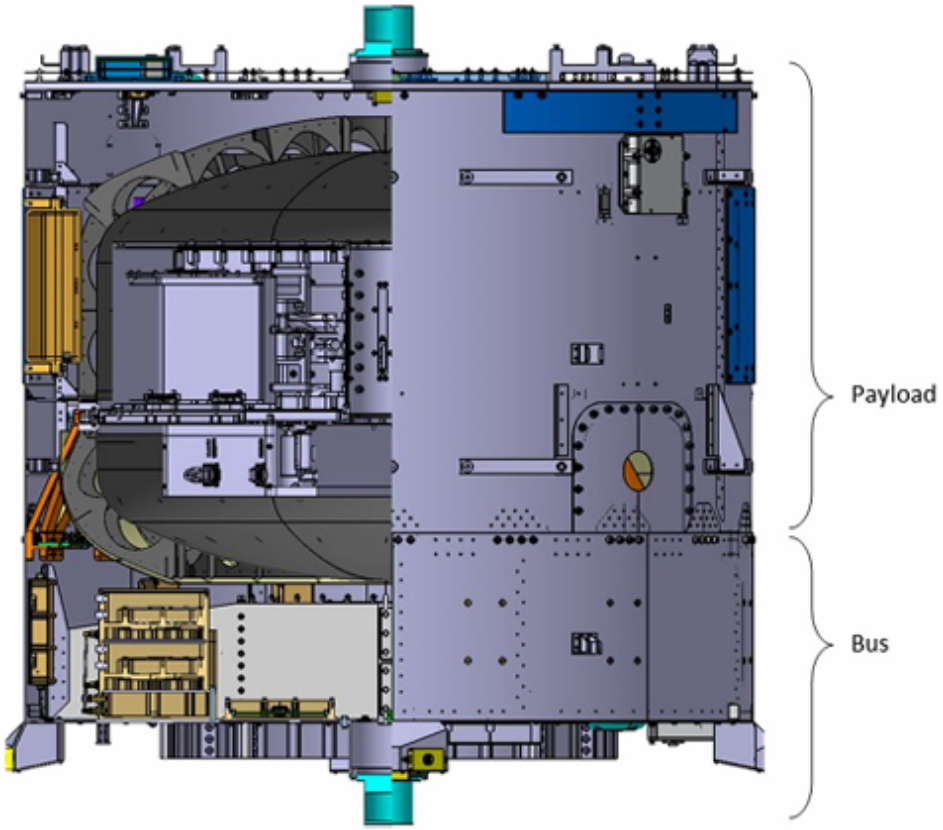
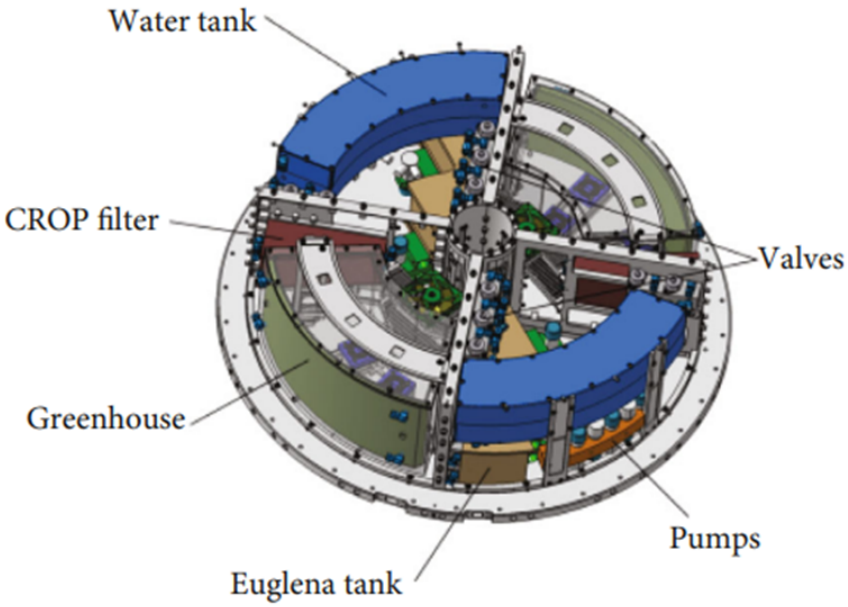
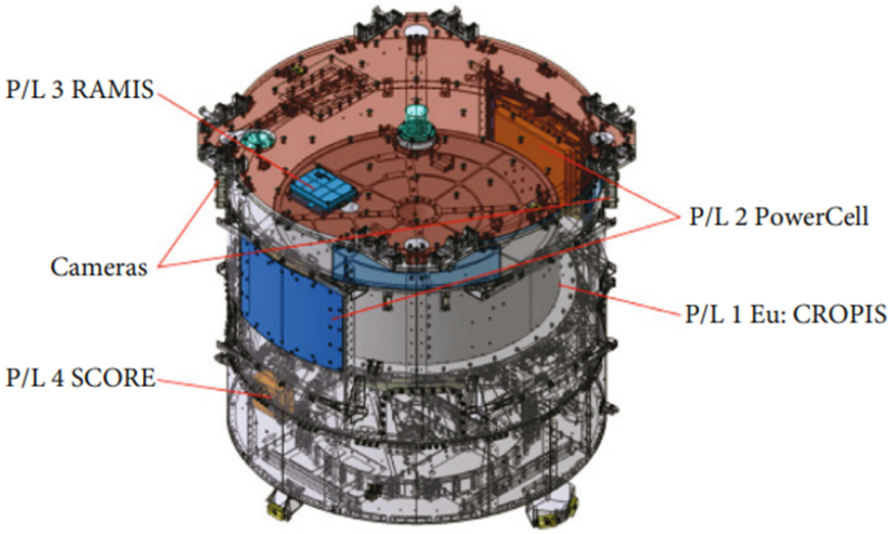


Figure 6: Eu:CROPIS primary payload module [34].



- PowerCells in Space – a secondary bioexperiment from NASA Ames investigating microbial growth and synthetic biology under reduced gravity.
- RAMIS – an onboard instrument (Radiation Measurement in Space) to monitor cosmic radiation inside and outside the spacecraft.
- SCORE – a DLR compact on-board computer demonstrator (Scalable On-board Computing) with additional cameras for imaging tests.

Figure 7: Eu:CROPIS payload distribution [34].



Eu:CROPIS was launched on 3 December 2018 aboard a SpaceX Falcon 9 rocket into a Sun-synchronous orbit at 575 km altitude [36]. The mission was nominally planned for about one year of operation in orbit. [36] After 6 months of lunar gravity, the experiment was planned to be terminated [35]. Unfortunately, the primary scientific payload of the Eu:CROPIS mission

could not be activated due to a software malfunction. Nevertheless, the Eu:CROPIS satellite represented the German Aerospace Center's (DLR) first use of a compact satellite platform optimized for low mass and cost-effective deployment, incorporating novel lightweight structural designs [37].

In January 2019, a routine software update led to communication issues with the two internal greenhouse modules. Despite comprehensive error analysis and extensive ground-based testing, repeated attempts to re-establish communication—including module reboots and system-level interventions—were unsuccessful. Although the central biological experiment could not be initiated, the mission provided valuable engineering insights and operational experience for future DLR compact satellite designs. The lessons learned are expected to inform the development of subsequent missions and alternative flight opportunities are being evaluated to conduct the intended biological experiment [37].

The mission formally concluded on 31 December 2019 due to software issues that prevented activation of the core payload.

The incomplete scientific phase of the Eu:CROPIS mission, caused by software activation issues, underscores the importance of robust system validation and design integration. These principles also extend to cable harness development.

2.2.1 Harness Design and Integration

Inside the Eu:CROPIS satellite, an intricate cable harness connects all subsystems, instruments, and solar panels, distributing power and data throughout the 1 meter-class satellite. Harness design was a non-trivial aspect of the satellite's engineering due to the tight volume and the need to maintain balance for spin stabilization [34].

The harness routing process was largely performed manually in a CAD environment and relied strongly on the experience and intuition of the engineers. Instead of a fully automated routing approach, general routing paths were defined by placing a sequence of points within the structural model of the satellite. At selected locations, additional orientation constraints were specified to force spline segments in preferred directions, for example to avoid sensitive components, comply with bending constraints, or follow structural features. This approach allowed engineers to incrementally shape the harness geometry while accounting for mechanical interfaces, assembly constraints, and subsystem accessibility.

During assembly and integration, a structural model of the satellite was used to optimize the harness routing. The final flight harness installation was thoroughly verified during integration to confirm that all wires and connectors were properly accommodated and secured according to the design [34].

Notably, the mass and placement of the harness had measurable effects on Eu:CROPIS's center of gravity and moment of inertia. As a spin-stabilized spacecraft, precise mass distribution is critical to minimize wobble and achieve the desired artificial gravity gradient. Early mass property tests revealed a discrepancy between the predicted and measured balance of the satellite, partly because the initial CAD models did not fully capture the mass of harness cables and fasteners [34]. In fact, the measured angle between the actual spin axis (centroidal principal axis) and the intended body axis was off by about 6.8° before corrections [34]. Engineers resolved this by refining the spacecraft design models with very detailed representations of the harness and by adding small trim masses in strategic locations to rebalance the spacecraft [34]. After updating the model and adjusting the trim weights, subsequent measurements confirmed that Eu:CROPIS met all strict mass property requirements for stable rotation [34], underscoring the importance of automating harness mass calculations in CAD models for spin-stabilized satellites.

Overall, the Eu:CROPIS mission demonstrated a novel combination of biological life-support research and small-satellite engineering. The successful integration of a complex harness in a compact, spinning spacecraft – while supporting sensitive bioexperiments – highlights the mission's engineering complexity and the solutions developed by the DLR team. The lessons learned in harness routing, mass trimming, and system integration from Eu:CROPIS provide valuable experience for future missions with tight payload constraints and demanding environmental requirements [34].

3 Objectives

The motivation for this thesis lies in addressing the limitations of traditional cable routing methodologies and exploring how automation can streamline design processes, reduce costs, and improve the reliability of satellite systems. Furthermore, incorporating considerations such as bundling and bending radius ensures compliance with space standards and enhances overall system performance.

Based on this motivation, the main objectives of this thesis are defined as follows:

- Developing an efficient methodology for cable routing automation in satellite systems using Synera software [8] environment.
- Incorporating relevant constraints into the design process to ensure analysis with compliance with satellite reliability standards.
- Applying the created workflow to the Eu:CROPIS spacecraft.

In this thesis, the methodology integrates a CAD environment with an A* path finding algorithm and a spring–particle system logic framework. This combination enables both geometric optimization and adherence to practical design requirements, bridging the algorithmic approaches and engineering-driven design processes.

3.1 Scope and Research Questions

The scope of this thesis focuses on the automation of harness routing applicable to satellites. While manufacturing and electrical testing aspects are not directly addressed, their requirements are reflected through geometric and physical design constraints. The main research question guiding this work is:

How can an automated workflow combining graph-based routing and physics-based reconstruction improve the efficiency and quality of cable harness design in satellites?

Secondary questions include:

- How can design constraints such as bending radius, mounting regions, and cable segregation be logically represented and integrated within a CAD environment?
- To what extent can automated routing reduce design time while maintaining compliance with space system standards?

To ensure a controlled and representative evaluation of the proposed workflow, the Eu:CROPIS spacecraft is used as the reference mission. However, only a selected subset of the bus harness items is considered rather than the full spacecraft harness. This focused approach maintains the geometric and physical complexity necessary to validate the workflow's logic—covering path-finding, bundling, reconstruction, and verification—while keeping the computational scope traceable for detailed analysis.

The following chapter presents the methodological workflow developed to achieve these objectives and explains its implementation for the selected Eu:CROPIS harness.

4 Methodology

4.1 Problem description

The methodology developed in this thesis aims to model and solve the satellite harness routing problem through a structured, automated process integrated within a CAD environment. The problem is defined within a three-dimensional DS, R^3, containing both free routing areas and obstacles that must be avoided. Within this space, specified start and end points must be connected by cables, and a route refers to the path a cable follows through this environment.

The formulation presented in this thesis is one among several possible approaches. It focuses on investigating whether satisfactory routing solutions can be achieved through an automated process that combines a graph-based method using the A* algorithm (see Section 4.2.1: Path Finding) with a curve reconstruction technique inspired by physics-based method (see Section 4.2.2: Raphos Physics). Additionally, this work aims to understand the complexity of the problem under this approach and to develop a logical framework for routing multiple cables and bundles simultaneously. The automated solution should also facilitate the analysis of routing results with post-processing and evaluation of the resulting harness geometry.

Given the time constraints typically associated with such projects, a further objective is to enable the solution of industrial-scale problem instances significantly faster than current manual industry practices.

Although various factors influence harness design, this thesis focuses on the following key criteria:

- cable length,
- mounting space regions,
- collision free routes,
- bundling of cables,
- bending radius during reconstruction,
- distance between different cable categories, and
- mounting intervals, which are evaluated during the post-processing phase to verify adequate fixation spacing along the routed harness.

Finally, to evaluate the proposed workflow for cable routing automation, a use-case based on a completed mission is applied and analyzed.

4.1.1 Routing Constraints and Considerations

This section defines the required input data, design constraints, and routing considerations that govern the development and evaluation of the automated workflow.

- Input Data

  The data structure must capture all relevant input parameters, including connectors and geometric references:

  - **Harness specifications:** Each harness item data should include connector CAD identification name, cable diameter, cable category (as defined in Table 2) and density. Multi-connection harness items are grouped into bundles, whereas two-point connections are treated as individual cables.

Table 2: Category of cables. Source: Strowik, C. (2022) [38]. Note[1]: The given cable types are recommendations. Note[2]: The table may be extended if an additional I/F type is required. Note[3]: t_r is rise time and t_f is fall time.

Interface Type	Category	Cable
Primary DC Power	1	Twisted Pair (TP)
Secondary Power		TP
Motor Interface		Twisted Shielded Pair (TSP)
Commands: 5V to 28V, $\geq$ 1A	2	TP
DigitalCircuits: 5V to 28V, $t_r/t_f \geq$1µs		TSP
DigitalCircuits: 1V to10V, $t_r/t_f \leq$ 1µs		TSP
Commands: 5V to 28V, $\leq$ 1A	3	TP
DCCircuits $\leq$ 10V, $\leq$ 5A		TSP
DigitalCircuits: 5V to 28V, $t_r/t_f \leq$ 1µs		TSP
DigitalCircuits: 1V to 10V, $t_r/t_f \geq$1µs		TSP
Valve Position I/F	4	TSP
Analogue Signals		TSP
Bi-Levels Signals		TSP
Serial Data Lines	5	TSP (75Ω)
Bus Extension		TSP
User Time Clock Lines		TSP
Modulation Signal Lines		TSP
High Speed Data		TSP
IEEE802.3		TSP(120Ω)
High-Level Digital Circuits, with max. voltages $\geq$ 10V, $t_r/t_f \leq$ 1µs		TSP
Explosive Device Circuits	6	TSP

  - **Connector geometries:** The CAD geometries define start and end points for each connection, indicating precise connection locations in the satellite where the cables are initiated and terminated. Besides, the orientations for both ends of the cable can be extracted from the geometry.
  - **Satellite and component geometries:** The CAD model of the spacecraft and the components for defining structural and spatial boundaries.
  - **No-go areas:** Optional geometric zones (e.g., high-temperature regions) excluded from routing.

- Cable Routing Considerations

  The cable routing process must comply with a set of mechanical, electrical, and environmental design rules to ensure that the resulting harness meets the reliability and safety requirements of space systems. These rules are derived primarily from the ECSS and NASA standards, which define best practices for cable separation, bending, fixation, and mechanical support. In the developed workflow, these requirements are translated into algorithmic constraints and logical checks that guide or validate the automated routing process. The following list summarizes the key constraints considered in this thesis and how they are incorporated into the proposed methodology.

  - **Collision avoidance:** All cable paths must remain free from intersections with structural geometries.
  - **Length minimization:** Wires and cables should be routed the shortest way (Strowik,

C. 2022).

- **Bundling:** Logical grouping of cables into bundles. This helps integrating the hardness and prevents tangling. Branching is not be considered in this work.
- **Mounting regions:** Cable routing should follow paths located close to the spacecraft walls or structural panels, where fixation and mechanical support can be achieved. In the developed workflow, the Design Space (DS) is constructed to guide cable routing along or near these surfaces, ensuring that each automatically generated path remains in regions suitable for mounting and later mechanical fixation. Additionally, the DS directs cables from their connectors straight toward the nearest wall, providing a natural fixation point and ensuring that the harness is properly strain-relieved in accordance with ECSS-Q-ST-20-30C [26] and NASA-STD-8739.4A [32], which emphasize that cable bundles must be mechanically supported and strain-relieved, particularly near connector interfaces.
- **Segregation by category:** As mentioned in the Chpater 2.1.4, ECSS-E-ST-20-07C Rev.2 [29] requires segregation of wiring by circuit type. For instance, separating power lines or high-current switching circuits from low-level analog or digital signal cables. This spatial separation reduces coupling effects and noise propagation between different signal types. Similarly, NASA-STD-8739.4A [32] guidelines require separating high-power lines from low-level signal.

  In this methodology, cable segregation is implemented by assigning each cable a category (as defined in Table 2). The routing logic enforces a minimum separation distance between cables of different categories for different harness items. Each cable within a harness item is assigned its category individually, allowing cables of the same or different categories to coexist within a single harness item if required by the system design. The separation is set to 20 mm (2 cm) by default, but adjustable by the user depending on design requirements. This flexibility reflects practical engineering scenarios in which an engineer may intentionally group cables of different categories within the same harness item for reasons such as installation constraints, interface grouping, or system architecture, while still ensuring that category-based clearance rules are respected during routing for different harness items.
- **Thermal avoidance:** Routing avoids defined high-temperature zones (“no-go” areas).
- **Electrical constraints and prioritization:** According to ECSS-E-ST-20C [25] and NASA-STD-8739.4A [32], voltage drop is a critical parameter in spacecraft electrical design, as excessive losses can compromise power delivery and equipment performance. Because voltage drop is directly proportional to cable length, minimizing routing distance is particularly important for power and other high-current lines.

  To account for this in the automated process, a *user-defined prioritization* is implemented. Cables or bundles with stricter voltage or length requirements—such as power, high-frequency, or optical data lines—can be assigned higher routing priority. These are processed first during the path-finding phase to ensure the shortest feasible paths within the DS. Subsequent cables are routed according to their remaining priority level or geometric proximity between connection points.

  This prioritization logic enables the algorithm to reflect electrical sensitivity within the routing sequence, ensuring that critical connections meet voltage drop and performance requirements while maintaining overall routing efficiency.
- **Bending radius:** Compliance with minimum allowable bending radius requirements

is essential to prevent mechanical damage and to maintain the long-term electrical integrity of cables. Both ECSS-Q-ST-70-61C [27] and NASA-STD-8739.4A [32] specify minimum bending radii based on cable type, insulation material, and diameter.

In the proposed workflow, the minimum bending radius is not enforced during the voxel-based path-finding stage but is instead addressed during the reconstruction phase. After routing, the voxel-based paths are reshaped using a physics-based reconstruction implemented with the Raphos Physics solver, where bending behavior is modeled through dedicated geometric and physical goals.

The minimum allowable bending radius for each cable or bundle is defined according to Table 1. Rather than directly enforcing this minimum value, the reconstruction logic uses a user-defined factor applied to the minimum bending radius to define an *achievable target radius*. In this context, the standard-defined minimum bending radius serves as a reference value, while the applied factor promotes smoother curvature transitions.

Any local violations for the minimum bend radius that remain after reconstruction are identified and reported during post-processing for further analysis and user verification.

- Certain aspects, such as harness fixation, are evaluated only during the post-processing phase. The workflow verifies whether mounting space in certain intervals are available.
  - **Fixation:** Fixation is verified in the post-processing phase of the workflow to ensure that the automatically routed harness can be properly secured according to aerospace integration standards. Although ECSS and NASA standards do not define specific fixation intervals, both emphasize the need for adequate mechanical support to prevent motion, vibration-induced wear, or connector stress during launch and operation [26; 32].

    During verification, the workflow evaluates whether sufficient mounting regions exist along the routed path for potential fixation points. Default fixation intervals are provided, as summarized in Table 3, but these values can be updated by the user according to specific design or structural integration requirements. Shorter spacing is typically applied for thinner harness bundles to prevent excessive flexing, while thicker and stiffer bundles can tolerate slightly greater distances between supports.

Table 3: Distance between fixations.

Harness diameter	Interval Distance
< 5mm	75mm
> 5mm	100mm

These default values are used as a baseline for post-processing verification to confirm that the generated routing can be feasibly secured to the spacecraft structure without compromising mechanical stability.

- Outputs
  - 1. Optimized Cable Routing:
    - 3D model showing the proposed cable paths.
    - Calculated center of mass.
  - 2. Report:

* Information of harness itens and connectors.
* The checked constraints are described: minimum bend radius, intersection with the satellite structure or sub-systems, distance among different cable categories and mounting interval.

4.2 Mathematical Tools

This section introduces the mathematical and algorithmic foundations that support the proposed cable routing workflow. Two core tools are used to generate and refine cable paths: a path-finding algorithm based on A* search, and a physics-based reconstruction approach using Raphos Physics. Within the Synera environment, these tools are available as nodes and are combined through a low-code, node-based workflow defined in this thesis.

In Synera, a node is the fundamental unit of computation and logic, representing a single, well-defined operation within a data-flow system. Each node receives inputs, performs an operation (such as creating geometry or evaluating a condition), and produces outputs that can be passed to other nodes. Workflows are driven by data dependencies rather than explicit execution order: whenever an input changes, all dependent nodes are automatically updated. This makes nodes conceptually comparable to functions with explicit inputs and outputs, where overall behavior emerges from their interconnections rather than from a linear script.

The principle of node-based logic is illustrated in Figure 8, which shows a simple mathematical workflow. Two input parameters are provided to an *Addition* node, which computes their sum and forwards the result to an output node. Although trivial, this example demonstrates the core interaction model in Synera: data flows through connected nodes, and each node performs a clearly defined transformation.

Figure 9 illustrates a geometric example, where numerical parameters (radius and length) are connected to a *Cylinder* node. The node generates a three-dimensional solid body based on these inputs, which is then passed downstream for visualization or further processing. This example demonstrates how parametric geometry can be constructed by combining basic nodes, with geometry updating automatically when input parameters change.

Therefore, building logic or workflows in Synera consists of structuring a network of nodes that progressively transform data from inputs to outputs, where the sequence and interaction of operations are explicitly defined by the workflow designer. A typical workflow begins with parameter or data-generation nodes, continues through nodes that modify, evaluate, or filter this data using logical and mathematical operations, and ends with nodes that produce geometry or evaluation results. More complex behavior is achieved by decomposing problems into small, single-responsibility steps and chaining them together, allowing conditional logic, data filtering, and geometric rules to be expressed visually.

This modular and reactive structure provides the foundation upon which the path-finding and physics-based reconstruction logic used in this thesis is constructed, enabling the combination of multiple algorithmic tools into a coherent and adaptable routing workflow without the need for traditional scripting.

Figure 8: Workflow example in Synera: Addition calculation

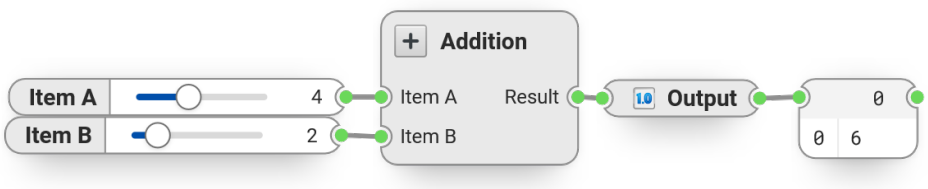
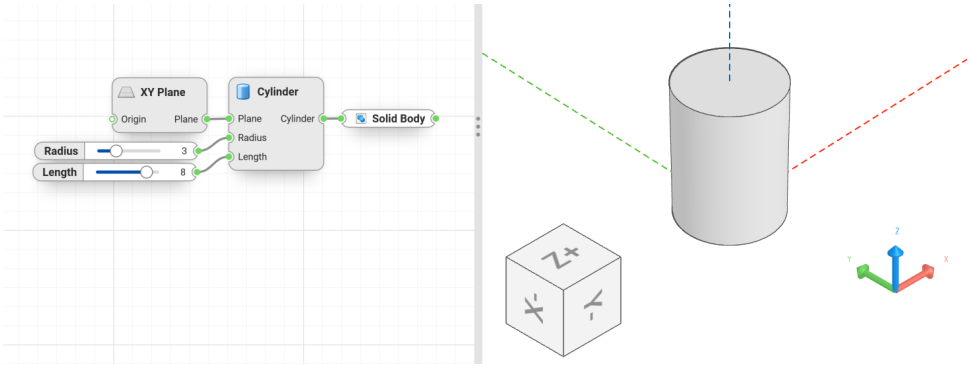


Figure 9: Workflow example in Synera: Cylinder creation generates a visualization of the geometry on the right side.



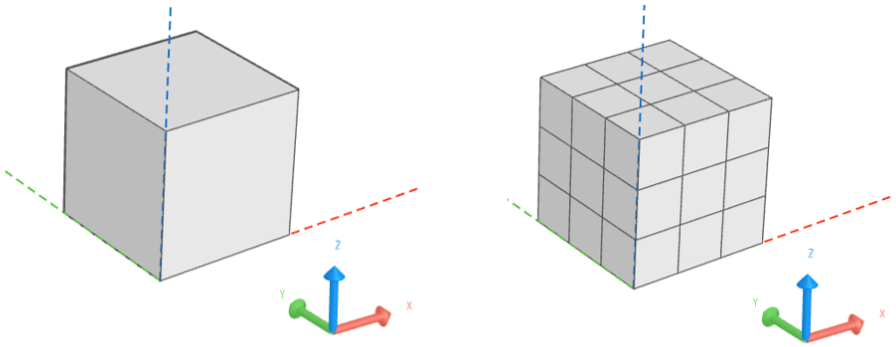
4.2.1 Path finding

The "Path Finding" node is a tool in Synera software [8] that is used to compute the shortest collision-free paths between point pairs within a 3D Design Space, by voxelizing the space and performing a stepwise search algorithm on the voxels [39]. This tool is based on the graph approach using the A* (A-star) search algorithm (both described in the Chapter 2.1.1).

The presented tool is useful for the generation of cable and pipe routings or cooling channels. It is done by analyzing the DS input and creating a voxel space with a specified resolution within the boundaries of the available space for the path [39].

Voxels (volumetric pixels) are small cubic units that discretize 3D space into a uniform grid. Each voxel represents a tiny cube of the design volume, analogous to how a pixel represents a small area in 2D (Figure 10). By filling the 3D design space with voxels, it is obtained a structured lattice of sample points. This provides a topological representation of space that simplifies spatial analysis [40; 41]. In other words, instead of dealing with complex curved CAD geometry directly, the environment is converted into a 3D grid of uniformly spaced cubes. Each cube, i.e. a voxel, can hold data (e.g. whether that region is free space or occupied by an object) and can serve as a graph node for routing algorithms.

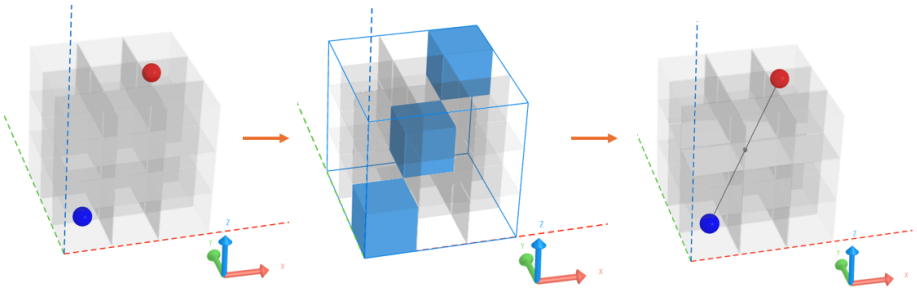
Figure 10: Geometry converted into a voxel 3×3×3 grid: (a) Initial solid body on the left side; (b) Voxelized geometry on the right side.



By voxelizing the CAD model's DS, we create a grid-based search space where A* search algorithm can operate efficiently. The uniform structure of a voxel grid makes neighbor relationships regular in every direction, which is convenient for graph-based path search [40]. Thus, voxels essentially turn the continuous 3D geometry into a set of discrete nodes that algorithms can traverse and the result is a polyline following the central path of the voxelization.

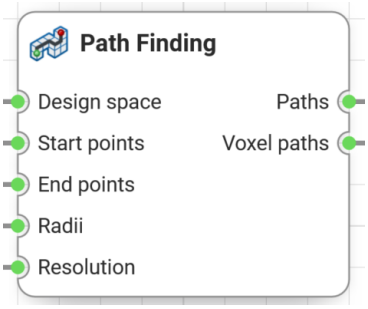
Figure 11 illustrates the stepwise process: the definition of start and end points, voxel-based graph construction, and the resulting shortest path identified by the algorithm.

Figure 11: Path Finding for the 3×3×3 voxelized geometry from Figure 10: (a) Start and end points are defined as the blue (near the origin) and red spheres; (b) A* search algorithm finds a traverse path within the discrete nodes, showing the solved path by the voxels; and (c) the final polyline is generated.



The inputs and outputs for the Path Finding tool in Synera is [39]:

Figure 12: Path Finding Node: Tool in Synera.



- Inputs:
  - DS (Spatial object, Item): the bodies can have any shape and not only solid bodies are accepted, but also the voxel spaces (Voxel Body).
  - Start points (Point 3D, List): The starting points must lay within the defined DS.
  - End points (Point 3D, List): The end points must lay within the defined DS.
  - Radii (Floating point number, List): The radii of the paths.
  - Resolution (Floating point number, Item): The voxel resolution used by the algorithm.
- Outputs:
  - Paths (Polyline, List): The computed paths.
  - Voxel paths (Voxel body, List): The computed voxel paths.

Moreover, the voxel size (resolution) is an important parameter. Smaller voxels capture finer details and narrow passages at the cost of more nodes and higher computation, whereas larger voxels reduce computation but may oversimplify the space [40].

Although this tool provides a practical method for route initialization, it does not consider key

harness-specific constraints such as bundling, bending radius, or fixation requirements. Therefore, in this thesis, the A*-based pathfinder is used as a foundation upon which additional logic is built to guide real-world cable harness design in satellites.

4.2.2 Raphos Physics

Raphos Physics is an essential tool used in the methodology to reconstruct cable paths with realistic physical behavior after an initial route is determined—particularly when path-finding alone cannot account for all design constraints such as minimum bending radii or interaction with complex geometries. The tool enables a more refined reconstruction of the cable by simulating how it would behave under real-world physical influences.

Integrated as an add-in within the Synera software environment, Raphos Physics provides a spring-particle simulation framework capable of handling form-finding, large deformation analysis, dynamic relaxation, impact simulation, and mesh optimization [42; 43]. Its core is a spring-particle solver that enables real-time simulation of mechanical systems undergoing significant deformations [42]. The solver is based on a state-of-the-art projection method with specific improvements that enhance stability and accuracy [11].

The underlying principle of Raphos Physics is grounded in a physical modeling approach where a system is discretized into a set of particles connected by springs. These particles interact with one another through forces, which may be functions of position, time, velocity, or other quantities [43]. For instance, a steel beam subjected to a coaxial load can be modeled as a spring with stiffness $k = \frac{EA}{L}$, where E is the modulus of elasticity, A the cross-sectional area, and L the length. To include bending behavior, the beam can be subdivided into smaller segments with rotational stiffness assigned between them. Such modeling flexibility allows Raphos Physics to approximate a wide range of mechanical phenomena in a simplified but effective manner [43].

A simulation in Raphos Physics consists of particles and a set of goals or constraints they must fulfill [44; 43]. Each goal has a relative weight, which for spring-like goals corresponds to their stiffness. Users can also introduce external forces such as point loads or gravity, and fine-tune simulation parameters including time step and damping to achieve the desired fidelity and performance [44].

The solver is inspired by the optimization approach described by Bouaziz et al. (2012) [11], which offers a unified framework for handling geometric constraints using iterative projection methods. This theoretical foundation enables Raphos Physics to balance precision and computational efficiency, making it well-suited for use in complex cable routing problems where physical accuracy and adaptability are required.

In practical use, the solver behavior can be tuned through parameters such as the number of iterations, convergence tolerances, and a damping factor. The damping factor is used to progressively reduce particle velocities at each iteration according to

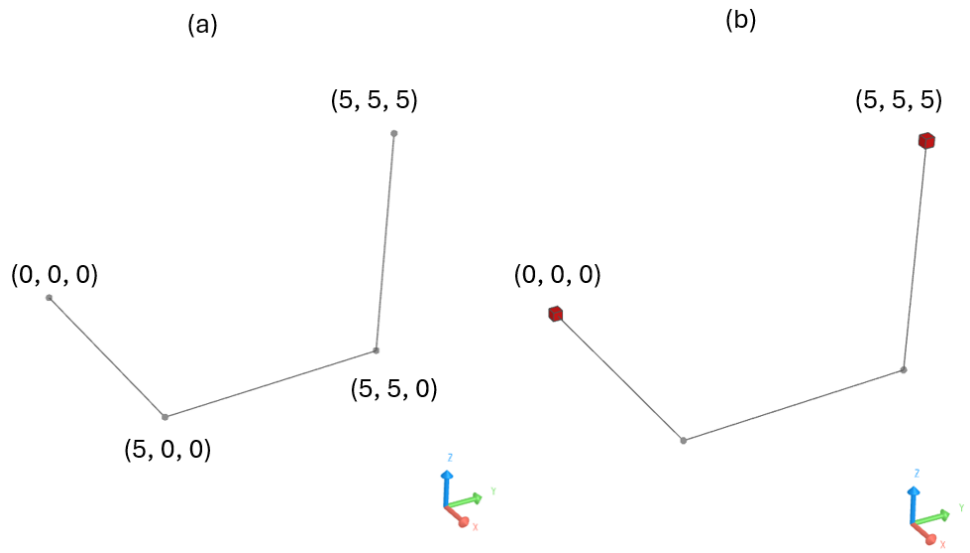
$$\mathbf{v}_{new} = (1 - d)\, \mathbf{v}_{prev}, \tag{2}$$

where d denotes the damping coefficient. This damping stabilizes the solution and prevents oscillatory behavior, especially in scenarios with strong competing constraints.

In order to illustrate the application of Raphos Physics, a simplified case is presented. Figure 13 shows four polylines defined by their spatial coordinates and anchored at both ends. These anchors act as fixed points, constraining the curve ends during the form-finding process.

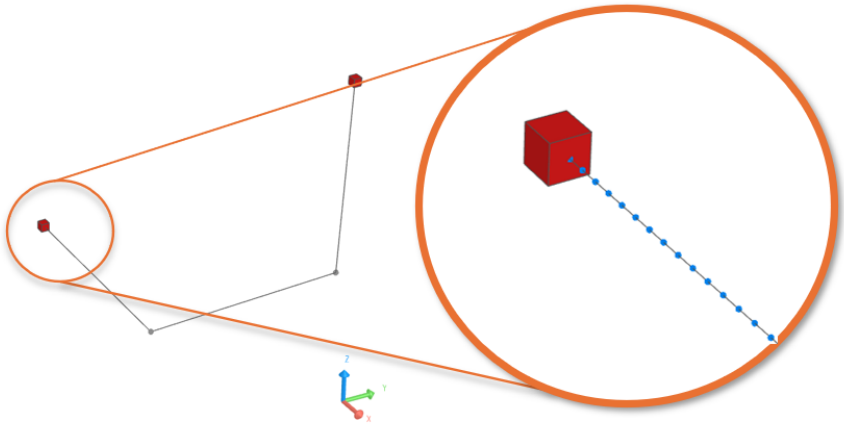
Once the geometry is defined, the polyline is discretized into particles connected by springs,

Figure 13: (a) Initial four polylines defined by points; (b) Start and end anchors are set to fix the curve ends.



as shown in Figure 14. This discrete model enables the application of Raphos Physics, which simulates physical forces and constraints to iteratively adjust the geometry. In this setup, the spring system accounts for the desired bending stiffness by imposing angular goals between segments.

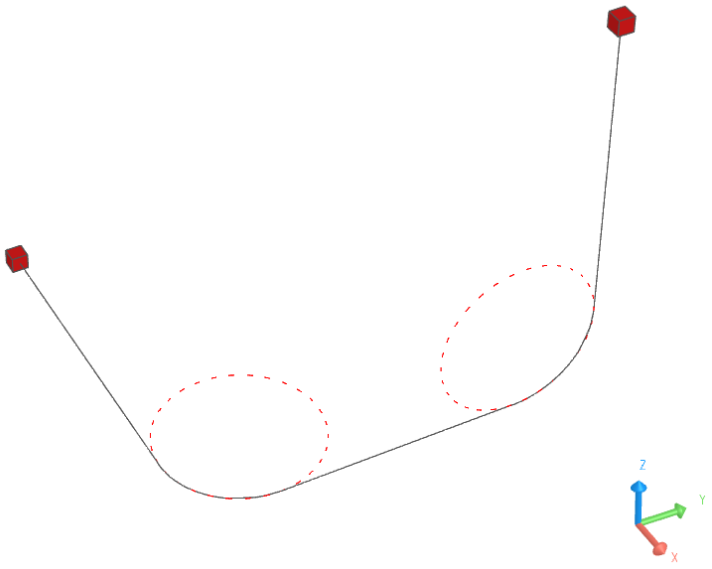
Figure 14: Zoom on the start anchor, showing the discretization into particles and springs.



After the simulation reaches equilibrium, the resulting polyline reflects a geometry that respects bending radii constraints. As shown in Figure 15, the previously sharp corners in the initial polyline are now replaced by smooth transitions that follow a curvature compliant with physical behavior, which is critical in real cable routing applications.

This process showcases how Raphos Physics can transform a basic polyline into a physically plausible cable path by considering mechanical constraints, which would otherwise be overlooked in purely geometric approaches.

Figure 15: Final deformed polyline after the solver enforces bending radii. Dashed circles highlight the defined radii transitions.

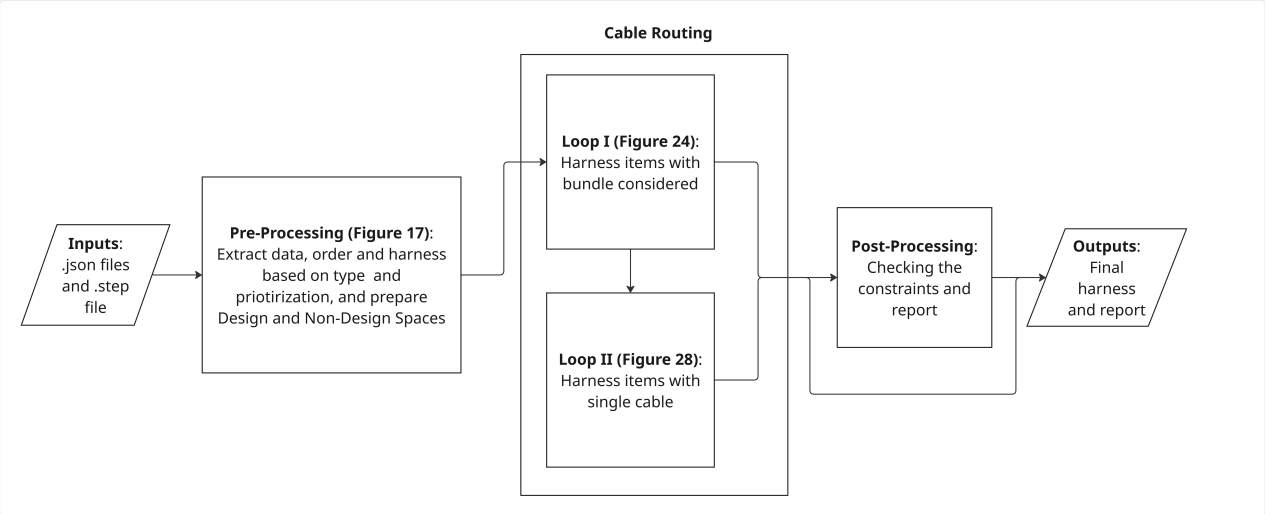


4.3 Model Description (Workflow)

This section presents the low-code, node-based workflow developed and implemented in the Synera software environment for the automated generation of cable harness routing. Building upon the mathematical tools described in Section 4.2, this workflow integrates the A* path-finding and Raphos Physics solvers within a logical sequence that handles data preparation, routing execution, and result verification.

The overall structure of the workflow is summarized in the flowchart presented in Figure 16. It illustrates the logical progression from data input and pre-processing to cable routing, verification, and output generation.

Figure 16: Flowchart – General workflow.



The workflow automates the generation of cable routings for both bundled and single cables.

The inputs are provided in the form of a `.json` file containing the definitions of connectors and harness items (those with only two connectors will be routed as a "single cable", and those with more than two connections will be considered as a bundle) and a `.step` file that contains the geometric assembly with its attributes. Additionally, another `.json` file specifies the names of the geometry elements that compose the satellite's structural and component models.

During the pre-processing phase, the script extracts and organizes the geometric and harness data, classifying each harness item according to its type and the prioritization defined by the user in the `.json` configuration file. The main body of the workflow is divided into two iterative loops:

(1) Loop I — cable routing for harness items where bundling is considered, and

(2) Loop II — routing of individual cables.

The resulting routes are then passed to the post-processing phase, where the solution is checked against defined design constraints and a report is generated.

For all operations, the coordinate system follows the convention where the bottom part of the spacecraft structure lies on the XY-plane with its center positioned at the origin (0, 0, 0), and the Z-axis represents the vertical direction. This orientation ensures consistency in spatial reference throughout all steps of the workflow.

The following subsections describe the four main stages of the developed workflow in detail:

- **Section 4.3.1 Pre-processing**: extraction and preparation of input data,
- **Section 4.3.2 Cable Routing**: automated generation of paths for bundles and single cables,
- **Section 4.3.3 Post-processing**: verification of routing constraints and reporting, and
- **Section 4.3.4 Limitations**: discussion of the model's assumptions and restrictions.

4.3.1 Pre-Processing

The pre-processing stage prepares all necessary geometric and harness data before initiating the automated routing algorithm. It establishes the data structures, imports the 3D geometry, classifies harness items, and defines the broad regions in which cable paths can or cannot be routed. The pre-processing is divided into two main parts:

- Pre-processing I: extraction and preparation of geometrical and harness data.
- Pre-processing II: definition of first broad possible and non-possible regions for the cable routing, i.e., DS and Non-Design Space (NDS).

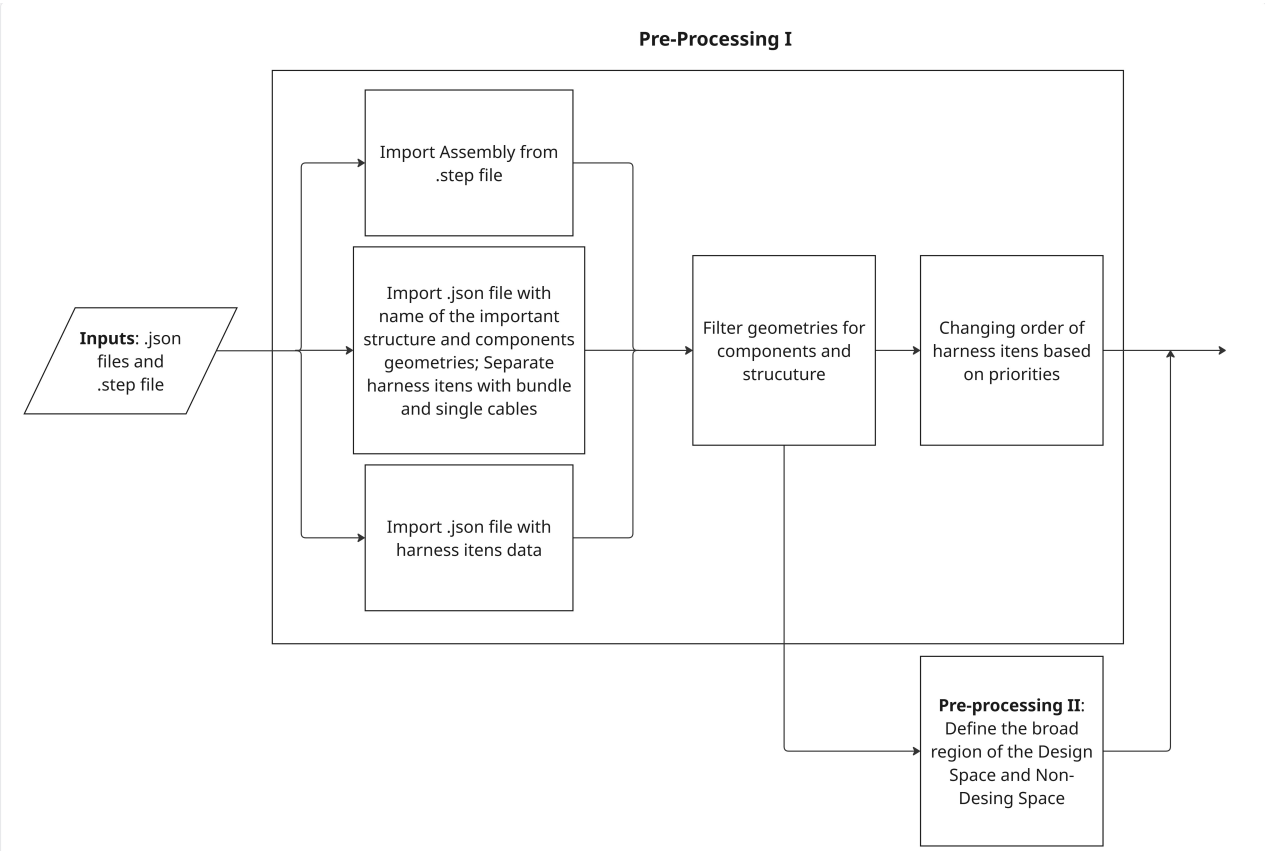
As shown in Figure 17, the pre-processing workflow comprises those two stages.

Pre-processing I

The objective of Pre-processing I is to extract the necessary geometries from the CAD model and organize the input data describing each harness item. This ensures that all harness and connector information is properly linked to the corresponding 3D geometries before routing begins.

In this first phase, Pre-Processing I focuses on importing, filtering, and organizing the input data. The process begins with reading the CAD assembly from the provided `.step` file, from which the necessary geometries are extracted.

Figure 17: Flowchart – Pre-processing workflow.



The workflow also imports a `.json` file that contains the harness data, including names, connector identifiers, connection pairs, diameter, category, density, and user-defined priorities. This information is used to establish a direct correspondence between each cable connector and its associated geometry within the CAD model.

A second `.json` file specifies the attribute names of the relevant geometries that represent the spacecraft structure and subsystem components, ensuring that only these essential elements are retained for routing purposes.

After the geometries are imported according to their attribute names, the harness items are classified according to their configuration. Items with more than two connectors are treated as bundled harnesses, while those with exactly two connectors are defined as single cables. This classification determines how each item will later be routed within the workflow.

Once classified, the harness items are ordered based on their routing priority. If a priority value is explicitly defined by the user in the `.json` file, it is directly adopted. As mentioned in the Section 4.1.1, this functionality is particularly relevant for power lines or other connections where a reduced cable length is required to minimize voltage drop, as recommended by ECSS-E-ST-20C [25]. If no priority is specified, the workflow automatically computes the Euclidean distance between each pair of connectors and orders the cables so that shorter connections are routed first. Bundled harnesses are always routed before single cables because they typically define the main harness topology and occupy larger spatial volumes.

In this way, Pre-Processing I ensures that all harness and geometric data are properly linked, filtered, and organized for the automated routing loops that follow.

Pre-processing II

Once the harness data and geometric models are prepared, Pre-Processing II defines the allowable and restricted regions within the spacecraft volume for cable routing. This stage generates a broad preliminary DS, which represents the region available for routing, and a NDS, which includes the structural and component volumes that must be avoided. The NDS is defined using the geometry of the spacecraft's main structure, subsystem components, and any additional internal obstacles.

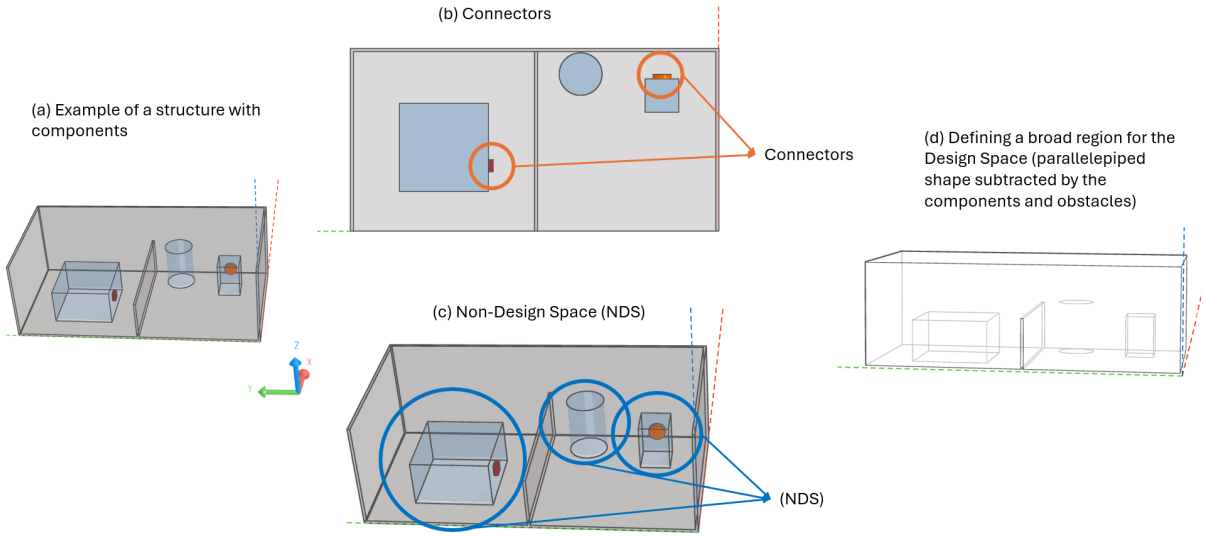
In Figure 18, a simple example of a structure is defined in 18-a. In 18-c, 2 components with connectors and one cylindrical obstacle are shown as example.

The DS is derived as the complementary (negative) volume of the NDS, representing the free space available for routing. The implemented automation supports the definition of DS for typical spacecraft geometries, including cylindrical, rectangular (parallelepiped), and cubic forms. The user selects the appropriate geometry type according to the spacecraft configuration. Figure 18 shows an example of DS for in 18-d.

Wall surfaces are also extracted in this step to enable later alignment of cable segments perpendicular to nearby walls, an important consideration for fixation and mounting analysis.

Optionally, the DS can be subdivided into smaller spatial sections to improve computational efficiency and enable more localized routing. This subdivision step does not yet determine which sections will be used; rather, it creates a structured framework that allows the workflow to later identify and select only the regions relevant to each harness item automatically. This approach significantly reduces computation time by limiting the routing process to the most relevant spatial areas. The division of the DS into sections may need to be adjusted depending on the satellite structure and internal layout to ensure that all relevant routing regions are adequately

Figure 18: Simple example with NDS extraction and Design-Space creation.



captured. For the Eu:CROPIS case study, the spacecraft volume was divided by internal walls, allowing independent processing of each region.

At this stage, however, the DS remains a broad region; in subsequent workflow steps, it will be refined around the spacecraft walls to ensure that cable paths remain close to mounting surfaces and comply with fixation constraints.

4.3.2 Routing Workflow

4.3.2.1 Overview of the Routing Loops

After the data extraction and prioritization steps, the routing process begins with two sequential loops designed to address the different types of harness items. The first loop processes harness items that include bundled cables, while the second handles individual (non-bundled) cables. This separation ensures that bundled connections are routed first according to their defined priorities, followed by the individual cables, which use their own prioritization logic.

This subsection provides an overview of the overall procedure before describing each loop in detail.

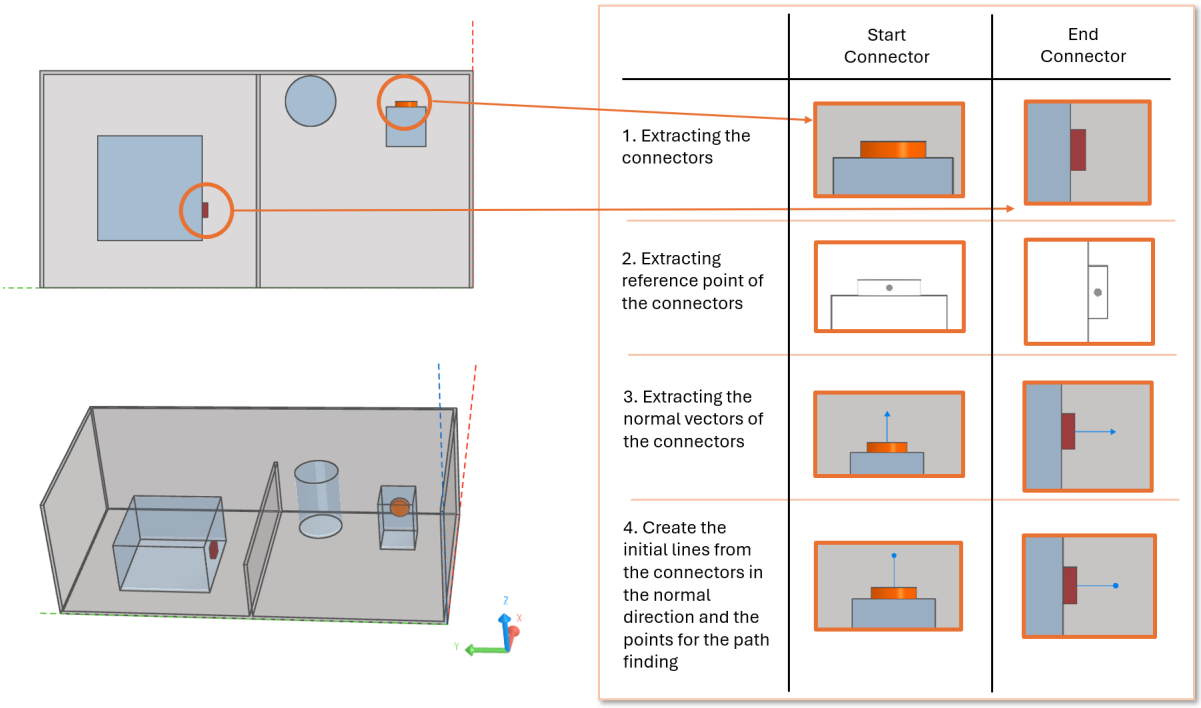
Before the routing loops can be executed, additional pre-processing steps are required to define the precise start and end conditions and to prepare the design environment.

Pre-processing III extracts the geometric information of the connectors, including their centroids and normal vectors. These normal vectors define the intended connection direction, ensuring that each cable exits and enters the connector orthogonally. This prevents the formation of unrealistic bends or immediate curvature near the connector interface, which would not occur in an actual harness. As illustrated in Figure 19, start and end lines are created along the connector normals, and the routing start and end points are displaced to the end of these lines. This guarantees that the generated paths begin and terminate in the correct direction relative to the connectors, resulting in a physically consistent cable geometry.

The normal vectors are derived from the specific geometry of each connector and therefore depend on its shape and interface design. While this extraction approach can be adapted to

different connector geometries, commonly used connector types can be stored in a predefined library by the company or institute, allowing their geometric properties to be reused consistently across different projects.

Figure 19: Simple example with extraction of start and end points and lines for the routing based of connectors and their normal directions.



Pre-processing IV prepares the voxelized DS for routing. To improve computational efficiency, an optional coarse-resolution A* path-finding step can be executed to estimate the general route between the start and end points. This preliminary path is used to identify which spatial sections of the previously defined DS are relevant for each harness item, as described previously at the Section 4.3.1.

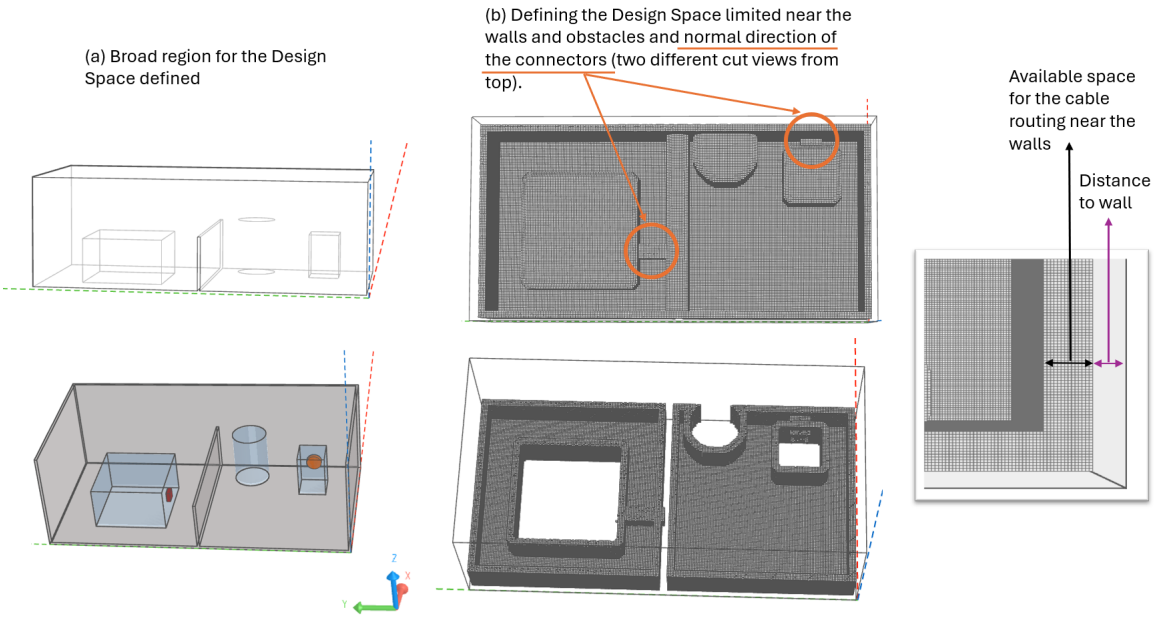
Once the relevant sections are determined, if this optional step is chosen, the voxel space is generated with the corresponding resolution, and the NDS regions are subtracted from it. Figure 20-b shows an example of this process for the geometries in Figure 19. For instance, with the cut-view, it is possible to observe that the regions occupied by components have been excluded from the DS, leaving only the available volume where cable routing can occur.

The DS is further refined by defining a boundary region near the spacecraft walls and components. This ensures that cable routes remain close to mounting surfaces where fixation points are available. The distance from the wall is used for mounting geometries such as clips. The available space for the routing is defined as larger than the cable diameter with an additional clearance margin. Furthermore, local DS regions are generated in the direction of each connector's normal vector to allow direct attachment to the wall. Figure 20-b shows both delimitation near the walls DS in the normal direction of the connectors.

The components can be used for mounting the cables. If a component is not allowed to attach any harness item, this should be added as no-go area. Thus, this regions will be deleted from the DS.

In addition, to prevent undesired interference, regions around unrelated connectors are excluded

Figure 20: Simple example with delimitation of the DS.



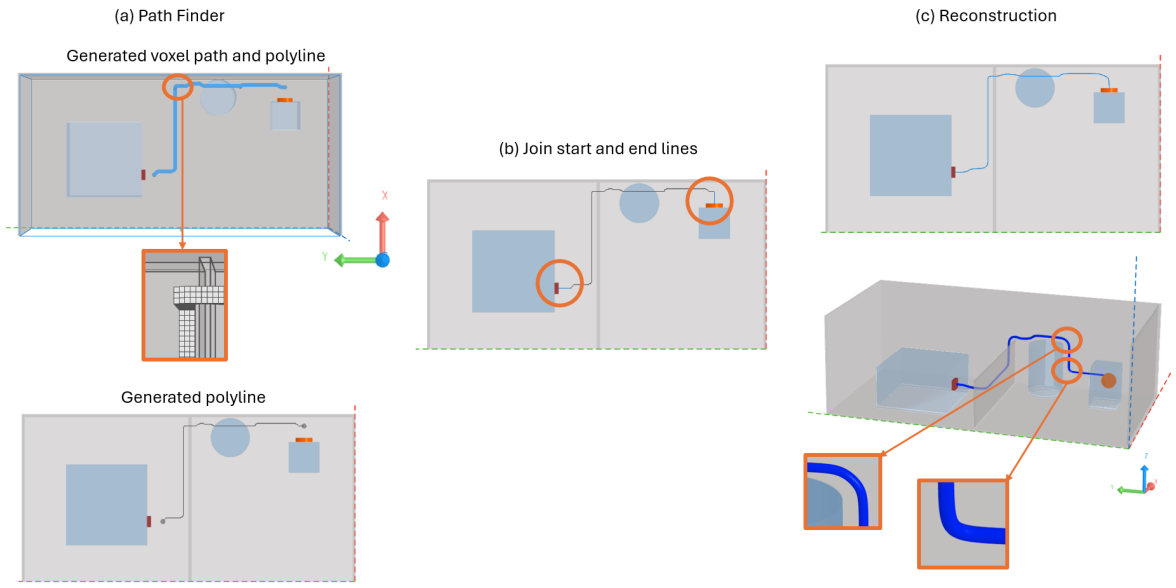
from the DS, ensuring that routes do not pass too close to other interfaces.

Once the environment and parameters are defined, the A* search algorithm computes the initial cable paths within the voxelized DS. The start and end points used in this step correspond to the displaced positions obtained in Pre-processing III (see Figure 19). The voxelized DS used by the pathfinder results from the preparation described previously (Figure 20).

Figure 21 illustrates the routing process: (a) the voxel-based path generated by the A* algorithm, (b) the connection of the path with the initial and end lines aligned with the connector normals, and (c) the reconstruction of the final cable geometry using *Raphos Physics*.

For harness items with bundles, before the reconstruction, the process includes three phases: (1) initial A* path-finding for all connectors, (2) estimation of a representative bundle path based on the shortest individual cable, and (3) a second A* execution constrained by this bundle path. The resulting geometry is then smoothed and physically reconstructed. For single cables, the process includes one A* pass followed by Raphos-based reconstruction. The detailed procedures for both loops are described in the next subsections.

Figure 21: Simple example with cable routing steps with one single cable.



When multiple harness items are routed, the DS must be updated after each iteration to prevent overlap between cables and maintain category segregation. After a harness item is routed, its occupied volume is subtracted from the DS before the next iteration begins. Additionally, cables from different categories must remain separated by a minimum distance, which is defined as the sum of the current cable's radius and the category-dependent clearance distance.

Figure 22 shows the DS and connectors before routing. In Figure 23, the first path (a) is generated and its buffered region (b) is removed from the DS to respect category-based clearances. The following routing step then operates within the updated DS (c), where the voxel field has been locally subtracted to explicitly exclude the buffered volume of the previously routed cable. In this updated DS, all voxels corresponding to the first path region—including its cable radius and the required category-based clearance offset—are removed and treated as NDS. As a result, the subsequent A* search is geometrically constrained to explore only collision-free regions that inherently respect inter-category spacing, without requiring additional penalty terms.

Figure 23-c therefore represents not merely a visual update of the DS, but a topological modification of the routing graph itself, where infeasible regions are eliminated prior to path-finding, ensuring that constraint compliance is embedded directly into the routing logic. Thus, the final configuration 23-d satisfies spacing and safety constraints between cable types. This subtraction logic is crucial for avoiding interference between electrical and data lines and for maintaining compliance with design standards.

Furthermore, to ensure that the final reconstructed geometry maintains these required distances, this separation constraint is also considered during the reconstruction step described later in the next subsection. It is important to add that the workflow allows flexibility in cases where cables of different categories belong to the same bundled harness item. In such situations, as mentioned in the Section 4.1, when the harness engineer determines that EMI is not a concern, these cables can be routed together within the same bundle.

Figure 22: Simple example for two single cables: Connectors and DS.

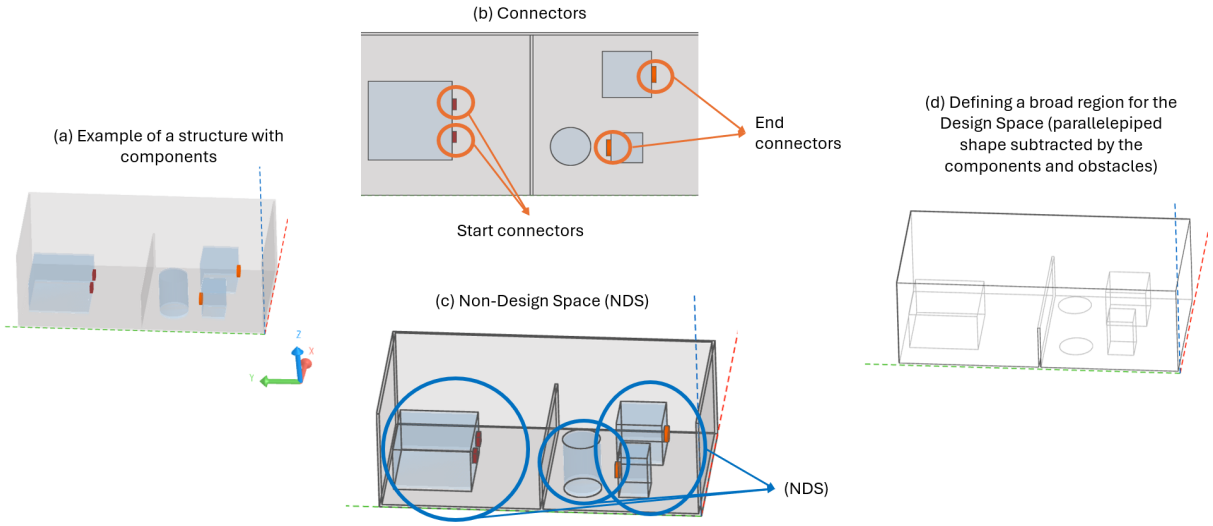
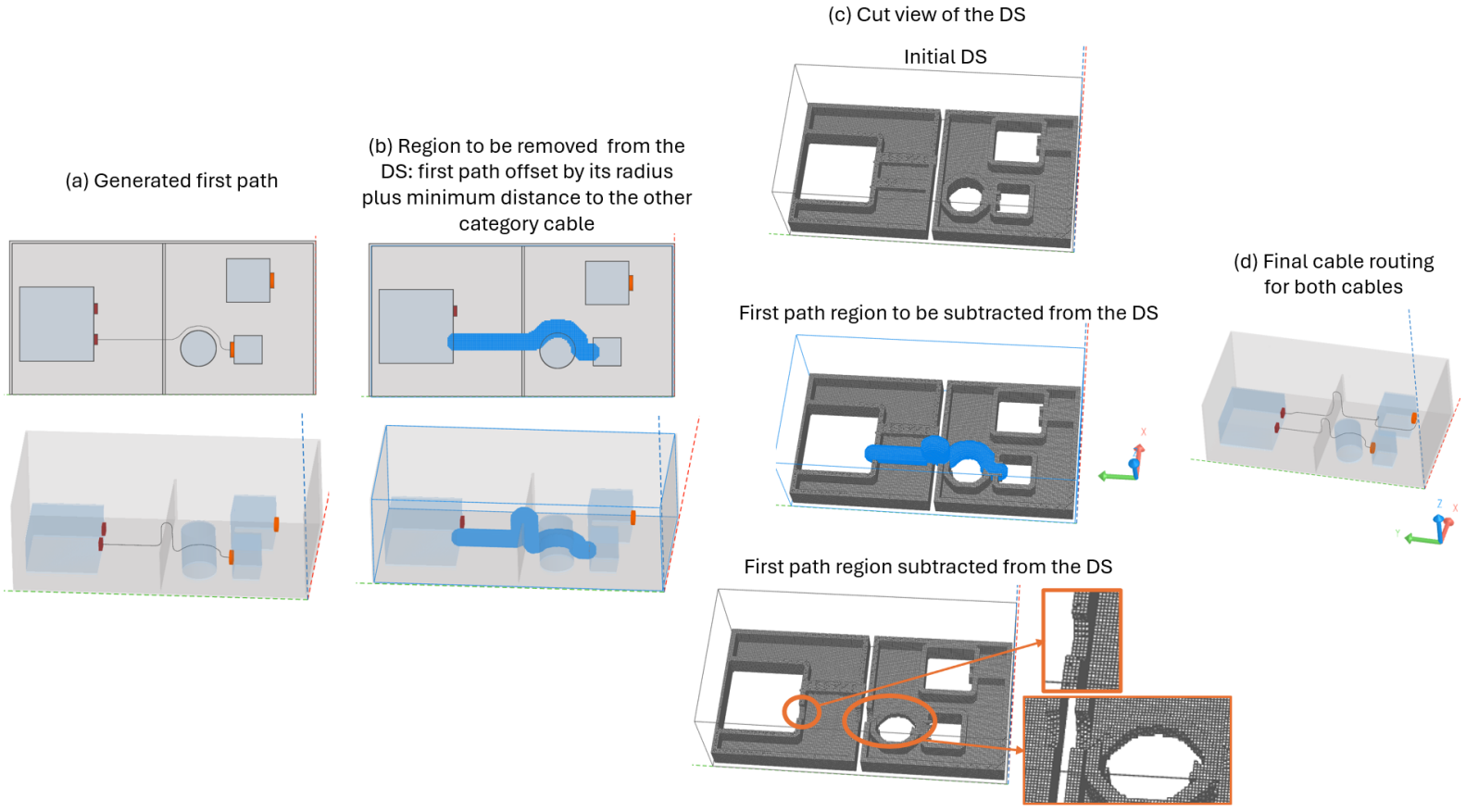


Figure 23: Simple example for two single cables: Subtraction of the region around the first cable.



4.3.2.2 Routing Loop I: Bundled Harness Items

For each iteration corresponding to a harness item that includes a bundle, the workflow begins by extracting the start and end points from the connectors, along with their normal vectors. Using these vectors, start and end lines are generated and slightly offset so that the path begins and ends in the normal direction of the connectors, ensuring that the cables do not curve around the interfaces. As an optional step, a preliminary coarse Path Finding is performed to identify the relevant spatial sections for faster computation. Afterward, the DS is delimited near the walls and objects as previously described, and the main Path Finding process begins. Once the first path baseline is generated, a bundle reference is defined, followed by a second Path Finding step and finally the reconstruction of the cable geometry.

Figure 24 shows the overall flowchart for this sequence. For the next iteration, corresponding to a new bundled harness item, the workflow subtracts the previously reconstructed paths from the DS, applying an offset distance according to the separation requirements between different cable categories.

In order to show an example, the Figure 22 presents the geometries for a harness item with two connections. After the first baseline of the path is generated (Figure 25-a), the bundle reference is created (Figure 25-b), followed by a second Path Finder step, and finally the reconstruction process (Figure 25-c–e).

The first routing step, **Path Finding I**, uses the A* search algorithm (described in Section 4.2.1) in a voxel-based environment to compute an initial baseline path, as shown in Figure 25-a. After this preliminary path is defined, the workflow determines the reference for the bundle. The bundle reference is set by selecting the cable with the shortest path length among all cables in the same harness item.

The external diameter (OD) of the bundle is then estimated based on the sum of the cross-sectional areas of the individual cables, calculated as:

$$OD_{\text{bundle}} = \sqrt{\sum_{i=1}^{n} d_i^2} \tag{3}$$

where d_i represents the external diameter of each cable in the bundle, and n is the number of cables. The process for identifying the bundle's start and end points is outlined in the flowchart in Figure 26.

In order to determine the region of the **bundle**, the workflow calculates the spatial relationship between the shortest cable (bundle reference) and the other cables of the same harness item. Specifically, it evaluates the distance between the points of the reference cable and those of the others, identifying the region where this distance is smaller than a user-defined threshold X. The points within this threshold define the interval that represents the bundle region. Since using simple Euclidean distance does not account for the presence of obstacles or walls within the spacecraft geometry, the workflow performs this analysis using a coarse voxelized DS. The reference cable is discretized into coarse points, and A* search is employed to calculate the actual path distance between these points and the corresponding points of the other cables, taking the environment topology into account.

This allows the algorithm to identify the true spatial proximity and define more accurate start and end points for the bundle region. If no points satisfy this distance X criterion, the closest point from the other cables to the bundle reference is selected instead. However, for cables that run parallel over a certain distance, this fallback approach can lead to suboptimal results.

Figure 24: Flowchart – Loop for the cable routing for the harness items with bundle in consideration.

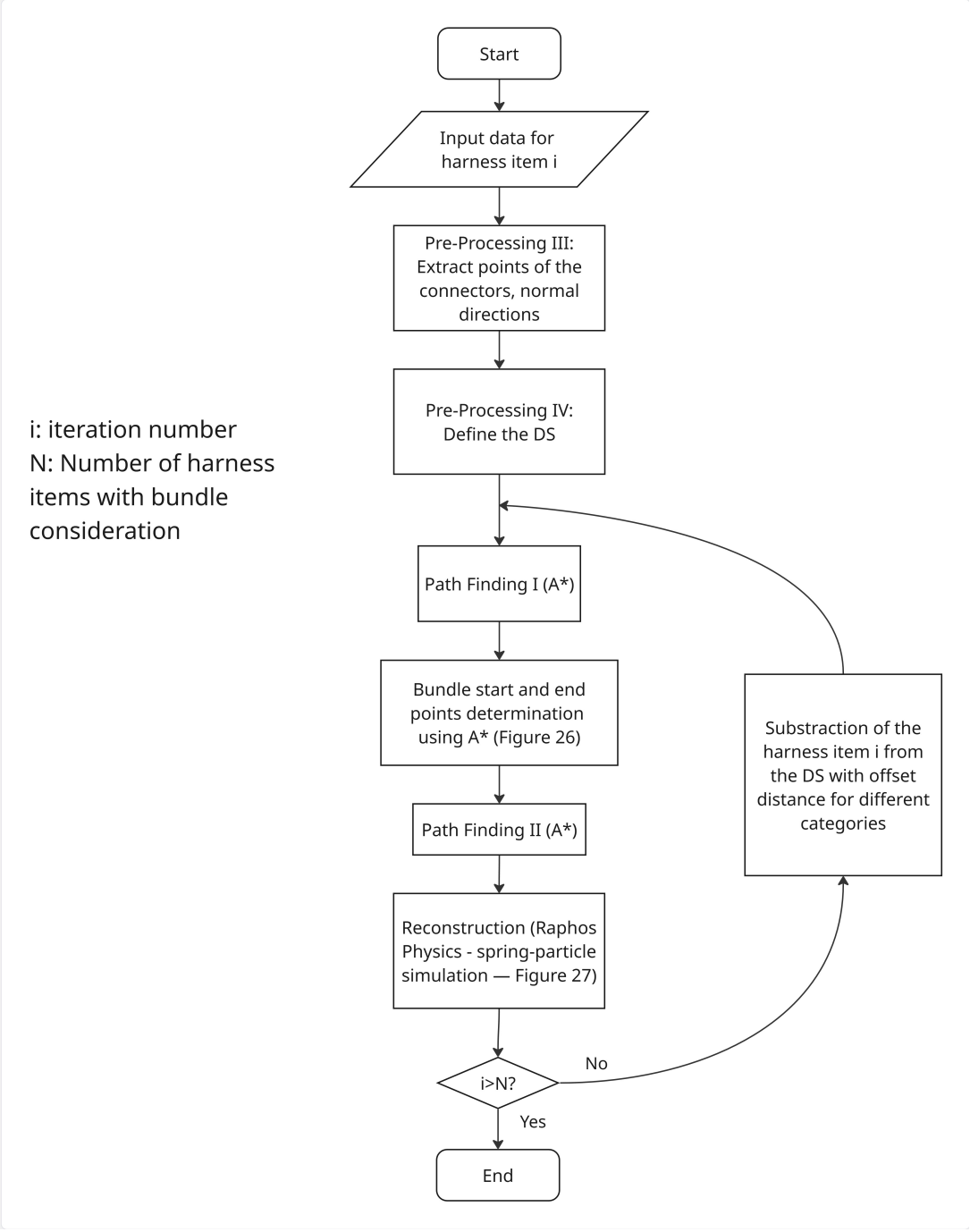
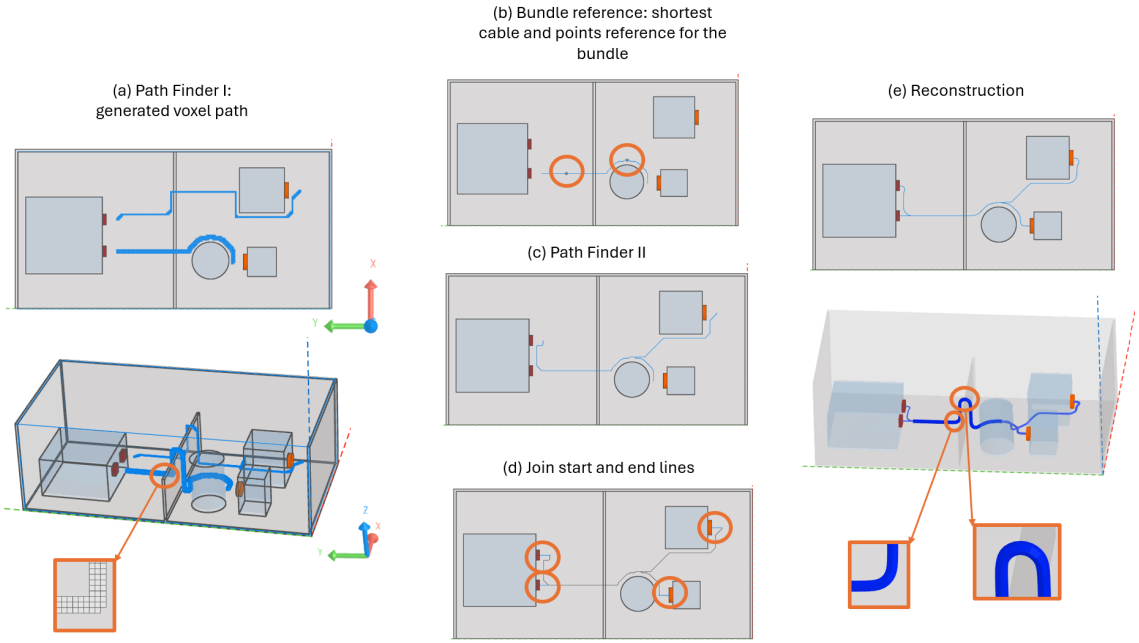


Figure 25: Cable routing example for the harness itens with bundle in consideration.



Because the Euclidean distance remains nearly constant along the parallel segment, the algorithm cannot determine which points should best define the bundle interval. Ideally, the start and end of the parallel region should correspond to the first and last points of proximity, but this approach lacks such contextual evaluation. Nevertheless, if the X criterion is not fulfilled, it already indicates that the routing configuration may not justify defining a bundle for these cables, and a message is generated in the final report to alert the user.

It is also important to note that the regions near the connectors (the beginning and end of the bundle reference) are excluded from this distance evaluation to prevent the algorithm from incorrectly associating short connector stubs as bundled segments. This distance is also user-defined and can be adjusted depending on harness density and routing constraints.

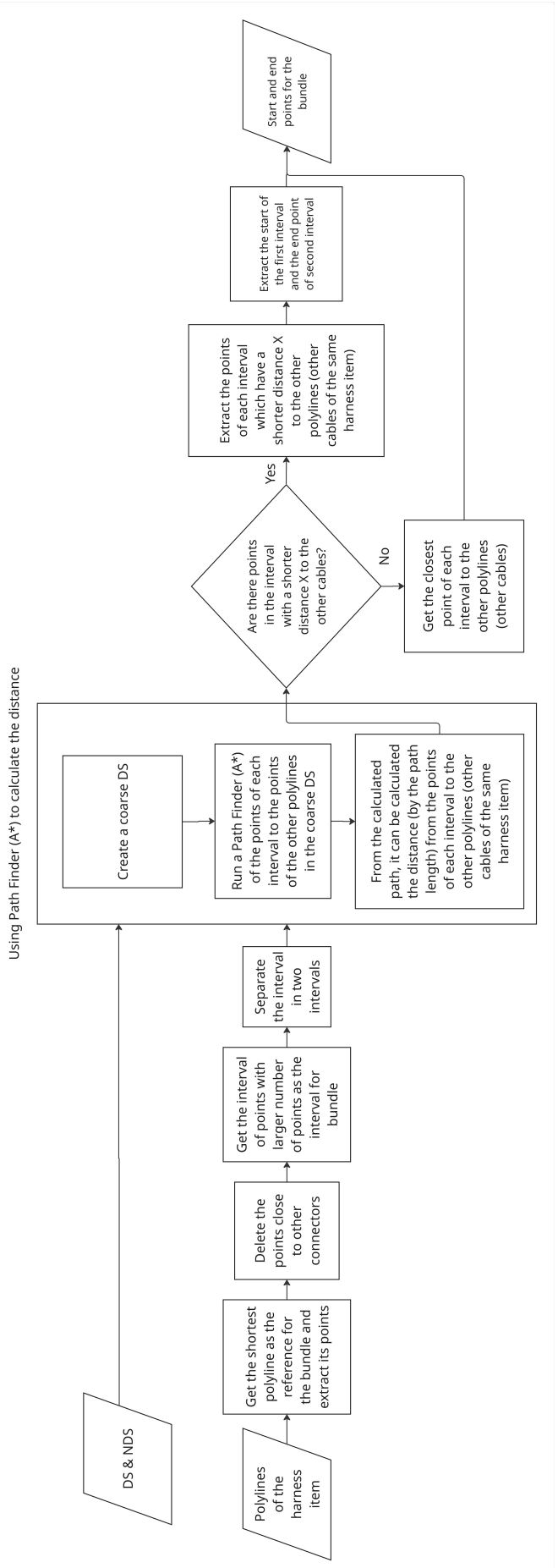
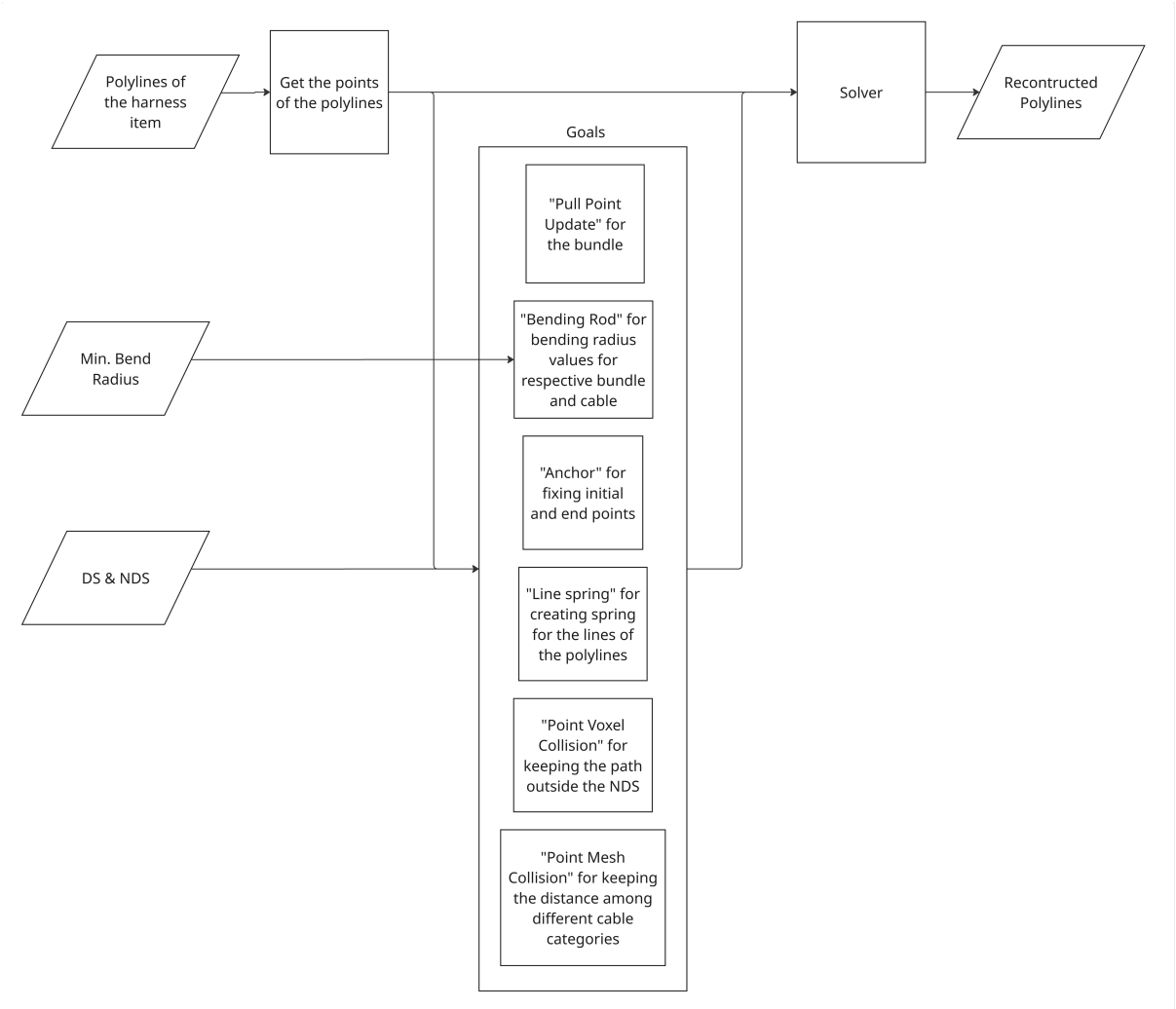


Figure 26: Flowchart – Creation of the Bundle reference points.

Once the bundle interval is defined, **Path Finding II** is performed using A* search again, this time with the bundle reference points as the defined start and end positions (Figure 25-c). In this step, the bundle region is not re-voxelized using the effective bundle diameter. In order to reduce computational effort, the algorithm runs the voxel-based path for the individual cable, without updating the occupied space related to the bundle. After the Path Finding II is finished, the start and end lines are joined with the computed path to form a continuous route (Figure 25-d).

The final step in the process is the **Path Reconstruction**, which employs *Raphos Physics*—a spring-particle simulation framework—to generate the final cable geometry (Figure 25-e). The theoretical basis of this method is described in Section 4.2.2. In this setup, each goal contributes to the spring system with a defined stiffness weight, allowing control over the physical behavior of the cable during relaxation. The reconstruction applies several goals, each addressing a specific geometric or physical constraint, as shown in Figure 27.

Figure 27: Flowchart – Reconstruction.



The *Line Spring* goal creates springs along the polyline segments, establishing the fundamental elastic behavior of the system. The *Pull Point Update* goal maintains the spatial cohesion of the bundle by continuously pulling the corresponding points toward their bundle reference locations during simulation.

The *Bending Rod* goal enforces bending stiffness in accordance with the specified cable or bundle properties. Because the solver seeks to reach the target value rather than maintain a strict minimum, the target bending radius is set by a user-defined factor $f_{BR} > 1$. Thus, the reconstruction target becomes $R_{target} = f_{BR} \times R_{min}$, ensuring compliance with the design constraints. The reference values for the minimum bending radius are taken from Table 1.

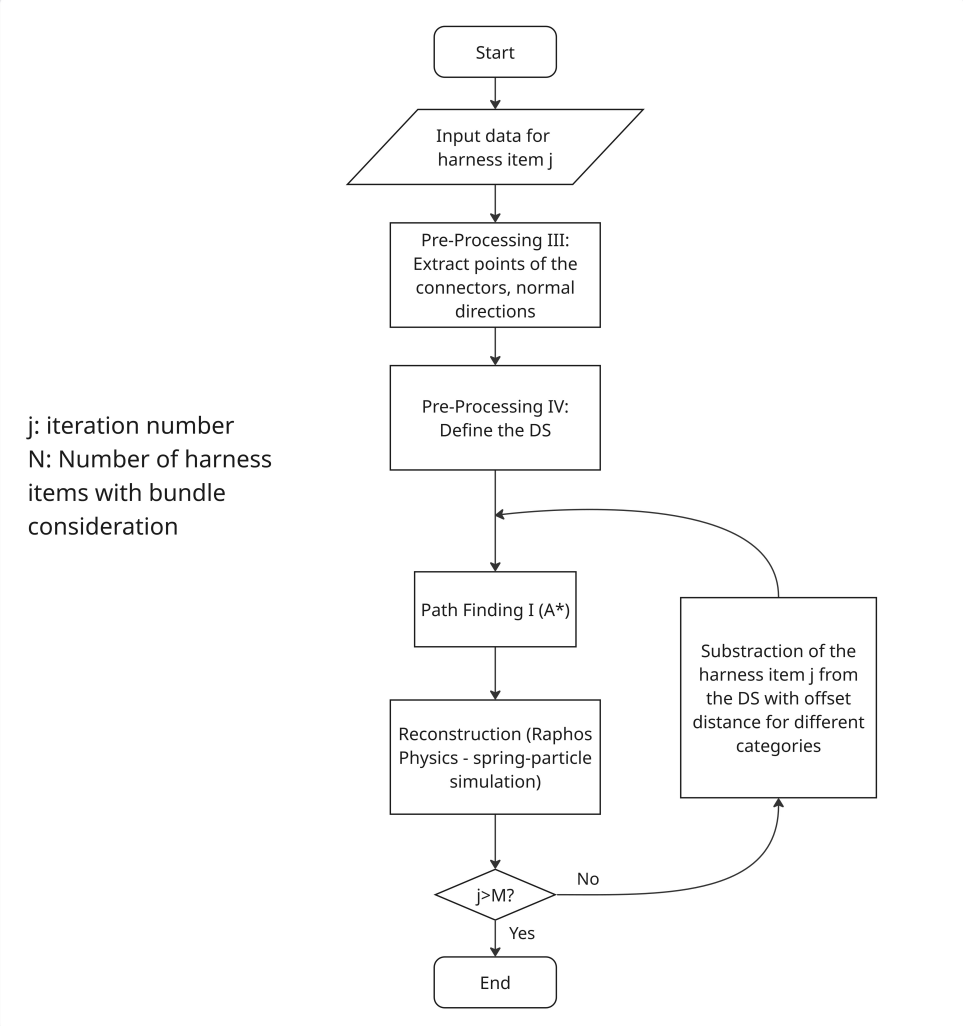
The *Anchor* goal fixes the start and end points, preventing undesired displacement during simulation, while the *Direction* goal aligns the cable segments near the connectors with their normal directions, ensuring smooth transitions at interfaces. To prevent penetration into structural or component geometries, the *Point Voxel Collision* goal keeps the points outside the NDS. Finally, the *Point Mesh Collision* goal ensures proper separation between cables of different harness items—especially those of distinct categories—thus complementing the subtraction process previously described in Figure 23.

4.3.2.3 Routing Loop II: Individual Cables

Compared to the bundled-harness loop, this process is simpler, as it deals with only one cable per harness item. For each iteration corresponding to a single cable, the workflow extracts the start and end points from the connectors, as well as their normal vectors, and generates start and end lines by moving the centroids along the connector normals. Optionally, a coarse path-finding step may be performed to preselect relevant spatial sections for routing, which helps reduce computational cost when the DS is large.

After the delimitation of the DS as previously described, the workflow executes the main path-finding process. The complete loop process for individual cables is illustrated in Figure 28.

Figure 28: Flowchart – Loop for the cable routing for the harness items with separated cables in consideration.



In this loop, the A* search algorithm is applied in a voxel-based space to generate an initial routing baseline, following the same principle as in **Path Finding I** from the bundled-harness loop (see Section 4.2.1). The resulting polyline is then connected to the start and end lines previously defined at the connectors, ensuring the path exits and enters each connector orthogonally. Once the preliminary path is created, the **Reconstruction** phase is carried out using *Raphos Physics* to generate a physically realistic cable geometry. The same goals used in the bundled-harness loop are applied here, except for the *Pull Point Update*, which is not necessary

since there is no bundle to maintain.

The *Bending Rod* goal enforces bending stiffness in accordance with the cable specifications. Because the solver seeks to reach the target radius rather than maintain a strict minimum, the bending radius is defined as a user-controlled factor above the minimum allowable value. Reference values for the minimum bending radius are listed in Table 1, with correction factors depending on the cable type. The *Line Spring* goal ensures continuity along the polyline by modeling each segment as a spring, while the *Anchor* goal fixes the cable's start and end points. The *Direction* goal aligns the cable segments near the connectors with their respective normal vectors. To prevent physical overlap, the *Point Voxel Collision* goal constrains the cable path to remain outside the NDS, and the *Point Mesh Collision* goal enforces proper distance to other harness items, particularly between cables of different categories where offset distances are required.

After all single cables are processed, the workflow advances to the post-processing phase, where the generated paths are validated and reported.

4.3.3 Post-Processing

The post-processing stage of the workflow focuses on verifying and documenting the generated harness routings. Its main objective is to automatically evaluate the routing results according to defined geometric and physical design criteria, ensuring that the generated harnesses meet the essential constraints for integration and reliability. The verification process runs in a loop that iterates through each harness item and its corresponding cables, systematically checking all relevant parameters.

For the **minimum bending radius** requirement, a secondary loop examines the points along each reconstructed polyline within a defined range, verifying whether the local curvature satisfies the specified bending limits. For **intersection** checking, the workflow compares the voxelized NDS with the meshes of the final reconstructed cables and bundles, detecting any physical overlap between the routed geometry and restricted areas. The distance-to-structure verification, which relates to the **mounting interval** requirement (Table 3), ensures that cables and bundles maintain appropriate proximity to the spacecraft structure so that fixation points can be properly installed. Finally, the **distance between different cable categories** is verified by measuring the spatial separation between cables belonging to distinct electrical or functional groups across different harness items. This check confirms compliance with EMC requirements and routing safety constraints. As a reminder, segregation is only enforced between cables of different categories, whereas cables sharing the same category may be routed closer when allowed by the applicable design rules.

Beyond verification, the post-processing phase also includes automated reporting functions. The workflow calculates the Center of Mass (CM) of the whole harness and compiles a comprehensive report summarizing the routing results. This report is automatically generated in Microsoft PowerPoint, containing both numerical data and visual information. It includes key parameters, compliance indicators, and rendered images of the final routed harnesses to support design review and documentation. Any point with detected non-compliance, such as curvature violations, intersections, or insufficient clearances, is highlighted in the report, enabling engineers to identify and correct issues efficiently.

To enable this automated documentation, the workflow makes use of the Raphos Office Connector [45], a utility add-in provided by Raphos Ltd. that allows the creation and modification of Microsoft Word and PowerPoint files directly from within the computational environment. In the context of this work, Raphos Office nodes are used to programmatically generate PowerPoint presentations by composing slides from predefined layouts and populating them with calculated values, tables, and images. Text elements, tables, and figures are inserted into specific placeholders using a structured logic that maps workflow outputs to corresponding report sections.

This approach allows the reporting structure to remain consistent across different harness configurations while ensuring that all relevant results are documented automatically. By integrating analysis, verification, and reporting within a single workflow, manual post-processing effort is reduced and traceability between routing results and documentation is improved.

This automated verification and reporting process ensures that the generated harness designs can be systematically evaluated and refined in subsequent analyses, providing a solid foundation for the discussion of workflow limitations and validation results presented in the following sections.

4.3.4 Limitations

While the developed workflow demonstrates a structured approach for automating cable harness

routing in satellite systems, several limitations remain inherent to its current implementation. These constraints primarily stem from simplifying assumptions and computational trade-offs adopted to ensure feasibility within a low-code environment.

Although the routing algorithm incorporates collision avoidance by excluding NDS regions, minor intersections between cables and structural elements may still occur during the reconstruction phase. This effect is more likely when higher bending radii are applied, as the geometric relaxation performed by the Raphos Physics solver can slightly deform the cable paths beyond the voxel-based collision boundaries.

The workflow also aims to minimize path length, but the presence of additional constraints, such as maintaining proximity to mounting regions or enforcing bundle formation, means that the resulting paths are not globally optimal in terms of total cable length. Similarly, since the bending radius constraint is introduced only during the reconstruction stage, the original voxel-based routing generated by the A* algorithm does not inherently account for curvature, leading to deviations from the shortest possible route after physical relaxation. This behavior highlights a fundamental trade-off between computational simplicity and geometric fidelity, as incorporating curvature directly into the path-finding logic would require significantly more complex cost functions and increased runtime.

Another limitation concerns mounting interval verification, which is only addressed in the post-processing phase. The current version of the workflow checks the feasibility of fixation spacing after routing, rather than incorporating it as an active constraint during the path generation process. As a result, the placement of potential fixation points is verified a posteriori, which may occasionally necessitate manual adjustment.

Finally, the bundle definition approach relies on the cable with the shortest path as a reference and uses a user-defined distance threshold to identify the shared bundle region. This simplifies the automation process but can lead to non-optimal bundling configurations, especially when cables follow complex or partially parallel trajectories. The method ensures consistent bundling behavior but does not yet evaluate all possible bundle combinations or dynamic bundle branching scenarios. In addition, as a simplification in Path Finder II, the bundle region is not re-voxelized using the effective bundle diameter; instead, the voxel path of the individual cable is reused to reduce computational cost. This approximation may locally reduce clearance margins or introduce minor intersections, particularly in densely packed regions.

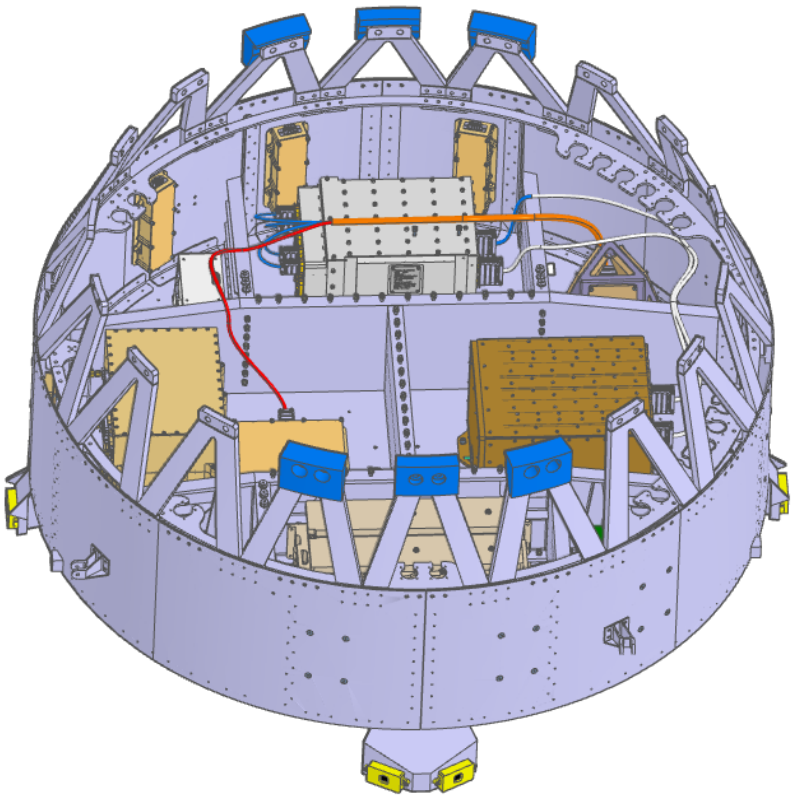
Despite these limitations, the workflow establishes a solid foundation for further refinement, offering a practical balance between automation, computational efficiency, and design flexibility within a parametric CAD environment. These aspects are considered when analyzing the workflow's performance and accuracy in the case study presented in the following chapter.

5 Use-Case and Implementation: Eu:CROPIS Mission

The workflow described in Chapter 4 was applied to the Eu:CROPIS spacecraft. To enable a controlled, reproducible comparison, a representative subset of bus harness items from the flown mission was selected (Figures 29–31). In the original Eu:CROPIS design process, these harnesses were routed manually in a CAD environment based on engineering experience, where general paths were defined by placing points and locally constraining spline orientations to guide the cables around structural elements and components. This manual reference serves as the baseline to compare to the automated workflow, while focusing on a subset keeps the CAD scope tractable and still exercises all major features of the method (bundling, category separation, wall-following DS constraints, and reconstruction).

In practical terms, the selected set includes (i) harness items with configuration of bundling and separated cables; (ii) short point-to-point runs, i.e., connections between components located close to each other that stress routing in confined corridors and reduce path redundancy; and (ii) cases with tight clearances near walls and structural ribs, i.e., narrow passages along the interior panels and stiffeners where the DS is thin and collision avoidance is most demanding. This combination is sufficient to validate the automation logic (DS shaping, prioritization, bundling, category separation, reconstruction, and checks) without requiring a full-bus re-route.

Figure 29: Eu:CROPIS bus segment.



The routed baseline from the original mission and the connectors used for this study are shown in Figures 30 and 31. This routing serves as the visual ground truth for qualitative comparison in the Results chapter (Chapter 6). The colors used to represent the harnesses are illustrative and intended only to support visual distinction. The classification into different cable categories should be interpreted based on the corresponding Table 4 rather than on color coding.

Figure 30: Selected harness items from Eu:CROPIS Bus. Harness colors are illustrative and used solely for visual differentiation.

(a) Selected harness items

(b) Top view of the selected harness items

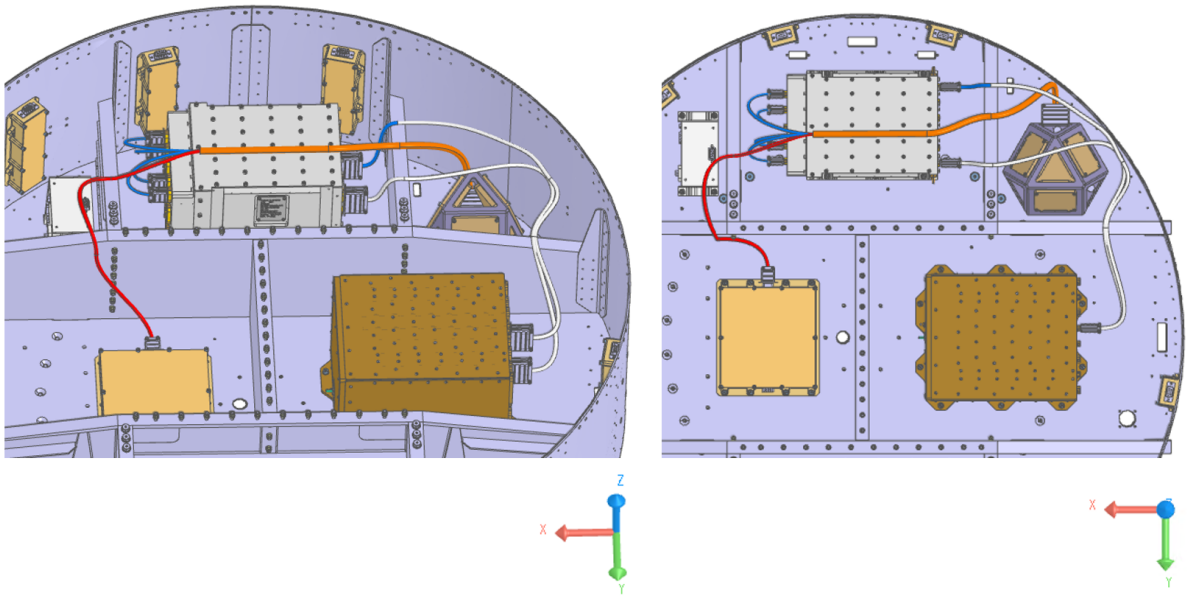
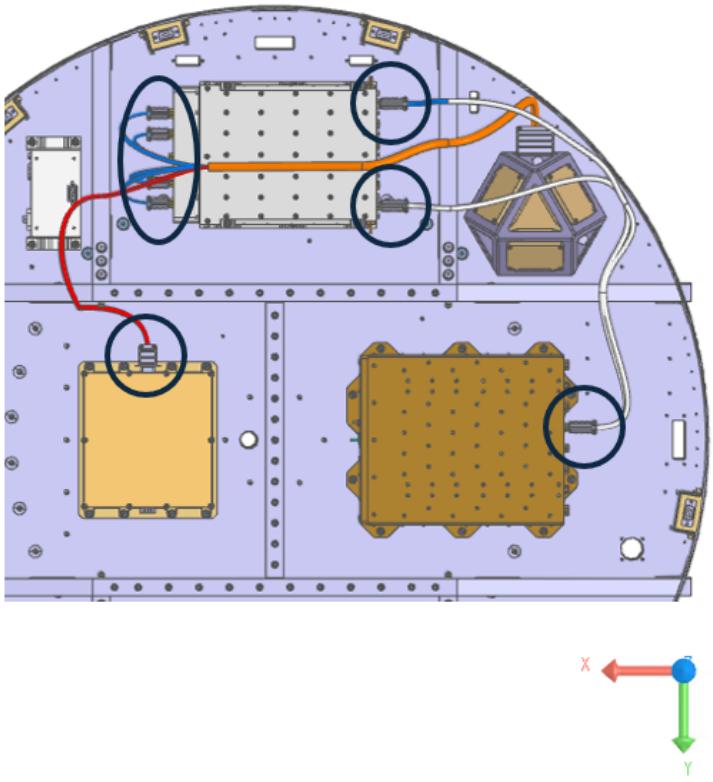


Figure 31: Selected connectors from Eu:CROPIS.



Inputs

The inputs used for the simulation comprised (i) a `.step` assembly with the bus structure and components, (ii) a JSON file listing the geometry names for extraction (structure vs. components - Table 5), and (iii) a `.json` file defining harness items (connectors, cable diameters, categories, user priority, and density). The Table 4 summarizes the harness data used, which the correspondent `json` listings are provided in Appendix A.

Table 4: Selected harness data from Eu:CROPIS.

Harness Name	**Connectors**	**Priority**	**Connections**	**Diameter** [mm]	**Category**	**Density** [kg/m³]
EC-RYSAT-07-03-000	CDH-OBC-J101, EPS-PCDU-J122	None	CDH-OBC-J101, EPS-PCDU-J122	3	2	2472.06
EC-RYSAT-07-03-001	CDH-OBC-J201, EPS-PCDU-J121	None	CDH-OBC-J201, EPS-PCDU-J121	3	2	2472.06
EC-RYSAT-07-03-008	AOCS-GYRO-0001-J1, CDH-OBC-J300, CDH-OBC-J301, CDH-OBC-J302, CDH-OBC-J303, AOCS-GYDC-0001-J2	None	AOCS-GYRO-0001-J1, AOCS-GYDC-0001-J2	3	1	2472.06
		None	AOCS-GYRO-0001-J1, CDH-OBC-J300	2.2	2	2472.06
		None	AOCS-GYRO-0001-J1, CDH-OBC-J301	2.2	2	2472.06
		None	AOCS-GYRO-0001-J1, CDH-OBC-J302	2.2	2	2472.06
		None	AOCS-GYRO-0001-J1, CDH-OBC-J303	2.2	2	2472.06

Table 5: Selected geometries from Eu:CROPIS.

Bus	Components
EC-BSFA-03-01-000-31	EC-BSFA-03-07-000-31, EC-RYSAT-06-01-000-31, aa-PFM1-assembly, EC-RYSAT-09-05-000-31, EC-RYSAT-09-06-000-31_Assembly, EC-RYSAT-14-01-000-31

The component selection (Table 5) was driven by spatial relevance to the chosen harness items (occupied volume, connector locations, and potential routing obstacles). The correspondent `json` file is also provided in Appendix B.

Simulation parameters

All simulations adopt a global reference frame in which the bottom plate of the bus is aligned with the XY-plane and centered near in $(0, 0, 0)$, with the $+Z$-axis pointing upward. As described in the Chapter 4, the Path finding is performed on a voxelized DS. The voxel resolution is selected per case to balance runtime and geometric fidelity: coarse grids are used for provisional tasks, such as temporary section pre-selection with resolution of 15 mm and bundle-interval detection with resolution of 30 mm. Whereas finer grids are employed for the definitive solution with resolution of 1.7 mm. Although the DS can be subdivided into spatial sections, this subdivision is optional; when present, the workflow itself (not the user) determines which sections are relevant to each harness item and restricts subsequent computations accordingly.

Minimum-curvature compliance during reconstruction is enforced via a target bending radius set above the normative minimum. For the present configuration, all conductors are smaller than AWG size 10 and non-coaxial. Consequently, based on the Table 1 according to NASA-STD-8739.4A [32] classes, the minimum reference is $R_{min} = 3 \times OD$. Because the solver converges toward a specified value rather than merely exceeding it, a user-defined safety factor $f_{BR} > 1$ is applied and the reconstruction target becomes $R_{target} = f_{BR} \times R_{min}$. For this use-case, the factor used was $f_{BR} = 1.5$.

For bundled segments, the external diameter is estimated from cross-sectional area equivalence, stated in the Equation 3. The bundle in this use-case has one cable $d_1 = 3.0\ mm$ and four cables with diameters $d_{2,3,4,5} = 2.2\ mm$. As result, the external diameter for the bundle segment is $OD_{bundle} = \sqrt{3.0^2 + 2.2^2} = 5.3\ mm$.

To detect the bundle region, as explained in Section 4.3.2.2, the workflow evaluates the spatial relationship between the shortest cable (used as the bundle reference) and the remaining cables of the same harness item using a predefined distance threshold X. In this study, this threshold is set to $X = 55\ mm$. The magnitude of this value is derived from the geometric characteristics of the harness connectors: the largest connector associated with the harness has a maximum external dimension of 52.81 mm. A small safety margin is added to this dimension, resulting in the chosen threshold, which ensures that cables intended to form a bundle are sufficiently close to be treated as a single harness segment.

Furthermore, spacing constraints between different cable categories are enforced using a minimum separation of 20 mm. Mounting-interval compliance is evaluated in a post-processing step using the default values defined in Table 3.

The physics-based reconstruction employs goal weights chosen to reflect the relative importance of each constraint. The *Line Spring* goal (weight 6000) establishes the basic spring-particle

representation along each polyline and provides the elastic backbone for shape updates. The *Pull Point Update* goal for bundles (weight 2000) biases co-located points toward the bundle reference while remaining sufficiently weak to allow local accommodation around obstacles; a lower value is used here than for structural goals because the bundle co-routing already derives from common references. The *Bending Rod* goal (weight 10000) is assigned a high value to prioritize curvature control and ensure that the trajectories meet the target bending radius. The *Anchor* goal employs a restrictive weight of 1×10^8 to fix start and end points, preventing any drift at the connector interfaces. The *Direction* goal (weight 2000) aligns the initial segments with connector normals; the moderate value reflects that geometry is already initialized normally and only requires guidance rather than strict enforcement. Collision management is handled by two complementary goals: *Point Voxel Collision* (weight 10000) keeps the solution outside the NDS represented in voxels, while *Point Mesh Collision* (weight 10000) maintains the distance from previously routed cables and allows compliance with spacing between categories.

These weight settings were applied consistently in both the bundled- and single-cable harness loops, with the exception of the *Pull Point Update* goal, which is omitted in the single-cable case since no bundle reference must be maintained.

Importantly, the selected weight values are qualitative and relative in nature: their magnitudes are defined primarily in relation to each other to reflect the order of importance of each physical or geometric constraint during reconstruction. For instance, goals enforcing fixed boundary conditions, such as the *Anchor* goal for the cable at connector locations, are assigned significantly higher weights to ensure positional stability.

Therefore, the chosen values are configuration-dependent and may require recalibration for different harness layouts, connector types, or spatial constraints. An exploratory optimization-based analysis of these weights is presented in the Outlook chapter (Chapter 7), illustrating how adaptive tuning could further improve reconstruction quality. However, this optimization is not part of the methodology of the present thesis.

For the Raphos physics solver, it was utilized a fixed number of 100 iterations. A time step of 1.00 was applied, defining the integration step size for each iteration. While smaller time steps can increase numerical accuracy, this value was selected as a compromise between solution quality and computational efficiency. A damping factor of 0.9 is used to progressively reduce particle velocities at each iteration according to Equation 2 presented in Chapter 4.2.2. This damping stabilizes the solution and prevents oscillatory behavior. Both the absolute and relative solver tolerances were set as default equal to zero, meaning that convergence is controlled exclusively by the iteration count rather than displacement-based stopping criteria.

Software and hardware

All runs were performed in Synera 25.03 (Magnificent Markus) with Raphos Physics 1.1.11.9. The workstation had an Intel(R) Xeon(R) E5-2687W v4 @ 3.00 GHz (2 processors), 256 GB RAM @ 2400 MHz and NVIDIA Quadro P2000 (5 GB). This configuration comfortably supported voxel-based path-finding and physics-based reconstruction at the resolutions stated above.

6 Results and Discussion

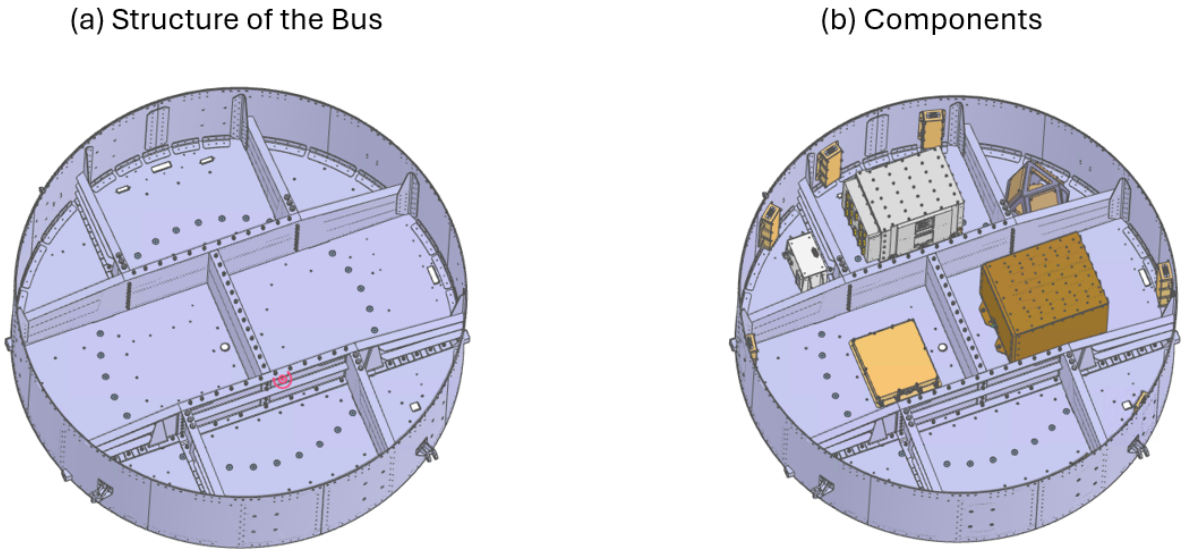
This chapter presents the results of the developed workflow, applied to the Eu:CROPIS case study, and discusses its performance across all stages of the process—from pre-processing to post-processing and final comparison with the original routing. Each subsection corresponds to one major workflow phase: Pre-Processing I and II, Routing Loop I (bundled harness items), Routing Loop II (individual cables), Post-Processing, and the final evaluation of results, including reconstruction goal optimization. Together, these steps illustrate the complete transformation from raw CAD data to validated cable paths. These results directly address the research questions defined in Chapter 3 by demonstrating how a combined graph-based and physics-based workflow can automate harness routing, integrate multiple design constraints within a CAD environment, and substantially reduce design time while maintaining engineering validity.

6.1 Pre-Processing I

The first stage of the workflow involves the identification and classification of the input geometries necessary for defining the DS and NDS. As described in Chapter 4, this step uses the imported `.step` model to separate the structural elements of the spacecraft bus from its internal components. The result is shown in Figure 32.

Figure 32(a) depicts the extracted structure of the Eu:CROPIS bus, corresponding to the base surfaces and supporting frames that delimit the routing volume, while Figure 32(b) shows the selected components that serve as routing obstacles and contain the relevant connector interfaces. These geometries directly correspond to the selection summarized in Table 5.

Figure 32: Pre-processing I – Extracted geometries.

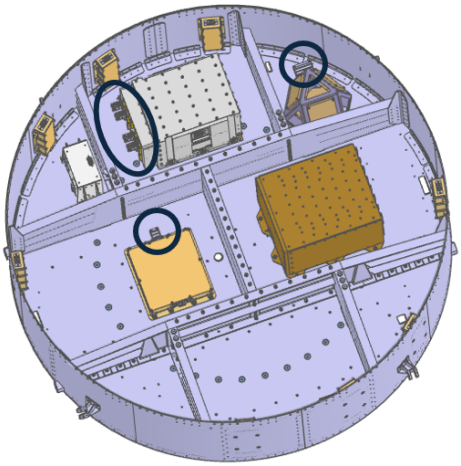


At this stage, the connectors are only identified according to the information from the harness data (Table 4), without yet defining their precise orientation or normal vectors. Their spatial locations are marked to ensure correspondence with the components as connection interfaces. Figure 33 illustrates the identified connector regions: (a) shows the connectors associated with the harness item containing a bundle, while (b) shows those related to the harness items consisting of single cables. The detailed extraction of connector positions and normal directions will be performed inside each routing loop, where they are used to define start and end points for the A* path-finding and reconstruction stages.

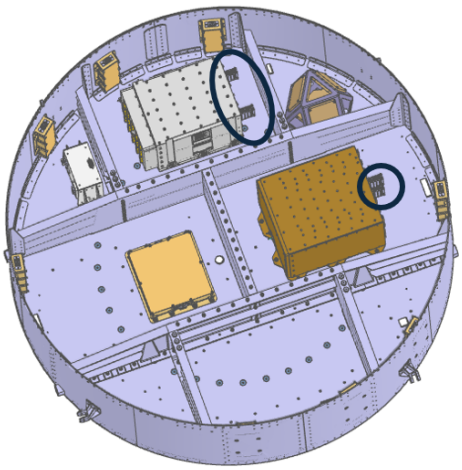
Overall, Pre-Processing I establishes the geometric foundation for the entire workflow. By separating structural and functional elements and preparing the environment for voxelization later, this step ensures that all subsequent routing and reconstruction processes operate on consistent and physically meaningful spatial data.

Figure 33: Pre-processing I – Extracted connectors.

(a) Connectors for the harness item with bundle



(b) Connectors for the harness items with single cables

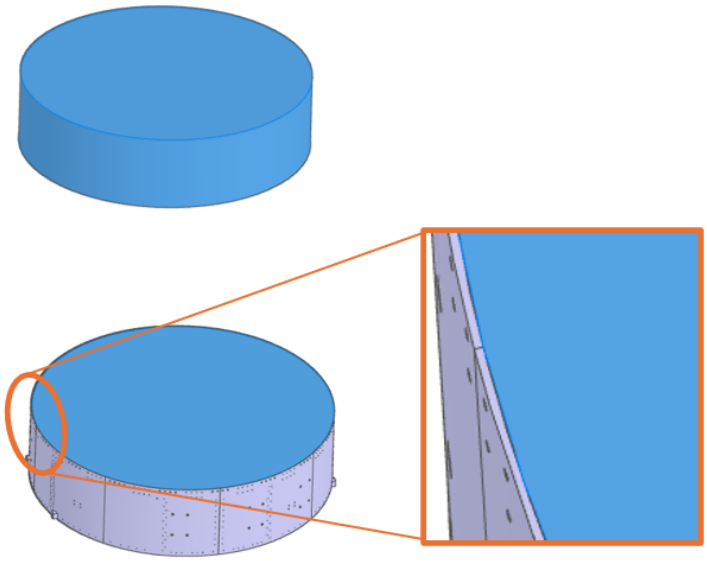


6.2 Pre-Processing II

Following the identification of the geometries and components in Pre-Processing I, the second pre-processing stage focuses on generating the volumetric DS and defining the DS used in the routing process. This step establishes the spatial boundaries in which the routing algorithm operates, ensuring that the cables are constrained to realistic paths while avoiding intersections with the spacecraft's structure and hardware.

The DS is automatically generated as a broad cylindrical region that encompasses the spacecraft bus and all relevant internal components. This representation defines the global area within which routing operations are permitted, and it serves as the reference domain for voxelization and path-finding. The chosen cylindrical form is consistent with the Eu:CROPIS geometry and simplifies subsequent spatial calculations. Figure 34 shows the automatically generated DS volume. This region encloses the entire bus while leaving sufficient margin to accommodate cable routes close to the internal walls.

Figure 34: Pre-processing II – Broad region for the DS defined as cylindrical volume. The zoomed-in view highlights that the DS is defined strictly within the interior of the structure, excluding the structural boundaries, which are not considered part of the DS.



Within the DS, the workflow identifies all NDS volumes that must be excluded from routing. These include the spacecraft's structural elements, mounted components, and mechanical details such as screws. Figures 35 and 36 show the extracted NDS geometries. Figure 35 illustrates the NDS corresponding to the bus and internal components, while Figure 36 focuses on the screws, which are treated as individual NDS objects to ensure that no routing path intersects their local volumes. Including these smaller details improves spatial fidelity during voxelization and prevents unrealistic cable placements in tightly constrained regions.

Figure 35: Pre-processing II: NDS – Bus and components.

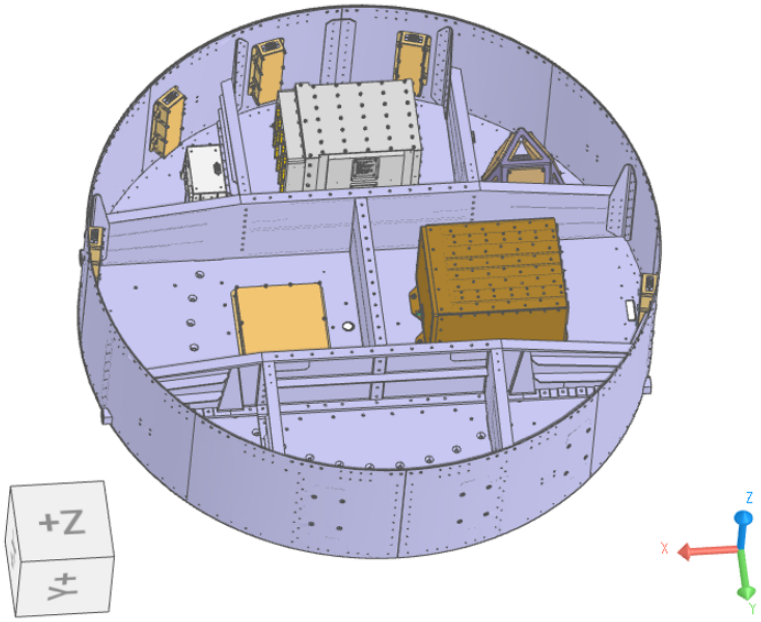
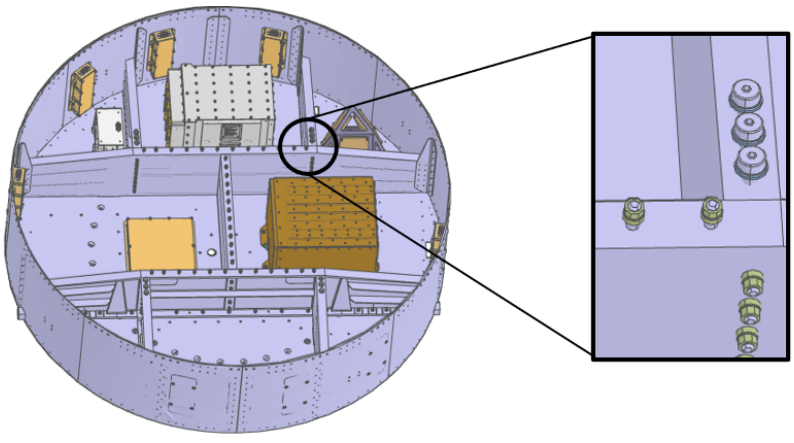


Figure 36: Pre-processing II: NDS – Screws.

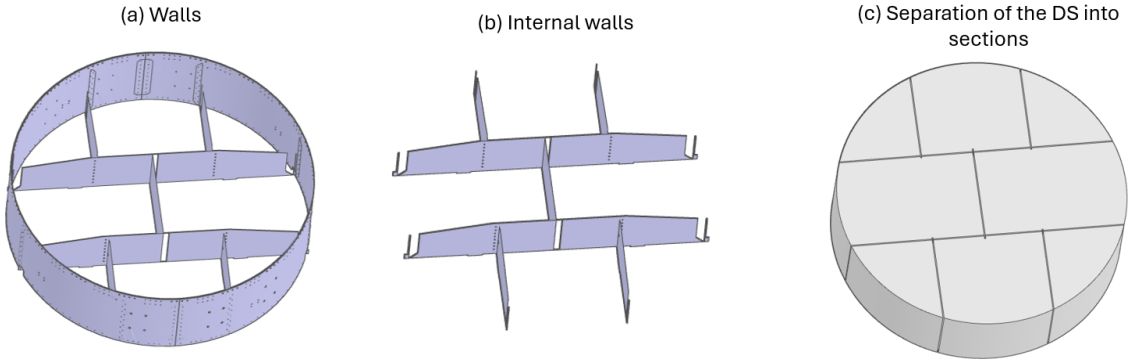


After the DS and NDS are defined, the workflow includes an optional step in which the routing volume is subdivided into smaller spatial sections based on the main structural walls of the spacecraft. This feature can be adapted for different geometries and is particularly useful in large or complex assemblies, as it enables the activation of only the spatial regions relevant to a specific routing task, thereby reducing computational time and memory usage in the voxel-based path-finding stage.

Although optional, this segmentation was applied in the Eu:CROPIS use case to optimize performance. The spacecraft bus features multiple internal ribs and compartments, which naturally divide the routing volume into distinct regions. By aligning the DS sections with these structural boundaries, the algorithm can limit voxelization and routing calculations to only the regions containing regions of interest, improving efficiency without compromising accuracy.

Figure 37 illustrates this process. Figures 37(a) and (b) show the extracted external and internal walls, which served as the reference surfaces for generating the partitions, while Figure 37(c) presents the resulting DS subdivision.

Figure 37: Pre-processing II: Separation of the DS into sections.



6.3 Routing Loop I: Bundled Harness Items

The first routing loop focuses on the harness item that contains multiple cables grouped into a bundle. This loop demonstrates the complete workflow from connector extraction to routing, bundling, and reconstruction within the Raphos Physics environment. Each stage combines geometric reasoning with automated spatial analysis, which aims to approximate compliance with both design and physical constraints.

In this case study, only one iteration of Routing Loop I was performed, since the Eu:CROPIS configuration used in this thesis represents only a partial harness of the spacecraft, and within this subset, there is only one harness that includes bundled cables. This setup enables a focused analysis of the workflow's behavior for bundled routing without the complexity of multiple bundle cases.

Pre-Processing III: Extraction of Connectors and Reference Data

At the start of the loop, the connectors corresponding to the harness item are identified and extracted. Their positions and normal vectors are computed directly from the CAD geometry, defining the entry and exit points for the routing process. The algorithm identifies both start and end connectors, calculates their local coordinate frames, and creates auxiliary reference points along their normal directions. These points are later used to initialize the first routing segments and avoid intersections with the connector surfaces.

The process is illustrated in Figure 38. The table schematically summarizes the extraction steps: (1) detection of the connectors, (2) identification of the reference points and normal vectors, and (3) creation of the initial lines extending from the connectors in the normal direction. This information defines the spatial conditions required for the subsequent A* path-finding operations.

Pre-Processing IV: Section Filtering and Local DS Definition

The next step selects the relevant spatial sections of the DS based on the previously defined segmentation (Section 6.2). Since the connectors of this harness are concentrated in a specific region of the spacecraft bus, only the corresponding DS sections are activated to reduce computational effort.

To determine which sections are relevant, a coarse path-finding step is first performed using a rough, low-resolution DS (Figure 40-a). This preliminary routing (Figure 40-b) provides an approximate trajectory that guides the workflow in selecting only the spatial sections intersected by the initial path (Figure 40-d). This approach effectively narrows the active DS to the regions that are most relevant for the harness routing.

Figure 39 shows the selected DS sections for the current harness item, highlighting the reduced spatial domain considered by the workflow.

After section filtering, the voxelized DS is locally refined around walls and obstacles, ensuring that the path-finding process avoids NDS regions while maintaining feasible clearance margins. Figure 40 illustrates the resulting DS configuration, including the volumes near the connectors and the extension along their normal directions. This localized DS serves as the input for the first routing stage (Path Finder I), which computes the voxel-based cable trajectories.

Figure 38: Harness item with bundle – Pre-processing III: Extracted connectors.

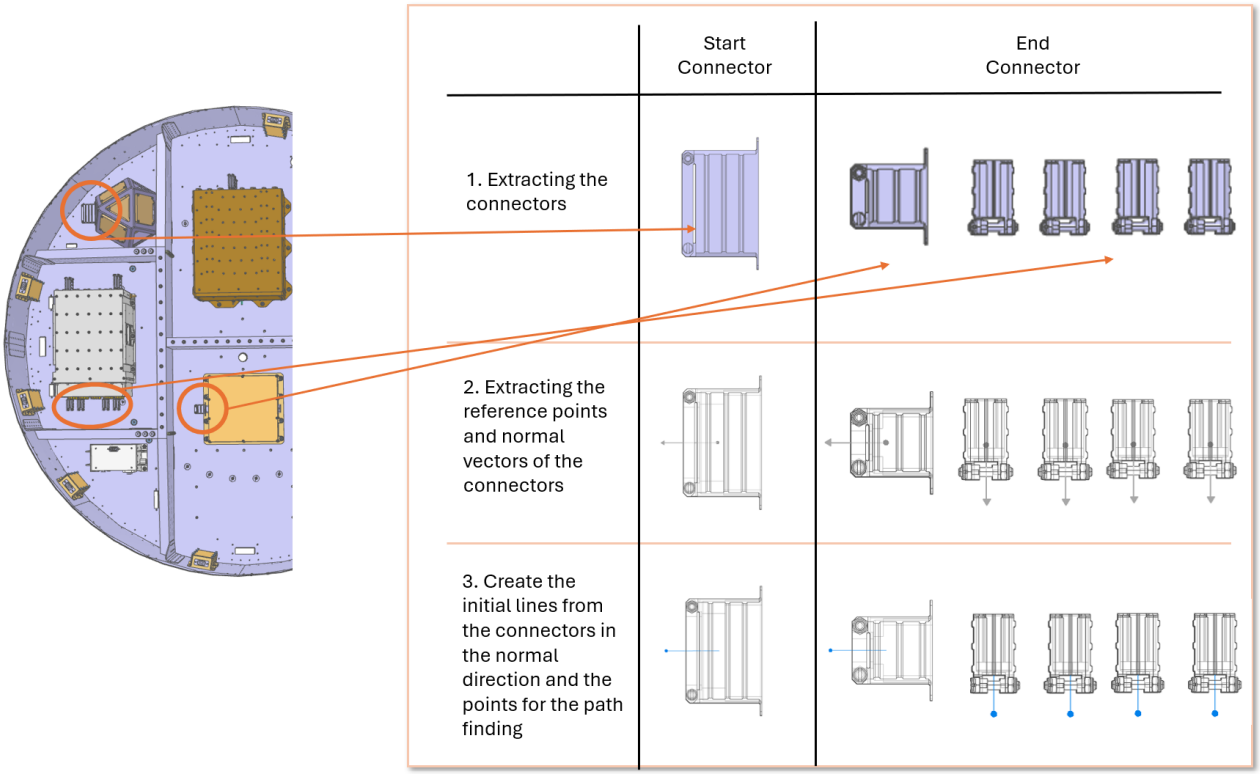


Figure 39: Harness item with bundle – Pre-processing IV: Selected sections.

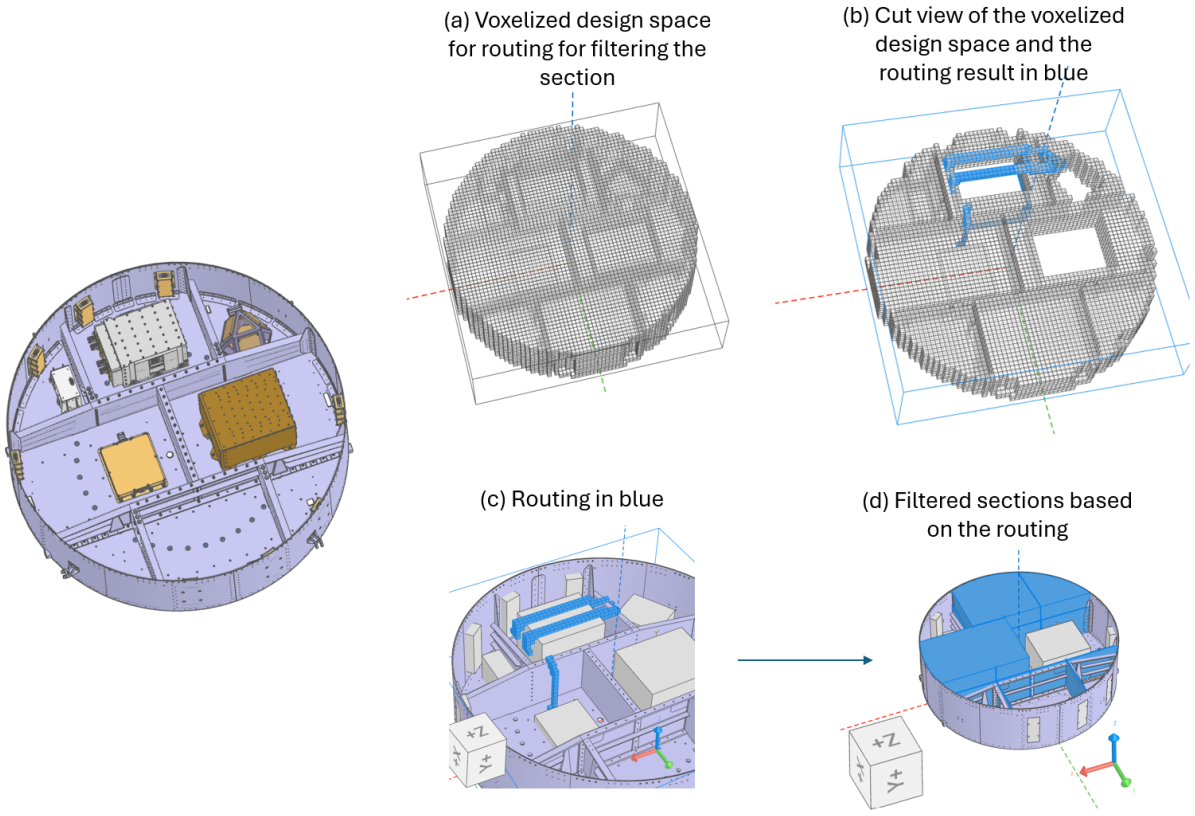
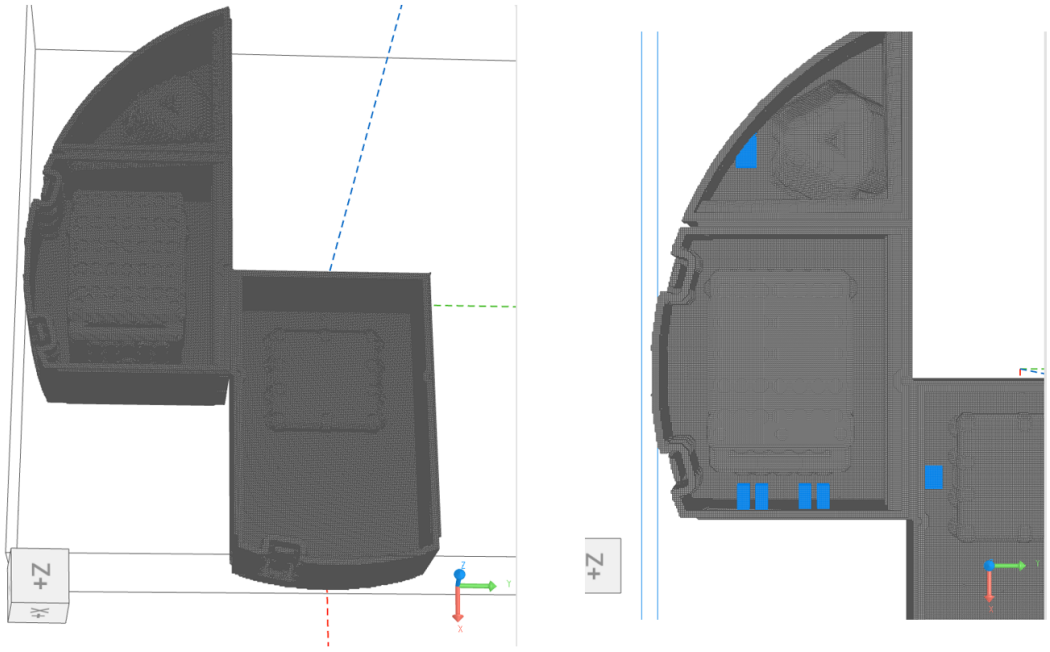


Figure 40: Harness item with bundle – Pre-processing IV: DS.

(a) Design Space limited near the walls and obstacles

(b) DS defined in the normal direction of the connectors



Routing and Reconstruction

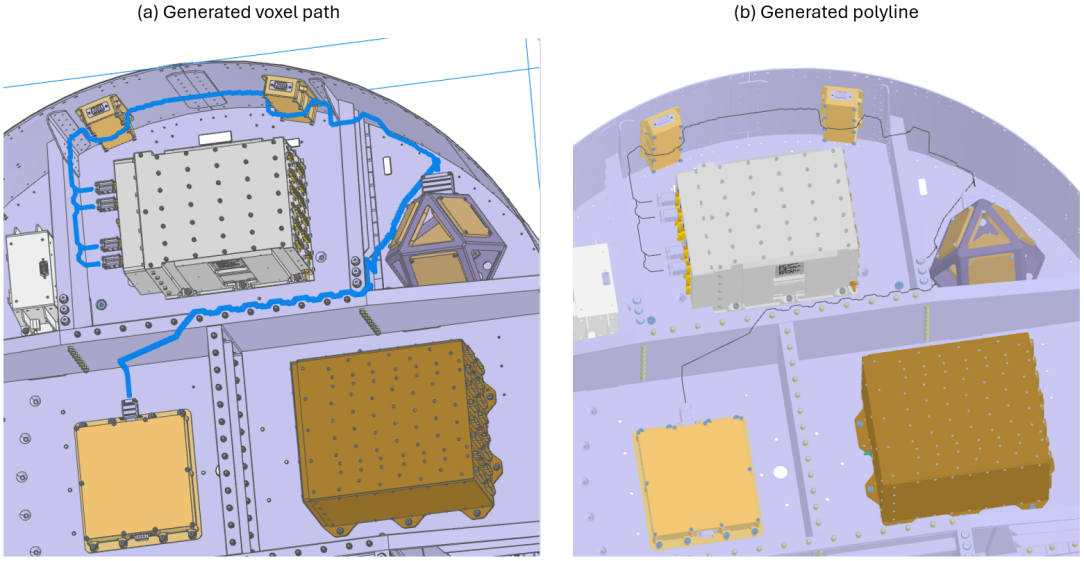
Once the local DS is established, the workflow proceeds to the routing phase, which consists of two A*-based path-finding stages and one physics-based reconstruction stage.

In the first path finding stage (Path Finder I), the A* algorithm computes the initial baseline paths for all cables belonging to the same harness item (Figure 41). These paths are generated within the refined voxelized DS, respecting NDS boundaries and the clearance margins defined during pre-processing. This initial routing provides the geometric foundation from which the bundled region can be identified.

Figure 41: Harness item with bundle – Path Finder I.

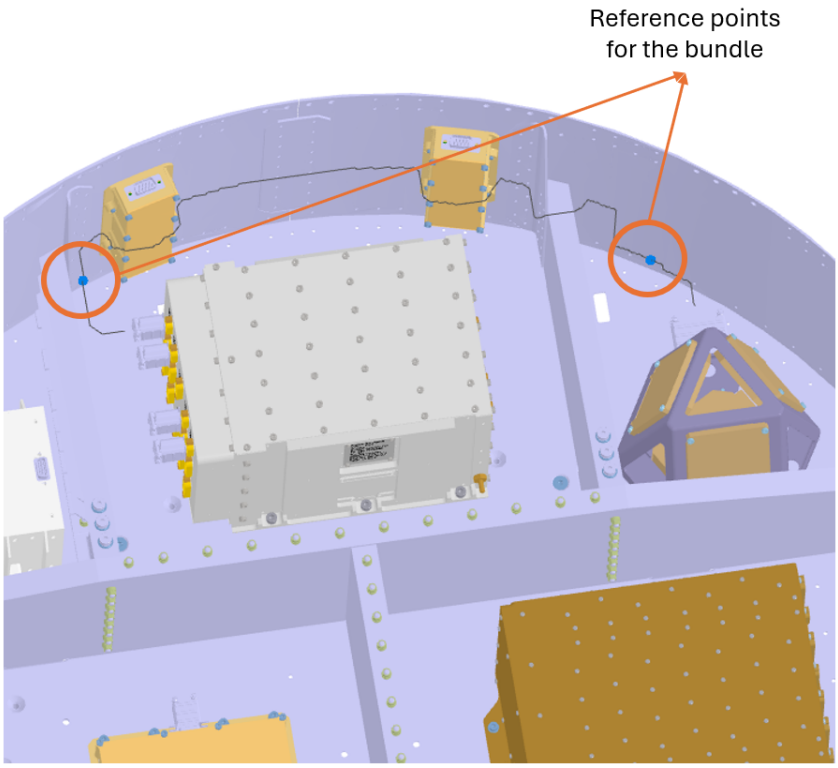
(a) Generated voxel path

(b) Generated polyline



Next, the bundle definition is carried out. The workflow automatically selects one of the routed cables, the shortest one, as the reference to define the bundle path (Figure 42). Based on this reference, the algorithm determines the regions where the remaining cables stay, establishing the start and end points of the bundled segment.

Figure 42: Harness item with bundle – Selected cable for the bundle.



After the bundle interval is defined, the second path finding stage (Path Finder II) is executed to generate the final trajectories of all cables, now constrained by the bundle path (Figure 43), while always considering the spacing rules between categories defined by the DS. This step refines the voxel-level path, providing a consistent basis for physical reconstruction.

Finally, the reconstruction and simulation phase employs Raphos Physics to transform the voxel-based polylines into smooth, physically realistic cable paths (Figure 44). In this phase, each cable is modeled as a spring–particle system, where multiple goals are applied with specific stiffness weights presented in the Chapter 5.

During the simulation, the solver iteratively minimizes the total system energy while satisfying these geometric and physical goals. The Bending Rod constraint is applied according to the minimum bend radius defined in Chapter 5, where all radii were calculated using the equivalent bundle outer diameter derived from the area-based formulation in Equation 3. This setup tries to make the reconstructed geometry comply with the bending requirements corresponding to each cable or bundle configuration, acknowledging that full enforcement depends on the solver's convergence and the interaction of competing geometric constraints.

The final configuration, shown in Figure 44, represents the reconstructed routing that balances geometric feasibility, mechanical plausibility, and spatial compliance within the spacecraft bus.

Although the minimum bending radius is applied during the reconstruction stage rather than embedded directly into the voxel-based A* path-finding, this is consistent with several approaches in the literature where routing and bend-radius handling are treated as sequential

Figure 43: Path Finder II.

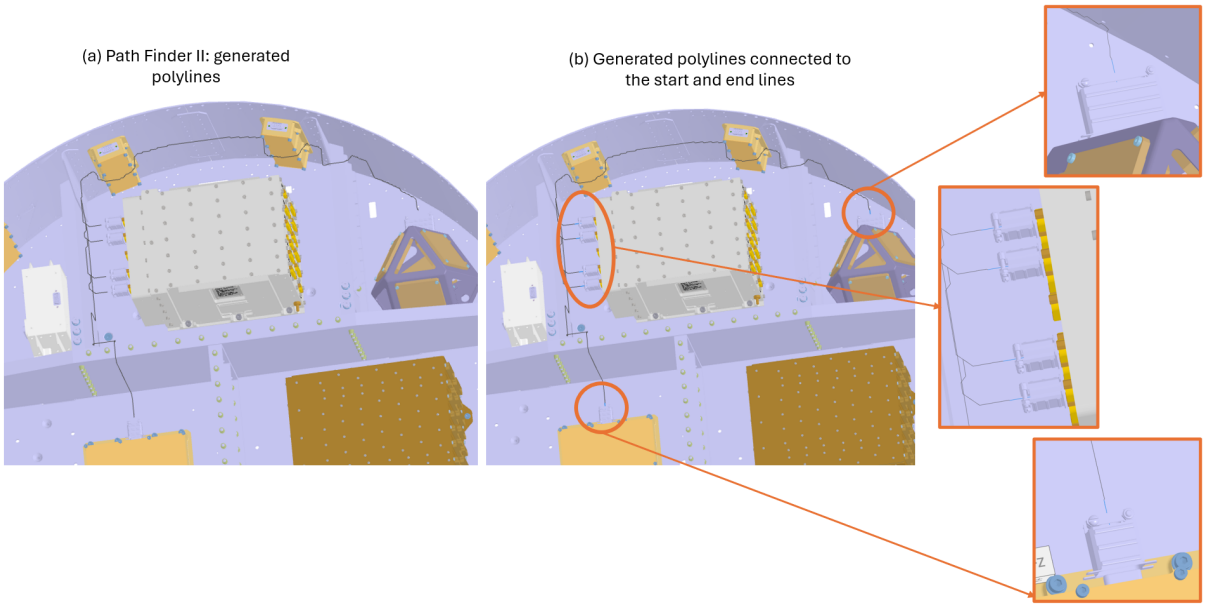
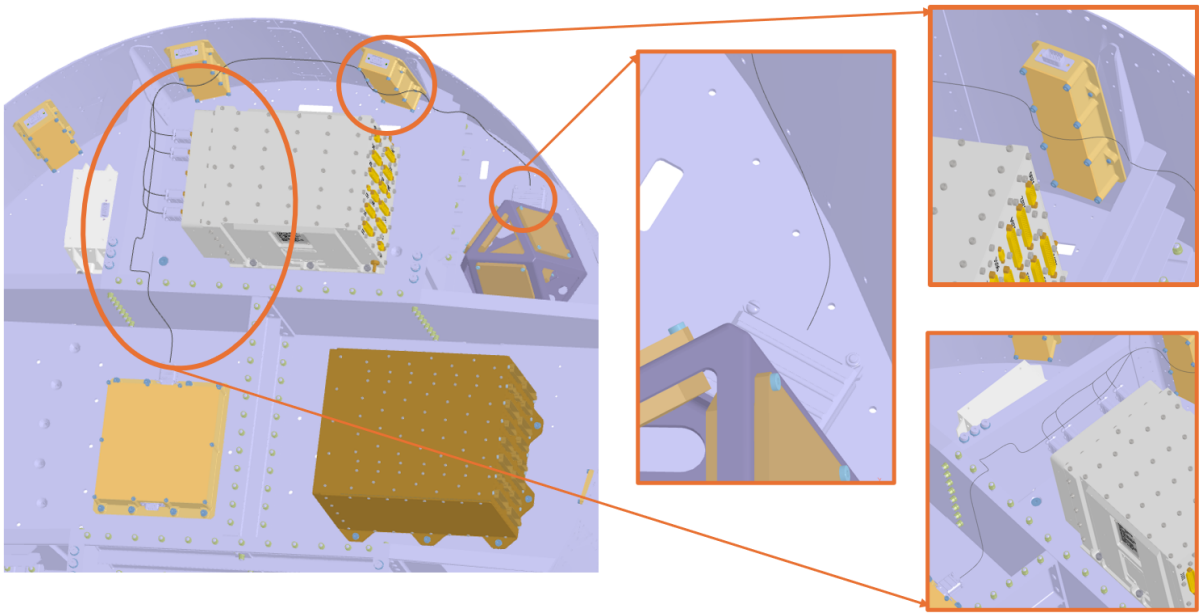


Figure 44: Reconstructed routing.



steps. In this thesis, the contribution is the explicit and CAD-integrated coupling of (i) a voxel-based graph search that produces feasible routes inside a deliberately restricted DS (near walls and components for mountability), (ii) an automated bundling logic that identifies shared intervals within the routed trajectories, and (iii) a physics-based reconstruction stage that converts voxel paths into smooth cable polylines under competing goals (collision avoidance, connector alignment, bundle cohesion, and curvature control). Importantly, curvature is not applied as a purely geometric smoothing operation: instead, the reconstruction uses a spring–particle model with a user-defined curvature target derived from the minimum bending radius (including the safety factor used in this work), which reduces violations while remaining computationally tractable in a low-code CAD workflow.

Therefore, compared to approaches that either prioritize shortest-path computation alone or rely on subsequent geometric correction, the presented workflow provides an end-to-end pipeline that links routing feasibility, bundling behavior, and physically plausible reconstruction in the same automated environment.

6.4 Routing Loop II: Single-Cable Harness Items

After the routing of the bundled harness item, the workflow proceeds to the routing of the remaining harness items composed of individual cables. In this configuration, the routing loop is executed separately for each cable, ensuring that the path-finding process accounts for previously routed cables through NDS subtraction.

The following subsection presents the results for the first iteration, corresponding to the first single-cable harness item.

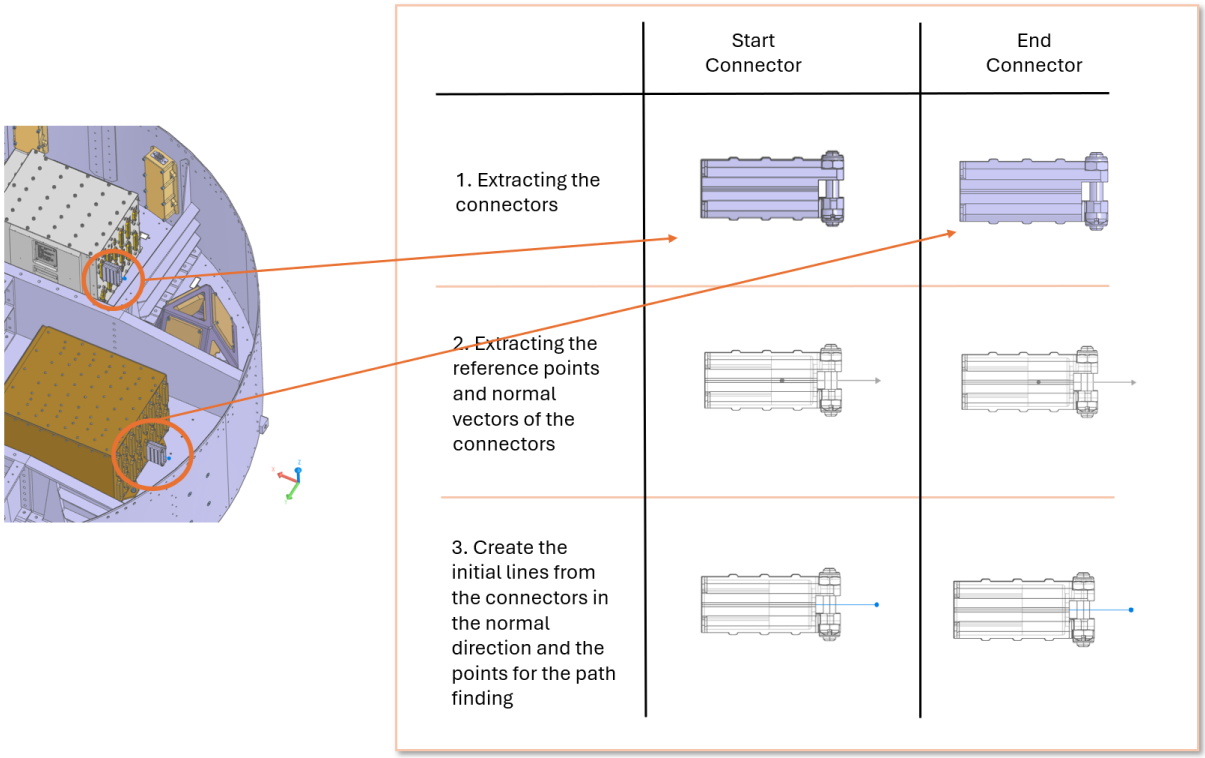
6.4.1 First Iteration (j=1)

Pre-Processing III: Connector Extraction

In this step, the workflow automatically identifies the start and end connectors associated with the harness item. As illustrated in Figure 45, the positions and normal vectors of the connectors are extracted, providing the necessary geometric data to define the connection direction and establish the reference geometry for routing initialization.

Based on these vectors, short initial lines are generated from each connector in its respective normal direction. The endpoints of these lines are then extracted and used as the initial path finding points, serving as anchor references for the subsequent routing step. This ensures that the routing process begins with a geometrically consistent alignment between the cable and connector interfaces.

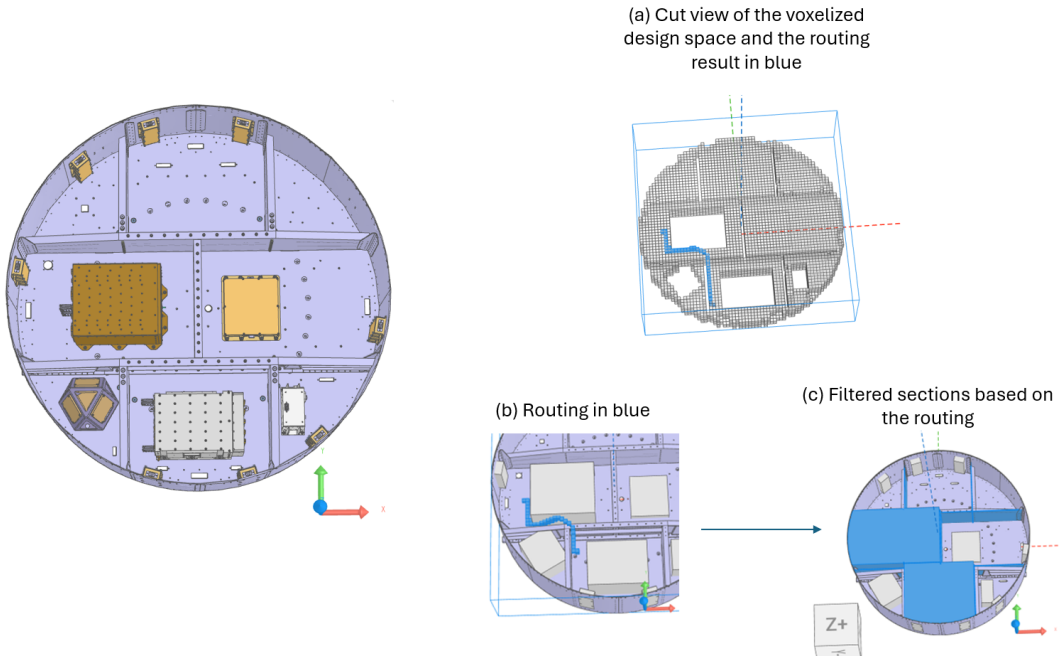
Figure 45: Harness item with single cable j=1 – Pre-processing III: Extracted connectors.



Pre-Processing IV: Section Filtering and Local DS Definition

Similar to the previous loop, the workflow performs section filtering to isolate the relevant DS regions associated with the routing task. An initial coarse path-finding is executed within a rough DS configuration (Figure 46-a), and based on the resulting trajectory, the sections intersected by the path (Figure 46-b) are selected (Figure 46-c). This strategy ensures that the routing focuses only on the necessary spatial regions, reducing computational load while maintaining sufficient coverage of potential routing corridors.

Figure 46: Harness item with single cable j=1 – Pre-processing IV: Selected sections.



The refined DS is subsequently created with finer voxel resolution and locally constrained near the walls and components. Additionally, the DS is extended only in the normal directions of the connectors to ensure spatial continuity with the interface geometry (Figure 47-b).

After this refinement, obstacles such as screws and mounting points are removed from the DS (Figure 47-c), preventing unnecessary collision checks and ensuring a clean routing domain.

Finally, the workflow subtracts the volumes corresponding to the routes of previously processed cables (48). This subtraction step also considers the distance constraints between cables of different categories, ensuring that the new path maintains proper separation from existing harnesses and avoids spatial interference.

Routing and Reconstruction

The routing process for the single cable begins with the execution of the A* path-finding algorithm within the refined DS. Figure 49 illustrates the generated voxel-level path, which ensures spatial feasibility by avoiding collisions with both structural components and previously defined NDS volumes.

Subsequently, the voxel-based path is converted into a polyline representation and connected to the short initial lines at the connector interfaces, ensuring continuous geometry between the connectors and the cable trajectory (Figure 50).

Figure 47: Harness item with single cable j=1 – Pre-processing IV: DS.

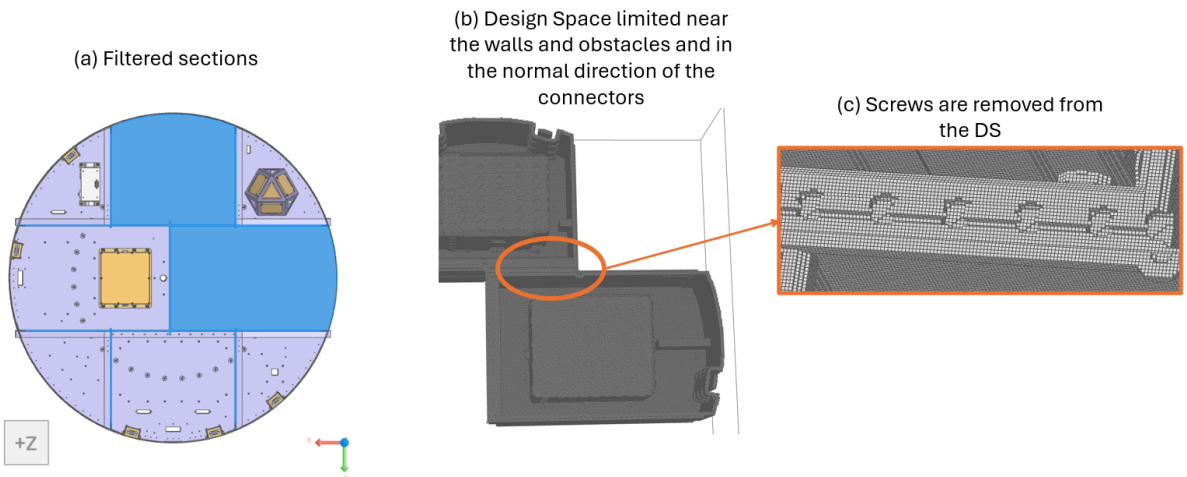


Figure 48: Harness item with single cable j=1 – DS considering different cable categories.

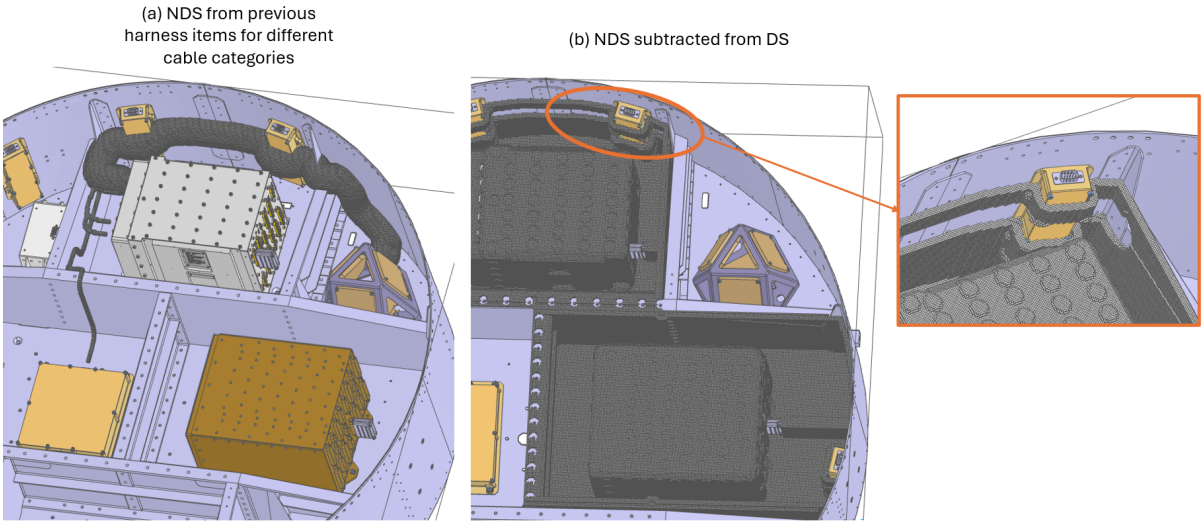


Figure 49: Harness item with single cable j=1 – Path Finder: Generated voxel path.

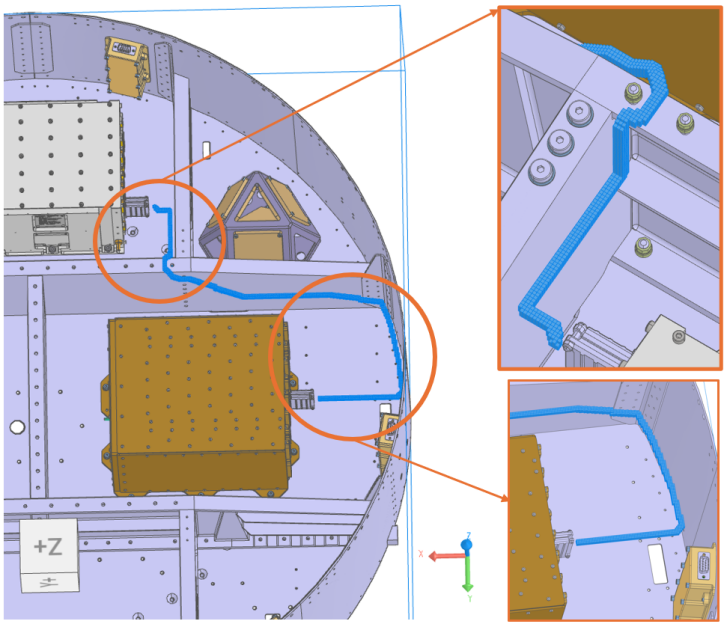
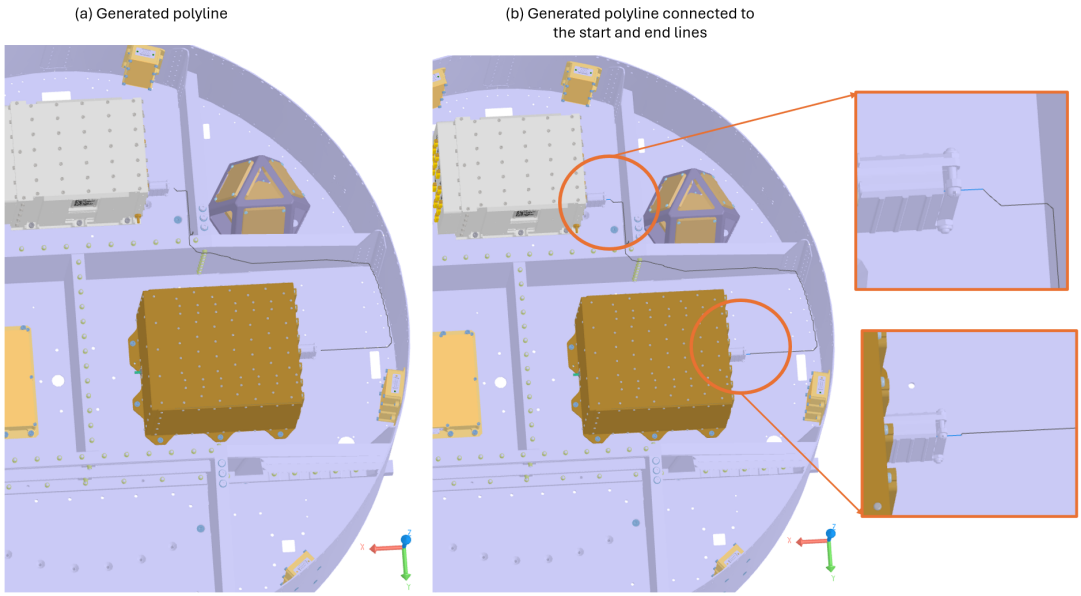


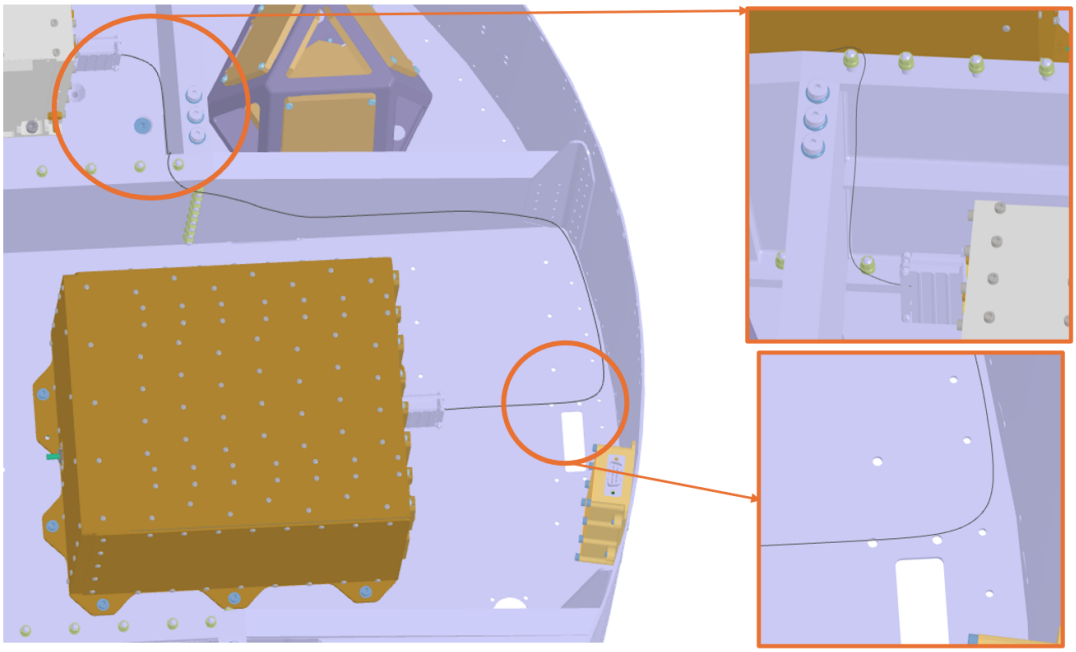
Figure 50: Harness item with single cable j=1 – Generated polyline path.



Finally, the cable geometry is reconstructed and physically simulated using Raphos Physics (Figure 51). The employed goals configuration were as presented in Chapter 5, combining geometric and physical goals to refine the cable curvature and enforce minimum bending constraints.

The bending radius is also based on the nominal cable diameter and equivalent bundle outer diameter derived from the Equation 3 in Chapter 5. As before, the solver tries to satisfy these bending constraints while balancing other geometric requirements, such as avoiding intersections and preserving the connector orientations. The final reconstructed polyline is shown in Figure 51

Figure 51: Harness item with single cable j=1 – Reconstructed polyline.



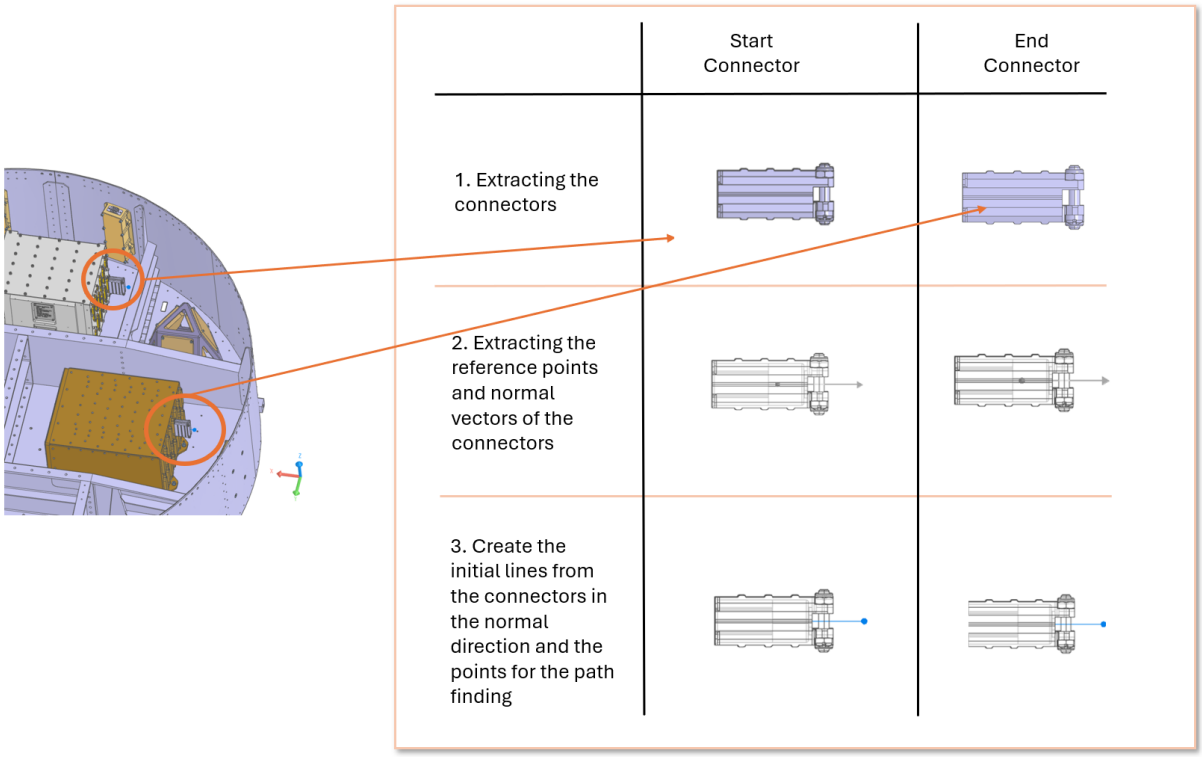
6.4.2 Second Iteration (j=2)

The second iteration of Routing Loop II follows the same general procedure as the previous one but applies it to the next single-cable harness item. This iteration further demonstrates how the workflow dynamically updates the DS and NDS between routing operations to preserve spatial feasibility and proper separation among cables of different categories.

Pre-Processing III: Connector Extraction

In this step, the start and end connectors of the harness are automatically identified. Their positions and normal vectors are extracted, as illustrated in Figure 52, providing the geometric basis for the connection direction. As before, short initial lines are generated along the connectors' normal directions, and the endpoints of these lines are taken as the initial path-finding points, serving as anchor references for the routing algorithm.

Figure 52: Harness item with single cable j=2 – Pre-processing III: Extracted connectors.

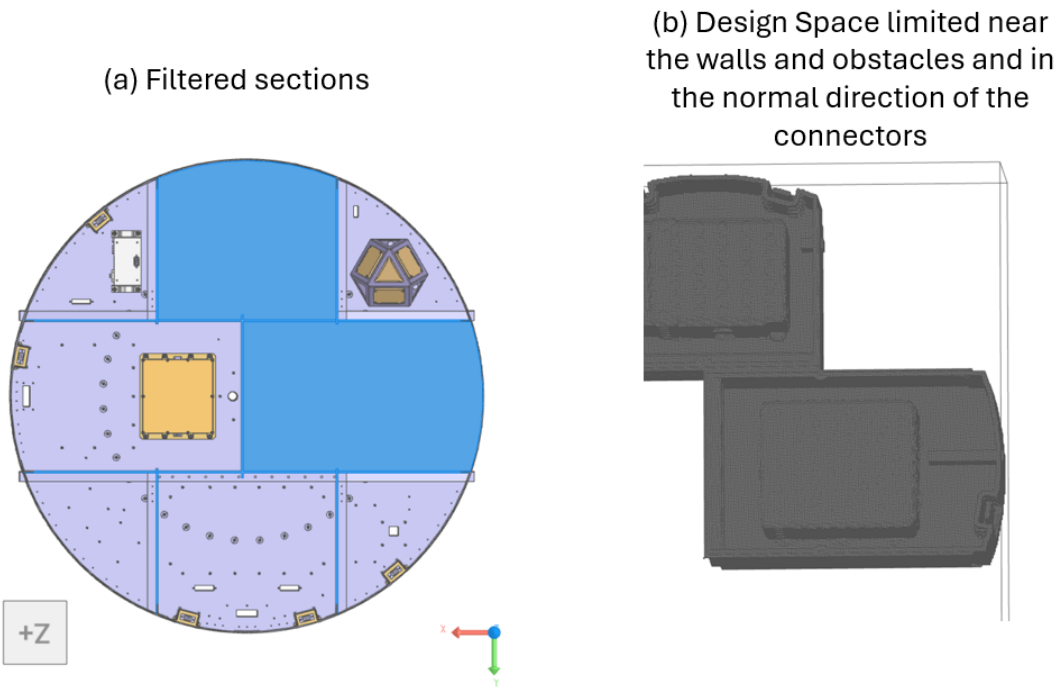


Pre-Processing IV: Section Filtering and Local DS Definition

A new coarse DS is first created based on the structural segmentation, and an initial rough path-finding is executed within it to determine which DS sections are relevant for this iteration. The sections intersected by the resulting path are selected, as shown in Figure 53-a, limiting the routing domain to the regions directly related to the cable trajectory.

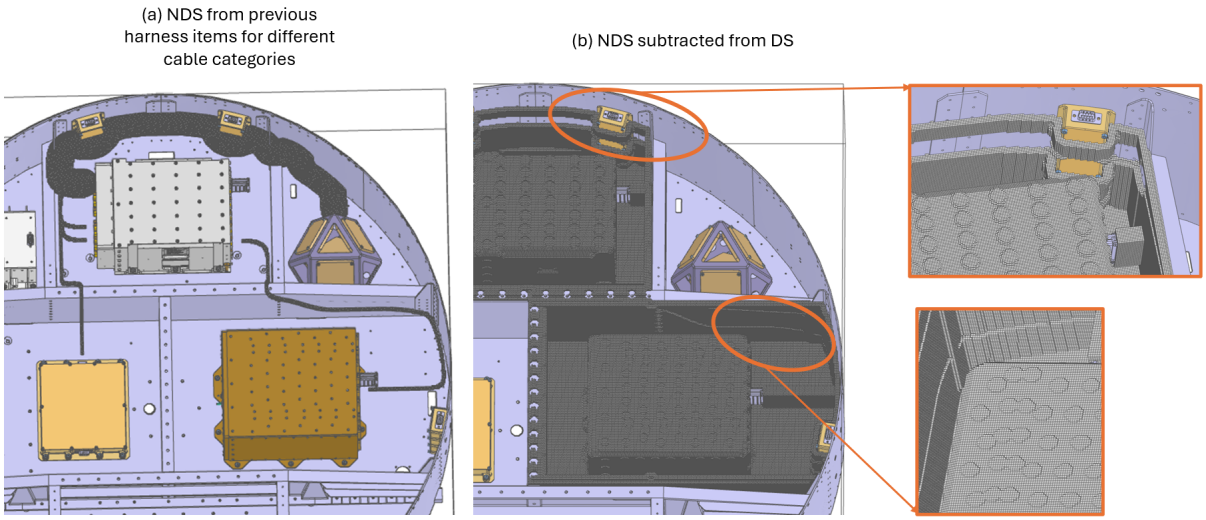
Next, a finer DS is generated around the selected sections. The DS is constrained near walls and components and extended only in the normal direction of the connectors to maintain alignment with their interfaces (Figure 53-b). Subsequently, obstacles such as screws and fasteners are removed from the DS to avoid unnecessary collision detection.

Figure 53: Harness item with single cable j=2 – Pre-processing IV: Selected sections.



Finally, the regions corresponding to previously routed cables are subtracted from the DS, taking into account the minimum distance requirements for cables of different categories (Figure 54). This step ensures that the new routing iteration respects both geometric clearance and inter-cable separation constraints.

Figure 54: Harness item with single cable j=2 – DS considering different cable categories.



Routing and Reconstruction

The A* algorithm is applied to the refined DS to compute the voxel-based route between the defined path-finding points. Figure 55 shows the resulting voxel path, which avoids both structural and previously defined NDS regions.

After the voxel path is generated, it is converted into a polyline representation and joined with the short initial connector lines, ensuring a continuous and physically consistent cable geometry (Figure 56).

Figure 55: Harness item with single cable j=2 – Path Finder: Generated voxel path.

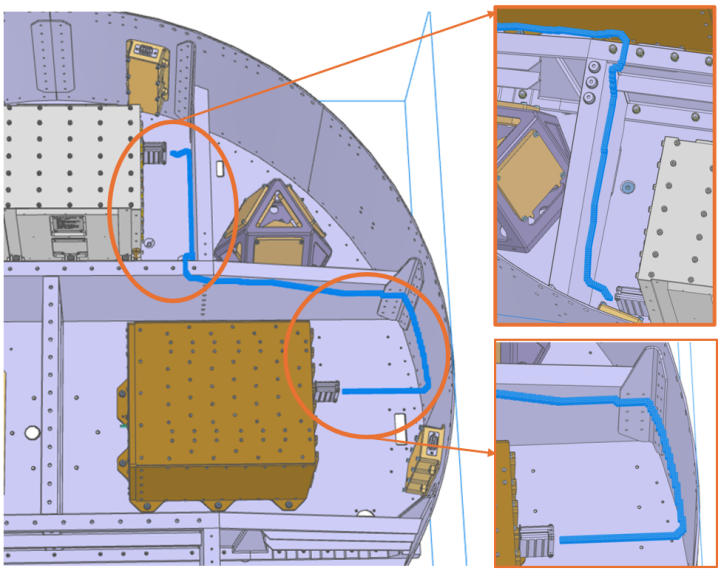
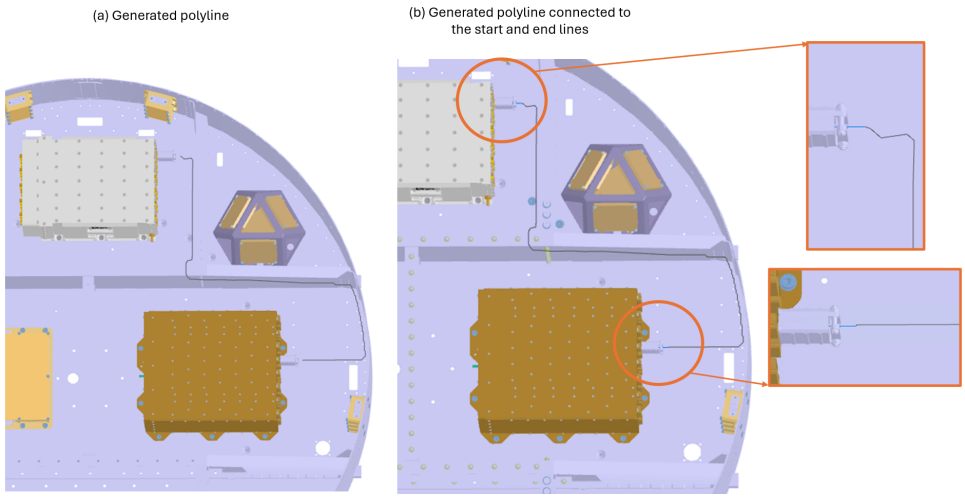


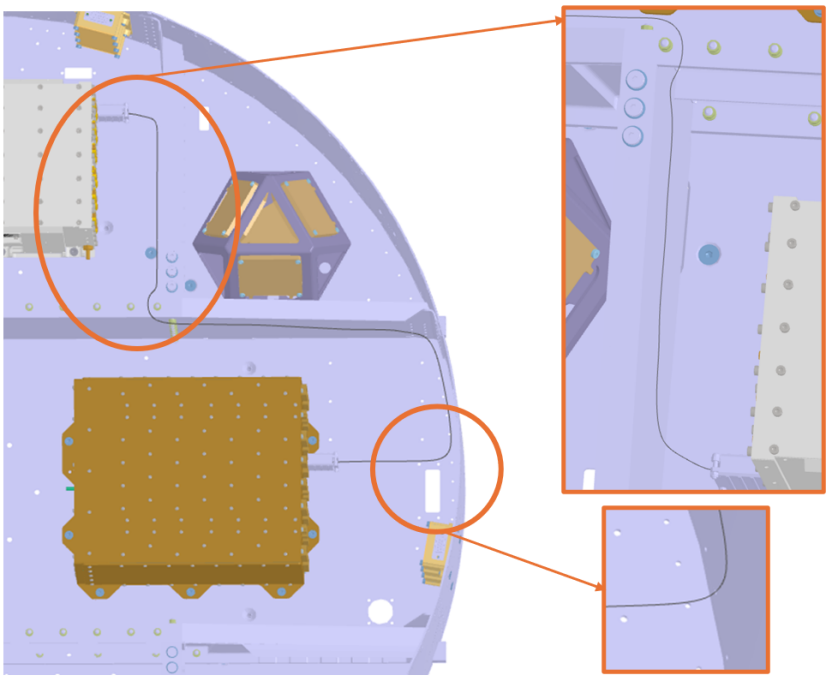
Figure 56: Harness item with single cable j=2 – Generated polyline path.



The reconstructed cable is then simulated in Raphos Physics using the same goals and solver configuration as in the previous iteration. The simulation refines the geometry by applying weighted goals, including curvature smoothing. The bending radius is again derived in Chapter 5, and the solver tries to maintain this constraint while balancing other geometric and physical objectives. Finally, the final reconstructed result is presented in Figure 57.

Overall, the results obtained from Routing Loops I and II demonstrate the capability of the

Figure 57: Harness item with single cable j=2 – Reconstructed polyline.



workflow to handle both bundled and single-cable harness configurations within the same satellite environment.

In Loop I, the workflow was applied to a multi-connector harness, showing how bundle formation and curvature reconstruction can be automatically derived from geometric and physical constraints. The single iteration of this loop validated the overall logic of the routing and reconstruction stages.

Loop II extended this approach to individual cables, where two sequential iterations illustrated the adaptive refinement of the DS and NDS after each routing step.

Throughout all loops, the workflow successfully adjusted the DS to the relevant structural regions, subtracted previously routed paths, and maintained inter-cable separation based on category-specific distance requirements. Although the reconstruction stage only attempts—rather than guarantees—to meet the required bending radius, the generated results exhibit geometrically coherent and physically plausible routings aligned with the connectors and the spacecraft structure.

Together, these outcomes confirm the flexibility and scalability of the method for automated, multi-stage cable routing, and demonstrate how the same routing logic can be consistently reused across different harness configurations, enabling scalable automation while preserving constraint awareness across multiple routing iterations.

6.5 Post-Processing

After the routing and reconstruction of all harness items, a post-processing stage was performed to evaluate the compliance of the generated geometries with the predefined design constraints. This stage is divided into two parts: (1) *checking*, which involves verifying whether the reconstructed geometries fulfill the required physical and geometric conditions; and (2) *report generation*, which summarizes the evaluated data for documentation and traceability.

Constraint Checking

The verification procedure automatically inspects the reconstructed paths and evaluates them against the main constraints defined in the workflow: minimum bending radius, intersection with NDS, distance to cables of different categories, and mounting interval compliance. Each constraint is assessed pointwise along the reconstructed geometry, and any violation is logged for further analysis.

This explicit separation between constraint-aware routing and constraint verification reflects a deliberate design choice of the workflow: constraints are embedded geometrically through DS shaping, category-aware subtraction, and physics-based reconstruction rather than enforced as hard optimization criteria during path-finding. This strategy enables efficient routing while preserving full traceability of constraint compliance in post-processing. In particular, the mounting interval constraint is not explicitly enforced during the routing or reconstruction steps, as mentioned in the Chapter 4, but is evaluated exclusively during post-processing as part of the verification procedure.

Table 6 summarizes the results of the constraint verification for all harness items. In this table, the number of points not fulfilling each constraint is reported, along with the corresponding harness reference. For the minimum bending radius constraint, four points were identified as not fulfilled in harness EC-RYSAT-07-03-008. Similarly, five intersections with the NDS were detected for the same harness item, primarily in regions with high curvature transitions.

No violations were found concerning the distance to cables of different categories, indicating that the applied subtraction of previously routed paths and category-based clearance distances was effective. However, deviations were observed in the mounting interval verification for multiple harnesses, where four fixation intervals exceeded the defined maximum spacing. These results are detailed in Table 6. In addition, the workflow stores the exact spatial locations of all detected violations. These points are subsequently used to generate the visual diagnostic images presented in the reporting stage, where each type of violation is highlighted along the final reconstructed route.

Table 6: Contraints checked for generated routes.

Constraint	**Number of points not fulfilled**	**Harness item**
Bend radius	4	EC-RYSAT-07-03-008
Intersection with NDS	5	EC-RYSAT-07-03-008
Distance to different cable categories among different harness items*	0	—
Mounting interval	4	EC-RYSAT-07-03-008, EC-RYSAT-07-03-000, EC-RYSAT-07-03-001

*As mentioned in Chapter 4, this is counted for different cables among different harness items and not cables in the same harness item.

These post-processing results demonstrate that while the workflow successfully generated geometrically consistent routings and respected inter-cable clearances, minor deviations in bending radius, intersection and mounting intervals still occurred. Such deviations are expected due to the simplified geometric relaxation model used in the Raphos Physics solver and the absence of active constraint enforcement during the voxel-based routing phase.

The next step presents the reporting process, where these results are structured and exported for engineering documentation and further analysis.

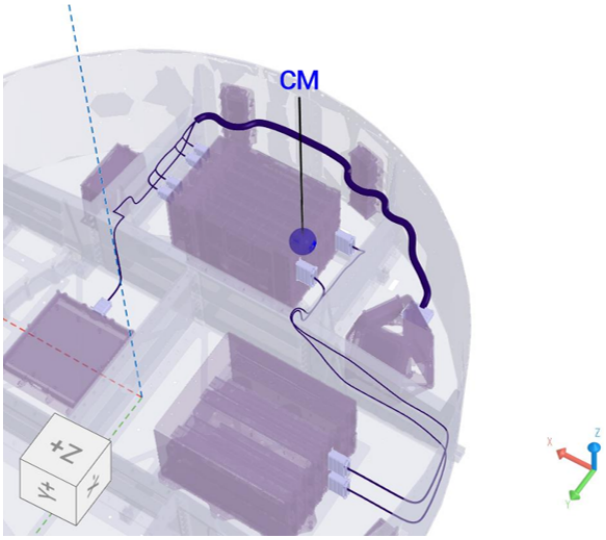
Report Generation

Following the constraint verification, the workflow compiles a visual and numerical report that consolidates all relevant information for traceability and validation purposes. This reporting stage allows engineers to assess the geometric integrity of the generated harness, identify potential violations, and locate them directly within the 3D model. A complete version of the generated report is provided in **Appendix C**.

In its structure, the report first presents a general overview of the full harness assembly, including the calculated CM, total routing length, and total mass of the generated solution. After this overview, the document is organized by harness item, and for each harness item the content is further subdivided by constraint type (bending radius, intersection, category distance, and mounting interval). Within each constraint section, the results are also broken down by electrical connection, enabling a clear and traceable evaluation of constraint fulfillment on a per-connection basis.

Figure 58 illustrates the final reconstructed routing of the Eu:CROPIS configuration used in this work, together with the calculated CM position of (-96.6, -323.6, 107.3) mm.

Figure 58: Final routing.



The reporting interface highlights specific constraint violations through color-coded visualization. Points where the minimum bending radius was not fulfilled presented in Table 6 are shown in yellow (Figure 59), while intersections with the NDS are indicated in red (Figure 60).

Additionally, the mounting interval verification identifies feasible attachment points in green and excessive spacing points in red (Figure 61). Since the constraint related to the distance between different cable categories was fully satisfied, no critical points appear in the corresponding report image. In this case the report simply displays the geometry without highlighted violations.

Figure 59: Final routing: Points where the bend radius was not fulfilled in yellow.

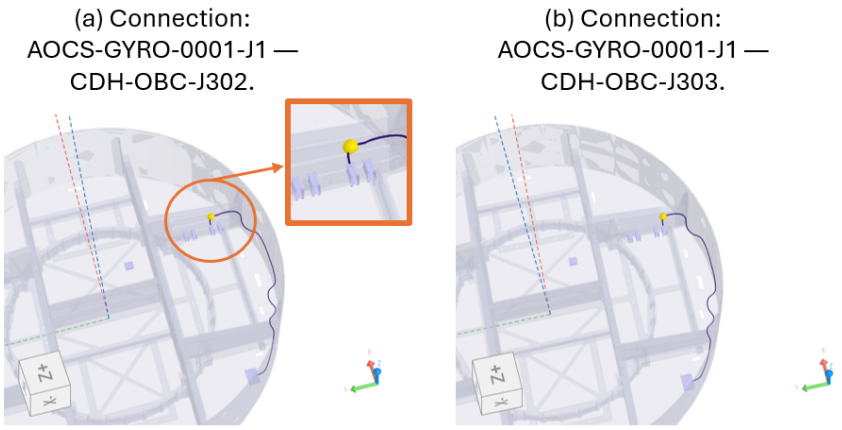


Figure 60: Final routing: Points where there are intersection to the NDS (in red).

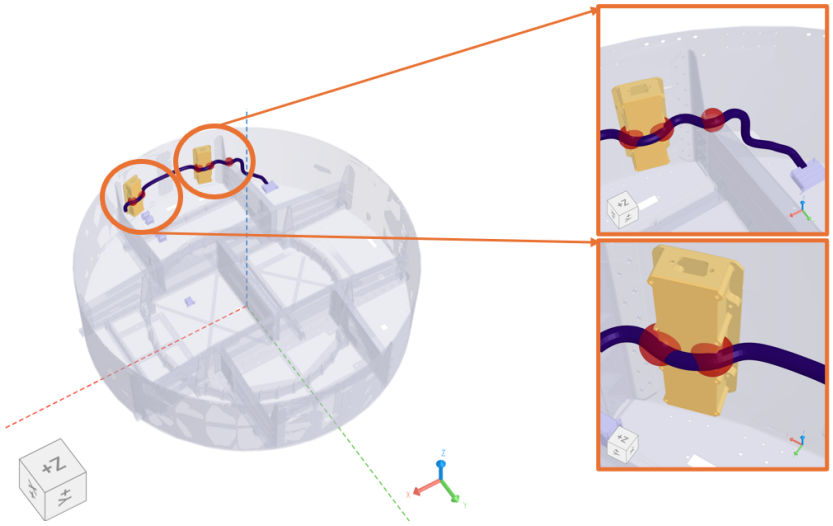
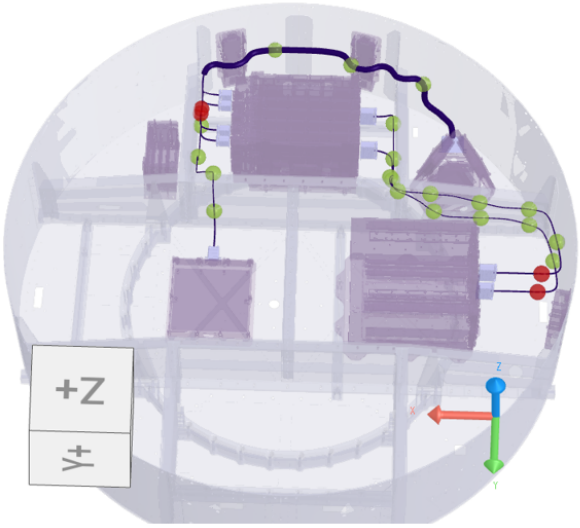


Figure 61: Final routing: Possible mounting points in green and distant points in red.



Overall, the reporting results demonstrate that the workflow successfully documents the routing performance and enables visual verification of all key design constraints. Although a few local deviations from the minimum bending radius and mounting interval were observed, these variations remain within acceptable limits for early-stage harness design.

Moreover, intersections with the NDS were also identified. Such deviations can be attributed to the relaxation effects of the Raphos Physics solver, which approximates cable deformation during reconstruction and may slightly adjust the geometry beyond the strict voxel-based constraint boundaries. This behavior aligns with the previously discussed limitations (Section 4.3.4), emphasizing the trade-off between computational efficiency and high-fidelity physical modeling in the current implementation.

6.6 Comparison Between the Original Harness and the Generated Routing

To assess the performance and reliability of the developed workflow, the final automatically generated harness routing was compared with the original Eu:CROPIS harness configuration. This comparison focuses on geometric accuracy, constraint compliance, mass-related implications, nd routing time, as these aspects are most relevant for satellite system integration and cost efficiency. The comparison also helps to evaluate the ability of the proposed workflow to replicate realistic routing behavior under automated conditions.

Overall Routing Comparison

For a consistent and traceable comparison, the original Eu:CROPIS routing was not only inspected visually but also reprocessed through the same reporting workflow used for the automatically generated harness. The original paths were imported using the corresponding `.json` data (connectors, diameters, categories and density), allowing the constraint-checking and mass-calculation procedures to be applied identically to both datasets. This ensures that any differences arise from routing geometry and not from differences in the evaluation process.

Figure 62 shows both the automatically generated and the original harness routings. Although both connect the same subsystems, the spatial paths differ considerably, particularly in their routing strategy around the structural compartments. The automatically generated harness follows a smoother and more regular trajectory along the available mounting surfaces, whereas the original design adopts a more compact layout with sharper directional changes and shorter inter-component distances. These distinctions reflect the methodological difference between automated routing—which prioritizes clearance, mounting surfaces, and global geometric consistency—and traditional manual routing, where designers exploit local geometric intuition and may implement longer bends or closer component passings to minimize path length.

Figure 62: Comparison of both harness routing.

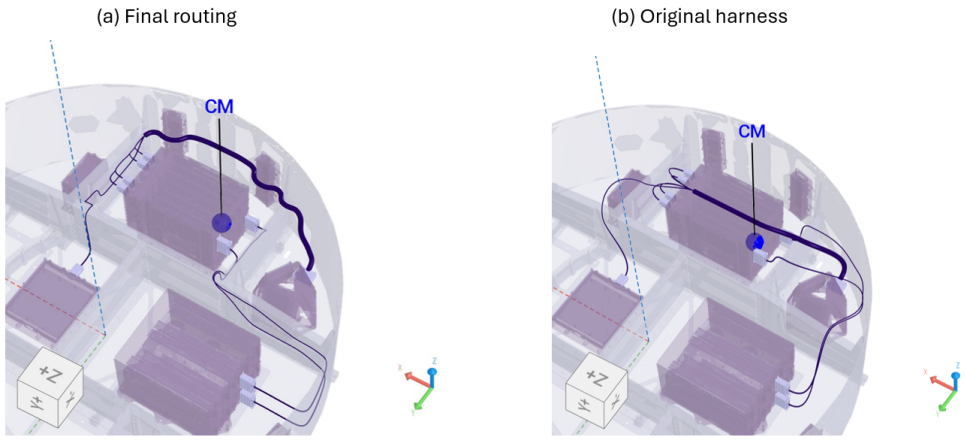


Table 7: Length and Mass Comparison of the Routings.

	Generated routing	Original method
Length [m]	5.9	4.7
Mass [kg]	0.21	0.17

The total cable length of the generated routing is approximately 5.9 m, compared to 4.7 m in the original configuration (Table 7), where the reported value represents the sum of the individual

lengths of all cables belonging to the harness item. This corresponds to a length increase of about 25.5%, arising primarily from two factors: (1) the voxel-based path-finding algorithm, which favors conservative obstacle avoidance and geometric regularity over strict shortest-path optimization, and (2) the restriction of the DS to regions located near walls and components, enforcing mountability and physical accessibility rather than allowing cables to traverse open central volumes.

Despite the increased length, the automated routing remains fully reasonable for early design phases, offering a clear and repeatable structure that aligns with standard spacecraft integration practices.

The difference in total length directly influences the overall harness mass, a relevant parameter for both system-level cost estimation and launch weight assessment. Since mass is proportional to cable length when identical materials and diameters are used, the generated routing exhibits a corresponding mass increase. Importantly, in this thesis the mass of the bundled harness item was computed based on the sum of the masses of its individual cables and not on the external bundle diameter, in order to ensure consistency with material density definitions. This same rule was applied to both the generated and the original routings, producing a fair comparison.

The resulting mass difference is approximately 23.5%, similar in scale to the length increase, and is expected given the routing characteristics described above. At this stage of development, this trade-off is acceptable, since the automated workflow significantly reduces manual design effort, increases consistency, and fully documents geometric constraint compliance. Future refinement of DS definition could further reduce this discrepancy while preserving manufacturability and design robustness.

In addition to geometric and mass-related differences, a key advantage of the automated workflow lies in its substantial reduction of design time. Table 8 contrasts the execution times of the automated process with the estimated effort required for a comparable manual routing task. While the original Eu:CROPIS harness was produced through traditional manual CAD modelling, the automated workflow completes all stages from geometric extraction to final reporting in approximately 1 hour and 11 minutes, compared to an estimated 28 hours of manual effort.

The most significant savings occur during the routing phase, where the automated A* + Raphos-based approach completes all path-finding and reconstruction steps in under 41 minutes, whereas manual routing requires iterative inspection of clearances, adjustment of bends, and re-evaluation of intermediate paths, amounting to more than 18 hours for the same harness items. Pre-processing and post-processing are similarly accelerated: geometric extraction, DS generation, and constraint checking are executed programmatically, removing the need for time-consuming manual inspection.

It should be noted that a significant portion of the manual design effort consists of interpretative tasks, such as assessing spacecraft geometry, anticipating routing impacts, and adapting cable routes after changes to nearby harness items. In the original workflow, the introduction of a new harness typically triggers substantial rework, as existing routes must be reassessed and manually adjusted to accommodate the change.

These efforts are inherently reduced when relying on a fully parametric workflow. Because the automated approach integrates previously routed cables and their categories directly into the DS, the addition of new harnesses is handled through automatic propagation of downstream effects, minimizing rework and improving overall consistency across the harness. Even when multiple design iterations are considered, the total effort remains significantly lower: five iterations at approximately 1 h 11 min each amount to just under 6 hours of total design time, compared to approximately 28 hours required with the original manual process

Overall, although the automated routing introduces a moderate increase in cable length and mass, these are offset by a dramatic reduction in design time and an improvement in repeatability. This illustrates the potential of automated routing tools not only to accelerate early design phases but also to maintain consistency across iterative spacecraft development cycles.

Table 8: Simulation time for the main stages of the workflow.

	Simulation time	**Time for original method**
Pre-Processing	13 min 45s	4 h 40 min
Routing	40 min 40 s	18 h 40 min
Post-Processing	16 min 53s	4 h 40 min
Total	1 h 11 min 18s	28 h*

*This accounts to the cumulative rework time required when adding a new harness. These efforts are reduced by using a parametric workflow.

A further aspect of the evaluation concerns the compliance of each routing with the geometric and physical constraints defined in the workflow. Tables 6 and 9 summarize, respectively, the constraint-checking results for the automatically generated routing and for the original Eu:CROPIS configuration (reprocessed through the same verification procedure for consistency). This comparison allows assessing how the automated workflow performs relative to the original design in terms of minimum bending radius, intersections with the NDS, cable segregation, and mounting interval requirements.

Table 9: Contraints checked for the original harness.

Constraint	**Number of points not fulfilled**	**Harness item**
Bend radius	6	EC-RYSAT-07-03-008
Intersection with NDS	1	EC-RYSAT-07-03-008
Distance to different cable categories among different harness items*	0	—
Mounting interval	14	EC-RYSAT-07-03-008, EC-RYSAT-07-03-000, EC-RYSAT-07-03-001

*As mentioned in Chapter 4, this is counted for different cables among different harness items and not cables in the same harness item.

The following subsections analyze each constraint in detail.

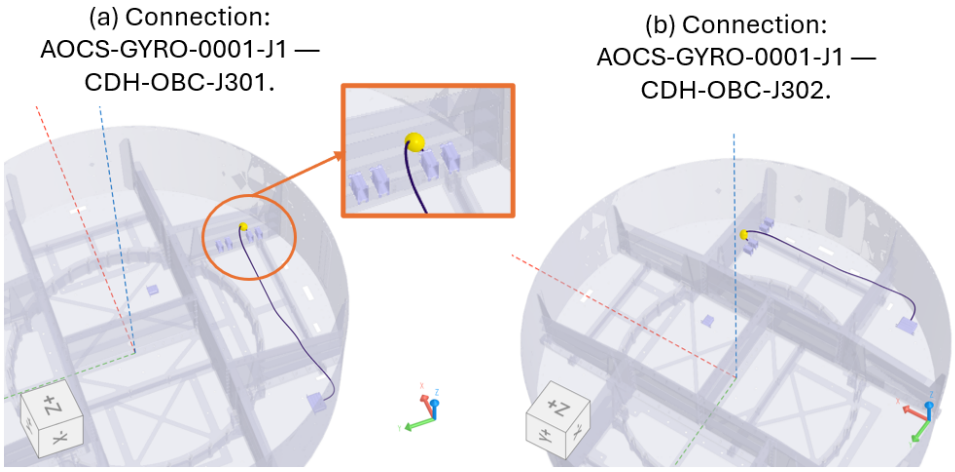
Bending Radius Compliance

The evaluation of the bending radius constraint is presented in Figures 59 (generated routing) and 63 (original routing). In both cases, localized violations are observed near connector interfaces and tight corner regions.

Quantitatively, the automatically generated harness exhibits **4 points** of noncompliance (Table 6), whereas the original routing shows **6 points** (Table 9) of violation in the same harness item (EC-RYSAT-07-03-008).

The improved behavior of the generated harness results from the reconstruction stage, which applies spring–particle relaxation with a predefined bending stiffness target. In this setup, the solver aims to reach a curvature corresponding to a user-defined radius larger than the minimum allowed value rather than enforcing a strict

Figure 63: Original routing: points where the bend radius was not fulfilled.



lower limit. Consequently, the method promotes smooth curvature transitions while reducing, but not entirely eliminating, local violations of the bending radius constraint.

Intersection with the NDS

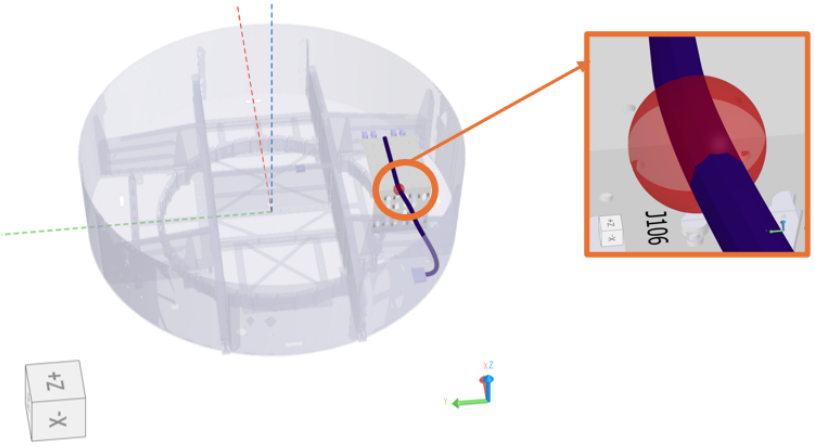
The intersection analysis (Figures 60 and 64) shows that the generated routing has **5 intersection points** (Table 6) with NDS regions, while the original harness presents only **1 intersection point** (Table 9) for the harness item EC-RYSAT-07-03-008. In the generated case, all intersection points occur within the bundled harness region, where the larger bending radius increases the likelihood of local surface penetration during the relaxation process. These minor intersections arise from the geometric adjustment performed by the solver as it accommodates the higher curvature constraint, occasionally extending slightly beyond the voxel-based collision boundaries.

Another contributing factor is related to the simplification applied in Path Finder II: the bundle region was not re-voxelized using the actual bundle diameter. To reduce computational time, the algorithm reused the voxel path initially generated for the individual cable, without updating the occupied space to reflect the larger cross-section of the bundle. Reconstructing this region with a bundle-specific DS would improve accuracy but would also require noticeably longer computation times, as the solver would need to regenerate and evaluate a higher-resolution voxel field. As a result, the simplified approach can lead to local intersections with nearby structures.

It should be noted that the intersection detection itself is performed using a voxel-based method, independent of the routing method. Consequently, the number of detected intersections is directly influenced by the voxel resolution employed during the checking phase. These deviations are on the order of the voxel resolution used for intersection checking and therefore represent small geometric offsets rather than significant penetrations, specially for the original harness. The observed deviations are on the order of the voxel size and therefore represent small geometric offsets rather than significant penetrations, particularly for the original harness. A finer voxel grid would reduce apparent penetration depth and margin uncertainty but at the cost of increased computational effort, whereas coarser resolutions may overestimate both intersections and clearance violations.

Although more frequent than in the original routing, these intersections are limited in magnitude and occur in non-critical areas. Overall, the observed deviations are consistent with the path finding limitation and relaxation effects discussed in Section 4.3.4.

Figure 64: Original routing: point where there was intersection with the NDS.

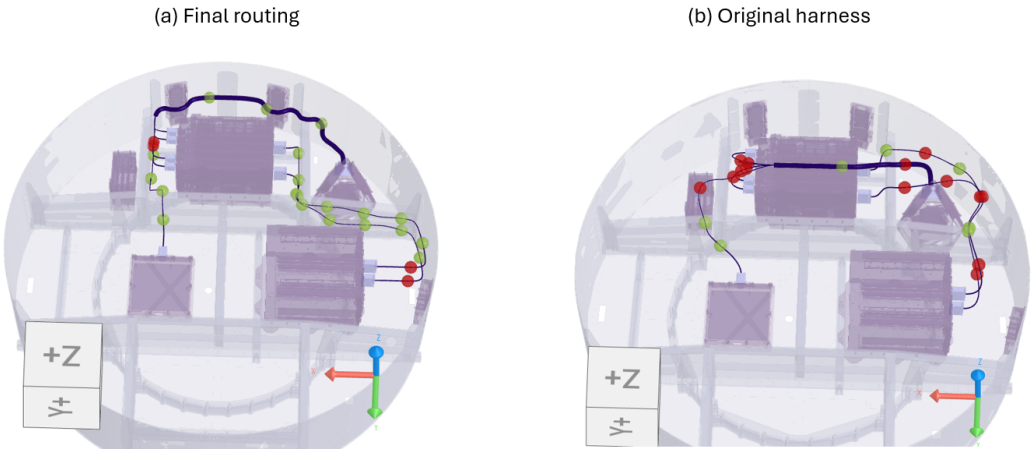


Mounting Interval Analysis

Figure 65 compares the mounting verification between the original and the generated routings. For the automatically generated harness, **4 points** (Table 6) exceeded the maximum allowable mounting interval, compared to a total of **14 points** (Table 9) in the original design (7 in EC-RYSAT-07-03-008, 4 in EC-RYSAT-07-03-000, and 3 in EC-RYSAT-07-03-001). Both configurations generally fulfill the required attachment spacing, but the automated approach provides a more uniform distribution and fewer critical gaps between fixation points.

This highlights that the automated constraint checking was not used as an active routing constraint, but rather as a post-processing evaluation tool. The improved fixture placement consistency therefore results from the DS configuration near walls and components, which naturally guided the routing toward regions favorable for mounting.

Figure 65: Mounting for both harness routing.



Distance Between Cable Categories

Both the original and the generated routings fulfill the required distance constraints between different cable categories among different harness items (Table 6 and 9). In the automated workflow, this compliance results from the subtraction of NDS volumes corresponding to previously routed cables and the reconstruction, whereas the original design achieved separation manually through engineering judgment. No violations were identified in either configuration, confirming the consistency of the implemented clearance logic.

The overall constraint comparison indicates that the original harness exhibits a slightly higher number of violations related to the minimum bending radius (six points, compared to four in the generated routing) and a considerably larger number of mounting-interval deviations (fourteen points, versus four in the generated case). This reflects the more compact and angular nature of the manually designed harness, where engineers often exploit tighter routing choices that may relax strict geometric constraints to minimize cable length and simplify integration.

Conversely, the automatically generated routing detected more intersections with the NDS (five points, compared to one in the original). As discussed earlier, this behavior is influenced by the conservative DS limitations imposed near walls for mountability and by the fact that, in the second routing stage, the bundle region is constructed using the cable path rather than recomputing a full bundle-aware voxelized path. Despite this, both routings fulfill the required distance constraints between different cable categories.

This numerical comparison establishes a balanced view: while the automated workflow improves regularity and reduces certain types of violations (notably bending radius and mounting spacing), it introduces a small number of local intersections that can be mitigated in future iterations through refined DS handling and improved treatment of bundle regions.

A qualitative inspection of the mounting regions reveals that the generated routing tends to align better with accessible structural surfaces, providing smoother attachment transitions and fewer abrupt curvature changes near screw bosses or panel interfaces. While the original harness follows more compact but irregular trajectories, the automated approach produces a more uniform spatial distribution. These findings support the applicability of the workflow to early-stage harness definition, where robust geometric behavior at mounting regions is prioritized over minimal routing length.

In summary, the comparison demonstrates that the developed workflow by combining graph-based path-finding with physics-based reconstruction achieves routing results comparable to the manually designed reference harness while improving constraint fulfillment and design consistency and significantly reducing the overall engineering effort and routing time. Although longer and heavier, the automatically generated routing satisfies the primary geometric and physical requirements to a large extent, while reducing the total routing time from several working days to approximately one hour, validating the methodology as a viable approach for automated harness design in spacecraft systems.

These results directly answer the secondary research question regarding efficiency: the automated workflow reduces routing time by more than one order of magnitude while maintaining structured constraint evaluation. This demonstrates that automation not only accelerates the design process but also improves repeatability and reduces dependency on individual engineering experience during early design phases.

7 Outlook: Optimization of Goal Weights

Although the optimization of goal weights was not originally part of the methodology described in Chapter 4, a preliminary investigation was performed to evaluate its potential for improving the reconstruction stage.

In this outlook, optimization was performed using the Optimization node of the Synera environment. The optimizer supports continuous, discrete, and mixed-integer problems, and allows single- and multi-objective formulations with inequality constraints. Internally, it combines an ACO strategy with a backtracking line-search method. This hybrid approach enables efficient exploration of weight combinations while progressively converging toward solutions that improve the selected objective responses. In this thesis, these objective responses are derived directly from the reconstruction behavior of the Raphos Physics solver, which is governed by weighted simulation goals.

Accordingly, the Raphos Physics solver relies on a set of weighted goals to guide the spring–particle simulation toward a physically plausible cable shape. In the context of this exploratory analysis, only a subset of these goals was considered for optimization: *Bending Rod*, *Point Voxel Collision*, *Direction*, and *Line Spring*.

These goals were selected because they directly influence the two constraint responses of interest during reconstruction: bending radius compliance and collision avoidance with the NDS. Other goals used in the full reconstruction setup, such as anchoring and bundle cohesion, were kept fixed to preserve topological consistency and avoid altering boundary conditions during the optimization study. By limiting the optimization to this subset, the analysis focuses on the trade-off between geometric smoothness, curvature control, and collision handling, without introducing additional degrees of freedom unrelated to the evaluated constraints.

In this analysis, the weights of these goals were automatically adjusted based on the constraint responses obtained from successive simulations.

Figure 66 shows the evolution of the weights for the harness with bundle (EC-RYSAT-07-03-008). The corresponding responses, presented in Figure 67, were evaluated with the objective of minimizing the number of points where the minimum bend radius was not fulfilled and the number of intersections with the NDS. The results indicate a gradual reduction of both metrics over successive simulations, suggesting that adaptive adjustment of goal weights can improve reconstruction quality by balancing constraint satisfaction with geometric smoothness.

For the single-cable harnesses (Figures 68 and 69), the results show stable behavior with no violations detected, demonstrating that the solver achieves convergence under weight variations. Since the single-cable cases exhibited stable convergence and no constraint violations across all tested configurations, additional weight-evolution plots were omitted to avoid redundancy.

This optimization procedure provides valuable insight for future development of the workflow. Automating the weight adjustment could enable adaptive reconstruction, where constraint satisfaction guides the solver toward more physically consistent geometries without manual tuning. Such an approach would extend the workflow from deterministic routing to a more feedback-driven optimization framework, strengthening the integration between geometric reasoning and physics-based simulation discussed throughout this thesis.

Figure 66: Optimization for weight goals: values of the weights – Harness with bundle EC-RYSAT-07-03-008.

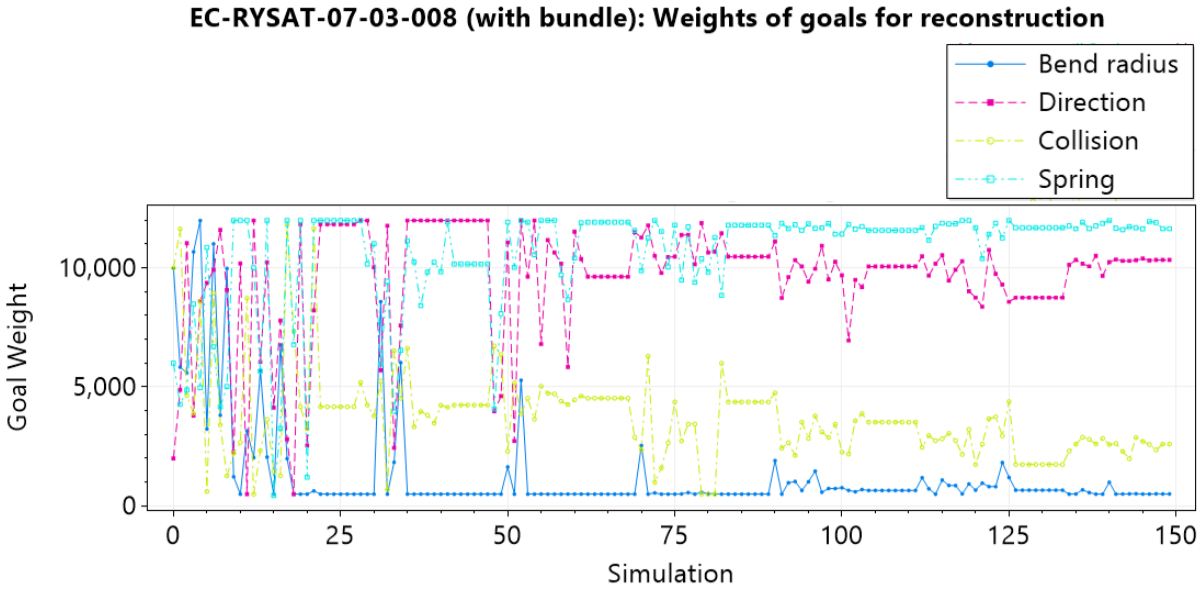


Figure 67: Optimization for weight goals: values of the responses – Harness with bundle EC-RYSAT-07-03-008.

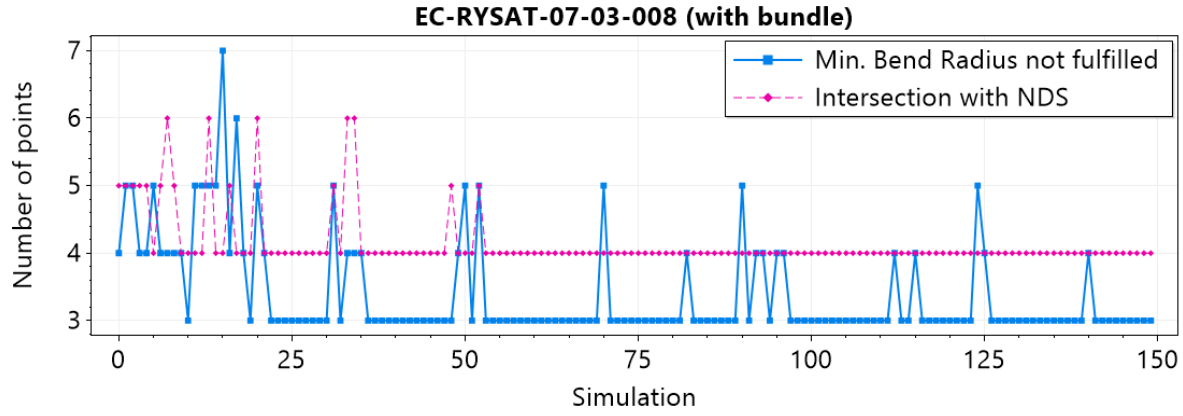


Figure 68: Optimization for weight goals: values of the responses – Harness with single cable EC-RYSAT-07-03-000.

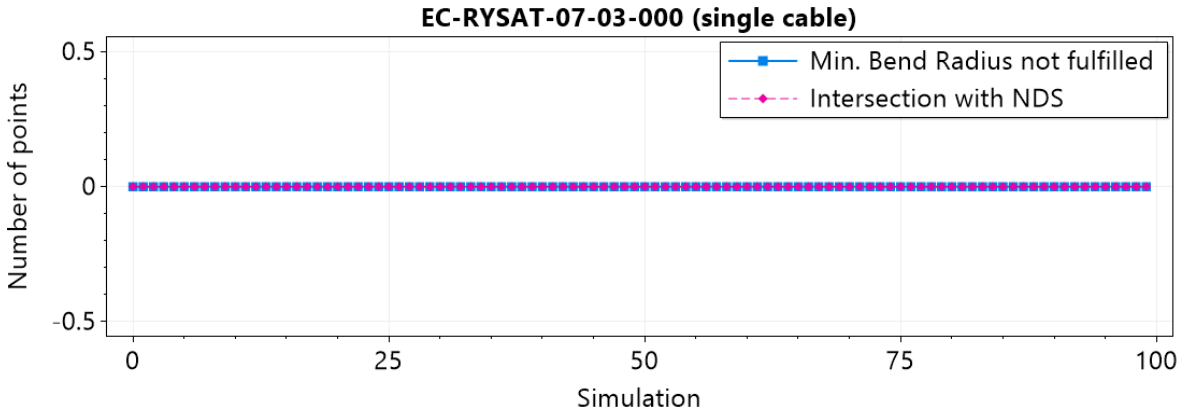
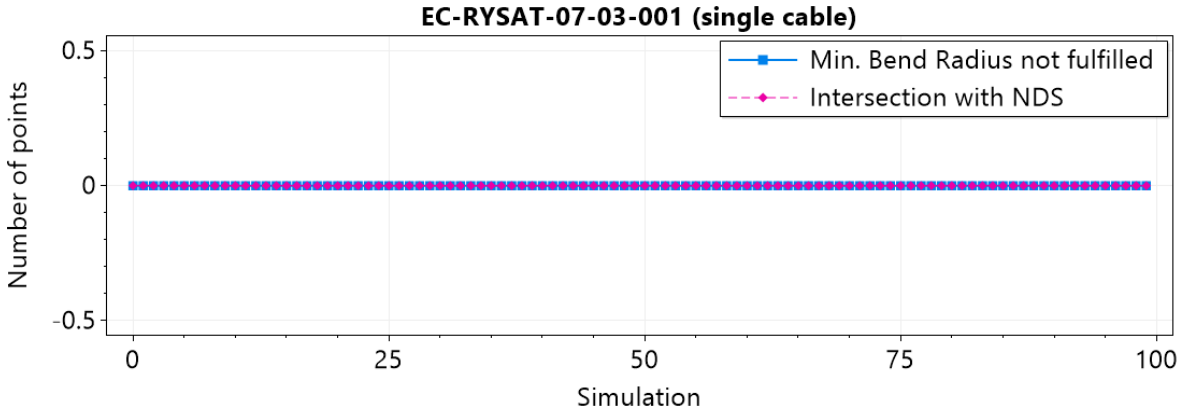


Figure 69: Optimization for weight goals: values of the responses – Harness with single cable EC-RYSAT-07-03-001.



8 Conclusion

This thesis addressed the CHRP in the context of satellite design, where increasing system complexity, limited installation space, and strict safety and integration constraints pose significant challenges for manual harness layout. The primary objective was to investigate whether an automated, algorithm-based routing workflow can support early-stage harness design while respecting geometric, functional, and category-based constraints commonly encountered in aerospace applications.

With this purpose, a voxel-based routing methodology was developed and implemented, combining graph-based path finding with sequential DS updates and physics-based reconstruction. The proposed workflow incorporates connector orientations, bending radius constraints, clearance rules between cable categories, and the handling of bundled harness segments.

By iteratively subtracting occupied regions from the DS based on the routed cable diameter and the minimum required distance between different cable categories, the approach ensures that subsequently routed harness items respect previously placed cables, thereby reflecting realistic spatial competition within confined satellite structures. Moreover, the DS is deliberately constrained to regions close to structural walls and components, promoting routings that are compatible with mounting surfaces. Mounting interval compliance is then explicitly evaluated in a post-processing step, allowing verification of adequate fixation spacing along the routed harness. In addition to geometric feasibility, the workflow demonstrates that a largely automated routing process can be achieved within a CAD-based environment, reducing the need for manual intervention during early-stage harness definition.

The results demonstrate that the proposed method is capable of generating feasible and constraint-compliant routing paths for representative harness configurations. The generated solutions show consistent adherence to defined clearance rules, avoidance of prohibited regions, and controlled path curvature near interfaces. The examples and the application to the Eu:CROPIS spacecraft further illustrate how the sequential routing strategy and bundle handling logic influence the final harness topology, providing transparency into the behavior of the algorithm. This application confirms that the proposed workflow can be successfully used on a real spacecraft configuration, thereby fulfilling the objective of validating the methodology using the Eu:CROPIS mission as a representative case study.

From a methodological perspective, this work shows that discrete, voxel-based path finding in combination with a reconstruction stage based on a spring–particle system logic can serve as a robust foundation for automated harness routing in complex three-dimensional environments. The modular structure of the workflow allows individual steps—such as DS generation, constraint handling, and routing—to be adapted or extended without altering the overall logic. While the presented results are not intended as a full industrial validation, they demonstrate the feasibility and consistency of the approach and highlight its potential as a decision-support tool during early design phases.

Furthermore, the combination of graph-based path finding with physics-based reconstruction improves routing quality by ensuring geometric consistency and constraint compliance, while simultaneously increasing efficiency through automation and repeatability compared to manual routing practices. By automating path generation, constraint handling, and verification, the workflow demonstrates the potential to significantly reduce routing design time while maintaining compliance with space system standards, making it particularly suitable for iterative early-phase spacecraft development.

At the same time, several limitations remain. The routing results depend on voxel resolution and heuristic parameters, and the current implementation focuses primarily on geometric feasibility

rather than optimality with respect to mass, manufacturability, or electromagnetic performance. Additionally, validation was limited to selected use case rather than large-scale industrial harness layouts. These aspects define clear boundaries for the conclusions that can be drawn and motivate further investigation.

Overall, this thesis contributes a structured and reproducible routing workflow that bridges the gap between manual harness layout and fully automated design. It provides a solid basis for future research and development aimed at integrating automated harness routing more deeply into satellite design and manufacturing processes.

9 Future Work

While the presented methodology demonstrates the feasibility of automated cable routing under realistic constraints, several avenues for future work can be identified to enhance robustness, efficiency, and industrial applicability.

The current routing strategy prioritizes feasibility and constraint satisfaction where the A* algorithm primarily aims to minimize routing length, while constraints such as bundling and bending radius are addressed in subsequent steps. Future work could extend the path finding toward multi-objective optimization, incorporating criteria such as total harness length, bundling, or routing smoothness based on a bend radius. The integration of optimization techniques could support trade-off analyses between competing design objectives. Additionally, adaptive heuristics or cost functions could be explored.

The geometric abstraction used in this work could be refined by incorporating more detailed physical properties of cables and bundles. For example, more advanced bundle modeling, including dynamic bundle formation and separation along the routing path, would further increase realism. In particular, the current bundle definition relies on selecting the cable with the shortest path as a reference and on a user-defined distance threshold to identify shared bundle regions. Therefore, future work could introduce a systematic evaluation of alternative bundle reference candidates and optimize the bundle definition based on the total length of the bundled harness item.

Another important extension concerns cost-oriented evaluation. Future developments could include the estimation of material costs, accounting for cable lengths, diameters, and connectors, as well as installation cost projections based on routing complexity, accessibility, and required mounting operations. Such analyses would further support early design trade-off studies.

Finally, further validation using different industrial case studies would be essential to assess the practical usefulness of the approach. Comparing more automatically generated harness layouts with manually designed reference solutions could provide insights into acceptance, efficiency gains, and potential limitations from an engineering perspective.

References

[1] E. Komninou, M. Vasile, and E. Minisci, "Optimal power harness routing for small-scale satellites," in *62nd International astronautical congress 2011*, 2011.

[2] V. Babuska, D. M. Coombs, J. C. Goodding, E. V. Ardelean, L. M. Robertson, and S. A. Lane, "Modeling and experimental validation of space structures with wiring harnesses," *Journal of Spacecraft and Rockets*, vol. 47, no. 6, pp. 1038–1052, 2010.

[3] J. M. Ritchie, G. Robinson, P. N. Day, R. G. Dewar, R. C. Sung, and J. E. Simmons, "Cable harness design, assembly and installation planning using immersive virtual reality," *Virtual Reality*, vol. 11, pp. 261–273, 2007.

[4] T. Karlsson, E. Åblad, T. Hermansson, J. S. Carlson, and G. Tenfält, "Automatic cable harness layout routing in a customizable 3d environment," *Computer-Aided Design*, vol. 169, p. 103671, 2024.

[5] Q. Jin, W. Sheng, C. Gao, and Y. Han, "A cable harness routing method based on the expected maximum crosstalk," *Ieee Transactions on Electromagnetic Compatibility*, vol. 64, no. 3, pp. 856–864, 2021.

[6] A. B. Conru, "A genetic approach to the cable harness routing problem," in *Proceedings of the first ieee conference on evolutionary computation. ieee world congress on computational intelligence.* Ieee, 1994, pp. 200–205.

[7] F. Ng, J. M. Ritchie, and J. Simmons, "The design and planning of cable harness assemblies," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 214, no. 10, pp. 881–890, 2000.

[8] S. GmbH. (2025) The tipping point in digitized product development. Synera GmbH. [Online]. Available: https://www.synera.io/connected-engineering

[9] M. Stanczak, "Optimisation of the waveguide routing for a telecommunication satellite," Ph.D. dissertation, ISAE-Institut Supérieur de l'Aéronautique et de l'Espace, 2022.

[10] T. Karlsson, "Optimization of cable harness routing," 2020.

[11] S. Bouaziz, M. Deuss, Y. Schwartzburg, T. Weise, and M. Pauly, "Shape-up: Shaping discrete geometry with projections," in *Computer Graphics Forum*, vol. 31, no. 5. Wiley Online Library, 2012, pp. 1657–1667.

[12] M. Malagoli and L. Cosquéric, "Space harness design optimization opportunities on ecss derating rules," in *Proc. Space Passive Component Days, ESTEC*, 2013, pp. 1–9.

[13] E. Keskin, O. Kara, S. Özbek, S. E. Aydin, and S. Demirdağ, "Electrical wiring design and development studies of a communication satellite," *Düzce Üniversitesi Teknik Bilimler Dergisi*, vol. 2, no. 1, pp. 1–12, 2024.

[14] D. Hambleton, P. Whitehead, J. Williams, and G. Jones, "Guidelines for spacecraft power and signal cabling," 1978.

[15] K. J. Yuan, "Design of a prototype machine to automate satellite wire harness assembly," 2020.

[16] K. Gwozdecky, "Methodologies for development of a modular wiring harness for use in small satellite constellations," 2020.

[17] R. Amini, G. Aalbers, R. Hamann, W. Jongkind, and P. G. Beethuizen, "New generations of spacecraft data handling systems: Less harness, more reliability," in *57th International Astronautical Congress*, 2006, pp. D1–4.

[18] W. Lei, F. Yu, H. Fengping, Z. Xuan, L. Jian, and L. Peng, "Application of lightweight wire in high orbit satellite," in *2021 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA)*. IEEE, 2021, pp. 251–255.

[19] M. F. Hasan, M. Atif, M. Atta-ur Rehman, and M. Saeed, "Development plan of harness model for small satellites," in *Conference: International Conference on Aerospace Science & Engineering (ICASE)-2016*, 2016.

[20] M. W. Stavnes and A. N. Hammoud, "A review of wiring system safety in space power systems," in *Intersociety Energy Conversion Engineering Conference*, no. NAS 1.26: 194439, 1993.

[21] E. DlJKSTRA, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 50, pp. 269–271, 1959.

[22] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.

[23] Y. Zhao, J. Liu, J. Ma, and L. Wu, "Multi-branch cable harness layout design based on genetic algorithm with probabilistic roadmap method," *Chinese Journal of Mechanical Engineering*, vol. 34, pp. 1–11, 2021.

[24] D. Zhang, Z. Liu, C. Zhou, and D. Zuo, "Multi-objective layout optimization of aircraft multi-branch cable harness based on mopso/d," in *2021 12th International Conference on Mechanical and Aerospace Engineering (ICMAE)*. IEEE, 2021, pp. 139–144.

[25] "Ecss-e-st-20c rev.2 (2022) – space engineering: Electrical and electronic," https://ecss.nl/wp-content/uploads/2022/04/ECSS-E-ST-20C-Rev.2(8April2022).pdf, 2022, european Cooperation for Space Standardization.

[26] "Ecss-q-st-20-30c dir.1 (2024) – space product assurance: Manufacturing and control of electrical harness," https://ecss.nl/wp-content/uploads/2024/09/ECSS-Q-ST-20-30C-DIR1(11September2024).pdf, 2024, european Cooperation for Space Standardization.

[27] "Ecss-q-st-70-61c – space product assurance: High reliability assembly for surface mount and through hole connections," https://ecss.nl/wp-content/uploads/2022/04/ECSS-Q-ST-70-61C(8April2022).pdf, 2022, european Cooperation for Space Standardization.

[28] "Ecss-e-20a – space engineering: Electrical and electronic," https://ecss.nl/wp-content/uploads/standards/ecss-e/ECSS-E-20A4Oct1999.pdf, 1999, european Cooperation for Space Standardization.

[29] "Ecss-e-st-20-07c rev.2 – space engineering: Electromagnetic compatibility," https://ecss.nl/wp-content/uploads/2022/01/ECSS-E-ST-20-07C-Rev.2(3January2022).pdf, 2022, european Cooperation for Space Standardization.

[30] K. Jasiński, "Altium – wiring for the final frontier: A guide to space-grade harness design," https://resources.altium.com/p/guide-to-space-grade-harness-design, 2025, european Cooperation for Space Standardization.

[31] "Ecss-q-st-70-18c – space product assurance: Preparation, assembly and mounting of rf coaxial cables," https://ecss.nl/wp-content/uploads/standards/ecss-q/ECSS-Q-ST-70-18C15November2008.pdf, 2008, european Cooperation for Space Standardization.

[32] "Nasa-std-8739.4a (30 june 2016) – nasa technical standard: Crimping, interconnecting cables, harnesses, and wiring," https://standards.nasa.gov/sites/default/files/standards/NASA/A/4/nasa-std-87394a_w_change_4_0.pdf, 2016, european Cooperation for Space Standardization.

[33] "Nasa sma newsletter – "updates to crimping, interconnecting cables, harnesses, and wiring" (jul 2016) – noting release of nasa-std-8739.4a," https://sma.nasa.gov/news/articles/newsitem/2016/08/25/updates-to-crimping-interconnecting-cables-harnesses-and-wiring-standard-published, 2016, european Cooperation for Space Standardization.

[34] T. Delovski, C. Düvel, F. Greif, A. Heidecker, S. Kottmeier, O. Mierheim, F. Nohka, and F. Orlowski-Feldhusen, "Eu: Cropis aiv program: Challenges and solutions for a spin-stabilized satellite containing biology," *International Journal of Aerospace Engineering*, vol. 2019, no. 1, p. 9190329, 2019.

[35] J. Hauslage, S. M. Strauch, O. Eßmann, F. W. Haag, P. Richter, J. Krüger, J. Stoltze, I. Becker, A. Nasir, G. Bornemann *et al.*, "Eu: Cropis–"euglena gracilis: combined regenerative organic-food production in space"-a space experiment testing biological life support systems under lunar and martian gravity," *Microgravity Science and Technology*, vol. 30, pp. 933–942, 2018.

[36] "Eu:cropis," https://www.dlr.de/en/rb/research-operation/missions/earth-observation-science/eu-cropis, 2018, dLR (German Aerospace Center).

[37] D. G. A. Center). (2025) Farewell to the eu:cropis mission. [Online]. Available: https://www.dlr.de/en/latest/news/2020/01/20200113_farewell-to-the-eucropis-mission

[38] C. Strowik, "Design, Manufacturing and Control of Harness," DLR (German Aerospace Center), Tech. Rep., 2022.

[39] S. S. . N. Neptune, "Path finding. category: Curve, create (software documentation). accessed via in-software help menu," 2025.

[40] Z. Lin, Z. Wang, T. Gong, Y. Ma, and W. Xie, "Voxel-based path planning for autonomous vehicles in parking lots," *ISPRS International Journal of Geo-Information*, vol. 14, no. 4, p. 147, 2025.

[41] K. Kim, Y. Yoon, B. C. Kim, J. Kim, S. Han, and S. Kwon, "Automatic cable routing based on improved pathfinding algorithm and b-spline optimization for collision avoidance," *Journal of Computational Design and Engineering*, vol. 11, no. 5, pp. 303–317, 2024.

[42] "Synera: Raphos physics," https://raphos.com/synera/raphos-physics/, 2025, raphos Ltd.

[43] "Marketplace: Add-in - raphos physics," https://portal.synera.io/marketplace/add-ins/42533b21-b141-4e16-94a2-46de117cd1cd, 2025, synera GmbH.

[44] S. S. . N. Neptune, "Raphos physics. category: Raphos physics, solver (software documentation). accessed via in-software help menu," 2025.

[45] "Marketplace: Add-in - office connector," https://portal.synera.io/marketplace/add-ins/ee205968-7231-4a80-a5b1-fa3f516f8ddb, 2025, synera GmbH.

[46] J. Blasco, C. Vernich, S. de la Rosa, M. Á. Esteban, and M. J. Méndez, "Optoelectronic link for optical satellite harnessing substitution in space communications," in *International Conference on Space Optics—ICSO 2008*, vol. 10566. SPIE, 2018, pp. 743–749.

[47] Dayashankara, Hariharan, Rao, Rajan, and Bokil, "Emc considerations in spacecraft harness design," in *1997 Proceedings of International Symposium on Electromagnetic Compatibility.* IEEE, 1997, pp. 239–242.

[48] W. Pemarathne and T. Fernando, "Wire and cable routings and harness designing systems with ai, a review," in *2016 IEEE international conference on information and automation for sustainability (ICIAfS).* IEEE, 2016, pp. 1–6.

[49] "Ecss-e-st-20-07c rev.1 (2012) – space engineering: Electromagnetic compatibility," https://ecss.nl/wp-content/uploads/standards/ecss-e/ECSS-E-ST-20-07C-Rev17February2012.pdf, 2012, european Cooperation for Space Standardization.

A Input: Harness Data – .json File

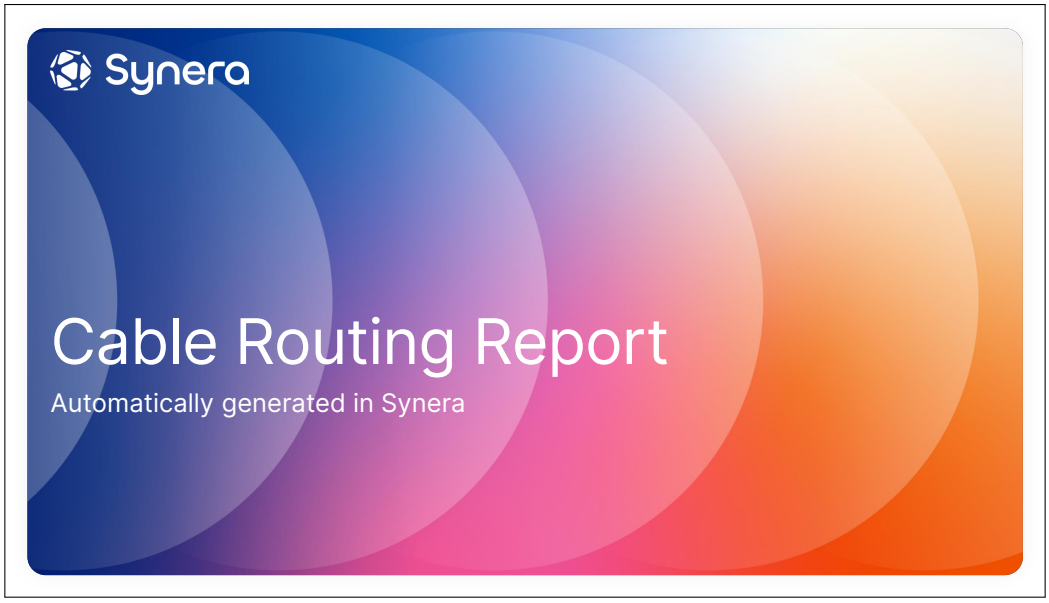

```
{
    "harnesses": [
        {
            "harness_name": "EC-RYSAT-07-03-000",
            "connectors": [
                {
                    "connector_name": "CDH-OBC-J101"
                },
                {
                    "connector_name": "EPS-PCDU-J122"
                },
                {
                    "priority": "None"
                }
            ],
            "connections": [
                {
                    "connector_1": "CDH-OBC-J101",
                    "connector_2": "EPS-PCDU-J122",
                    "diameter": 3,
                    "category": "2"
                }
            ]
        },
        {
            "harness_name": "EC-RYSAT-07-03-001",
            "connectors": [
                {
                    "connector_name": "CDH-OBC-J201"
                },
                {
                    "connector_name": "EPS-PCDU-J121"
                },
                {
                    "priority": "None"
                }
            ],
            "connections": [
                {
                    "connector_1": "CDH-OBC-J201",
                    "connector_2": "EPS-PCDU-J121",
                    "diameter": 3,
                    "category": "2"
                }
            ]
        },
        {
            "harness_name": "EC-RYSAT-07-03-008",
            "connectors": [
                {
                    "connector_name": "AOCS-GYRO-0001-J1"
                },
                {
                    "connector_name": "CDH-OBC-J300"
                },
                {
                    "connector_name": "CDH-OBC-J301"
                },
                {
                    "connector_name": "CDH-OBC-J302"
                },
                {
                    "connector_name": "CDH-OBC-J303"
                },
                {
                    "connector_name": "AOCS-GYDC-0001-J2"
                },
                {
                    "priority": "None"
                },
            ],
            "connections": [
                {
```

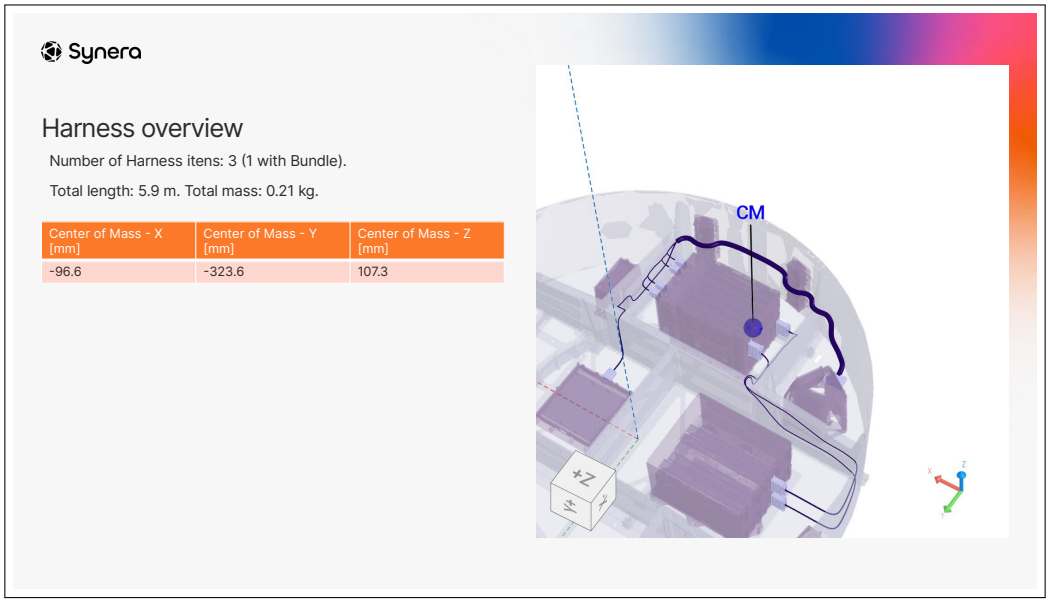
```
                        "connector_1": "AOCS-GYRO-0001-J1",
                        "connector_2": "AOCS-GYDC-0001-J2",
                        "diameter": 3,
                        "category": "1"
                    },
                    {
                        "connector_1": "AOCS-GYRO-0001-J1",
                        "connector_2": "CDH-OBC-J300",
                        "diameter": 2.2,
                        "category": "2"
                    },
                    {
                        "connector_1": "AOCS-GYRO-0001-J1",
                        "connector_2": "CDH-OBC-J301",
                        "diameter": 2.2,
                        "category": "2"
                    },
                    {
                        "connector_1": "AOCS-GYRO-0001-J1",
                        "connector_2": "CDH-OBC-J302",
                        "diameter": 2.2,
                        "category": "2"
                    },
                    {
                        "connector_1": "AOCS-GYRO-0001-J1",
                        "connector_2": "CDH-OBC-J303",
                        "diameter": 2.2,
                        "category": "2"
                    }
                ]
            }
        ]
    }
```

B Input: Components Data – .json File

```
{
  "Bus": [
    "EC-BSFA-03-01-000-31"
  ],
  "Components": [
    "EC-BSFA-03-07-000-31",
    "EC-RYSAT-06-01-000-31",
    "aa-PFM1-assembly",
    "EC-RYSAT-09-05-000-31",
    "EC-RYSAT-09-06-000-31_Assembly",
    "EC-RYSAT-14-01-000-31"
  ]
}
```

C Automatically Generated Report





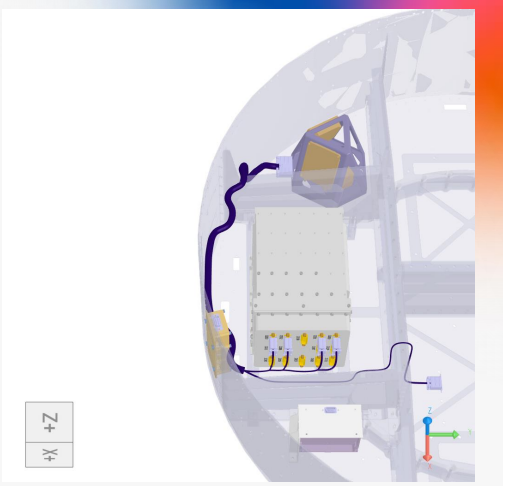
Synera

Harness overview for EC-RYSAT-07-03-008 (with bundle - X=55mm fulfilled.)

Min. bend radius: 15.9 mm.

Priority: None.

Connectors	Diameter [mm]	Category	Min. bend radis [mm]
AOCS-GYRO-0001-J1 — AOCS-GYDC-0001-J2	3	1	9
AOCS-GYRO-0001-J1 — CDH-OBC-J300	2.2	2	6.6
AOCS-GYRO-0001-J1 — CDH-OBC-J301	2.2	2	6.6
AOCS-GYRO-0001-J1 — CDH-OBC-J302	2.2	2	6.6
AOCS-GYRO-0001-J1 — CDH-OBC-J303	2.2	2	6.6



Synera

Harness Name: EC-RYSAT-07-03-008. Connection: Bundle

All points fulfilled the contraint for minimum bend radius 15.9.

Synera

Harness Name: EC-RYSAT-07-03-008. Connection: AOCS-GYRO-0001-J1 — AOCS-GYDC-0001-J2.

All points fulfilled the contraint for minimum bend radius 9.

Synera

Harness Name: EC-RYSAT-07-03-008. Connection: AOCS-GYRO-0001-J1 — CDH-OBC-J300.

All points fulfilled the contraint for minimum bend radius 6.6.

Synera

Harness Name: EC-RYSAT-07-03-008. Connection: AOCS-GYRO-0001-J1 — CDH-OBC-J301.

All points fulfilled the contraint for minimum bend radius 6.6.

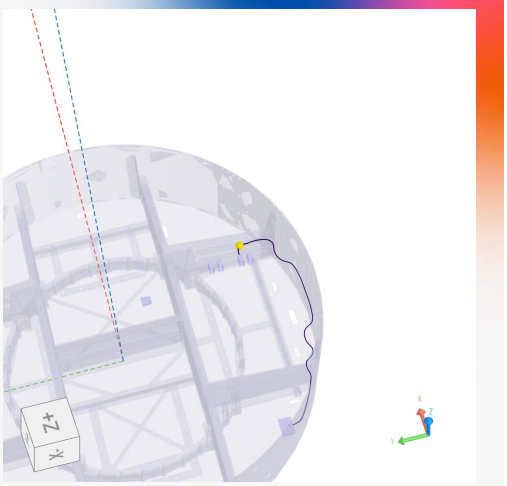
Synera

Harness Name: EC-RYSAT-07-03-008. Connection: AOCS-GYRO-0001-J1 — CDH-OBC-J302.

Points with constraints not met for minimum radius 6.6.

Points: (X = 177.357, Y = -363.352, Z = 129.196); -13%

(X = 176.433, Y = -361.482, Z = 129.191); -41.5%



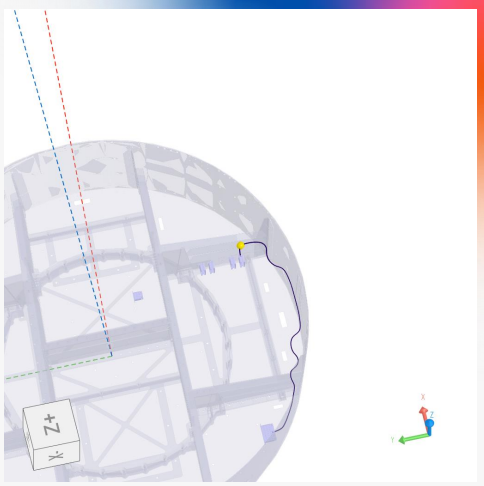
Synera

Harness Name: EC-RYSAT-07-03-008. Connection: AOCS-GYRO-0001-J1 — CDH-OBC-J303.

Points with constraints not met for minimum radius 6.6.

Points: (X = 177.705, Y = -384.867, Z = 129.193); -45.1%

(X = 176.751, Y = -383.526, Z = 129.189); -48.8%



Synera

Harness Name: EC-RYSAT-07-03-008. Connection: Bundle

Points intersecting the structure/components.

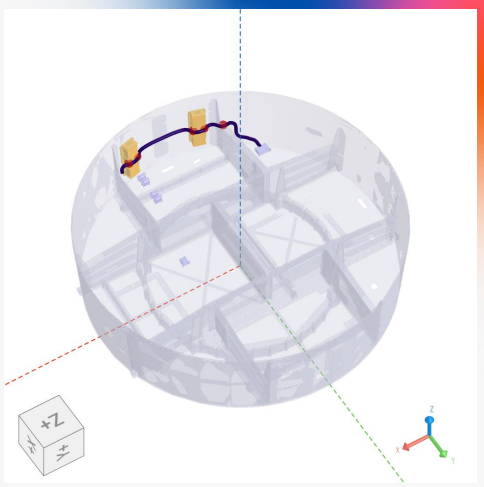
Points: Point(X = 106.05, Y = -460.9, Z = 129.05)

Point(X = -139.6, Y = -451.55, Z = 129.05)

Point(X = 140.9, Y = -449.85, Z = 129.05)

Point(X = -198.25, Y = -429.45, Z = 130.75)

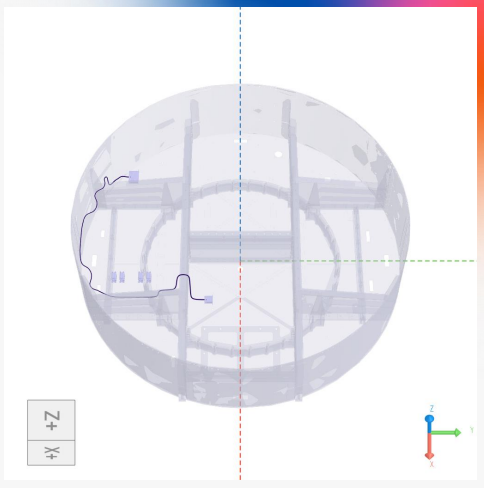
Point(X = -103.05, Y = -460.9, Z = 129.05)



Synera

Harness Name: EC-RYSAT-07-03-008. Connection: AOCS-GYRO-0001-J1 — AOCS-GYDC-0001-J2.

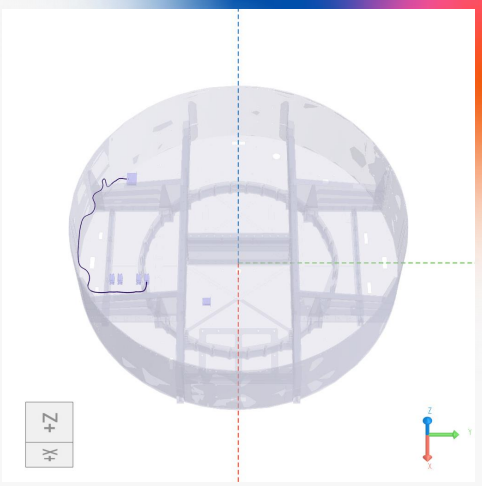
No intersection with the structure/components.



Synera

Harness Name: EC-RYSAT-07-03-008. Connection: AOCS-GYRO-0001-J1 — CDH-OBC-J300.

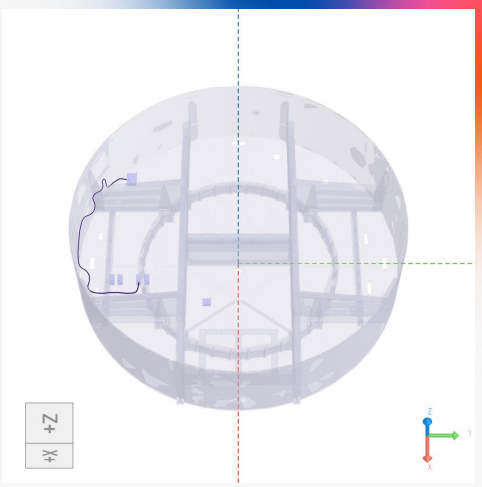
No intersection with the structure/components.



Synera

Harness Name: EC-RYSAT-07-03-008. Connection: AOCS-GYRO-0001-J1 — CDH-OBC-J301.

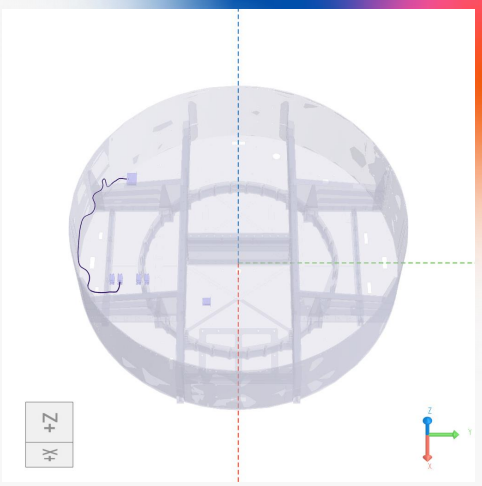
No intersection with the structure/components.



Synera

Harness Name: EC-RYSAT-07-03-008. Connection: AOCS-GYRO-0001-J1 — CDH-OBC-J302.

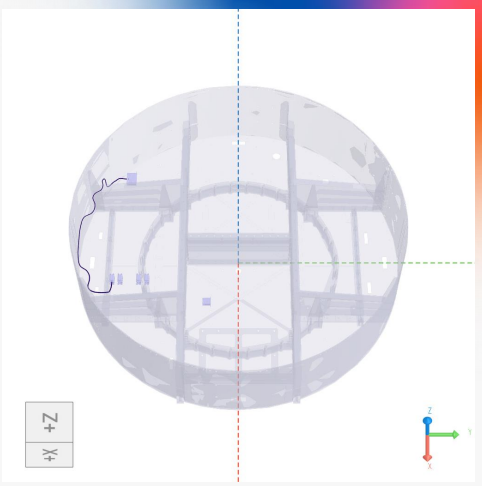
No intersection with the structure/components.



Synera

Harness Name: EC-RYSAT-07-03-008. Connection: AOCS-GYRO-0001-J1 — CDH-OBC-J303.

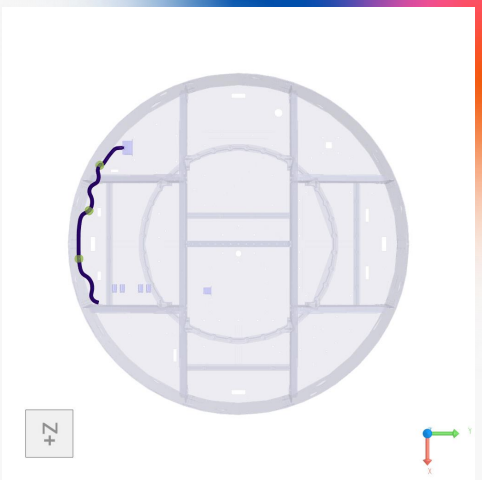
No intersection with the structure/components.



Synera

Harness Name: EC-RYSAT-07-03-008. Connection: Bundle

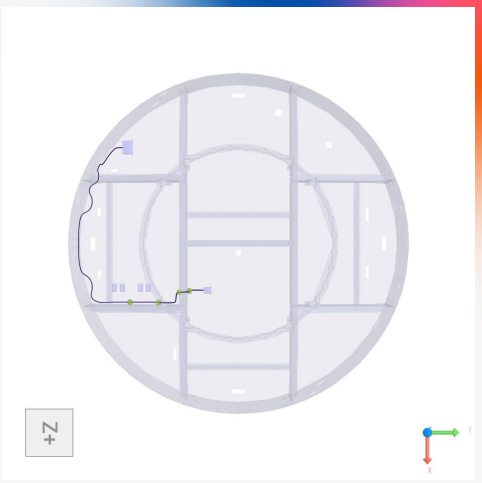
Mounting distance were fulfilled.



Synera

Harness Name: EC-RYSAT-07-03-008. Connection: AOCS-GYRO-0001-J1 — AOCS-GYDC-0001-J2.

Mounting distance were fulfilled.

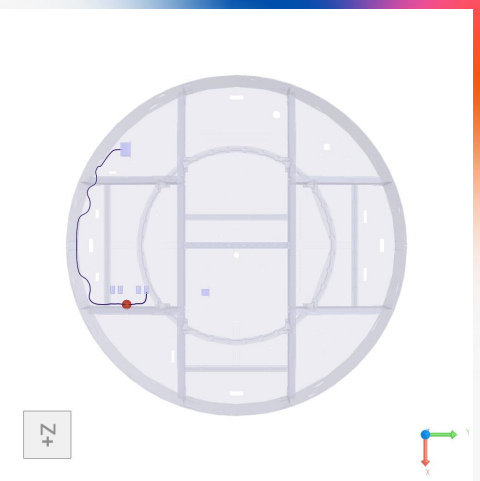


Synera

Harness Name: EC-RYSAT-07-03-008. Connection: AOCS-GYRO-0001-J1 — CDH-OBC-J300.

Points where the mounting distance are not fulfilled.

Points: Point(X = 179.682, Y = -333.841, Z = 129.2)

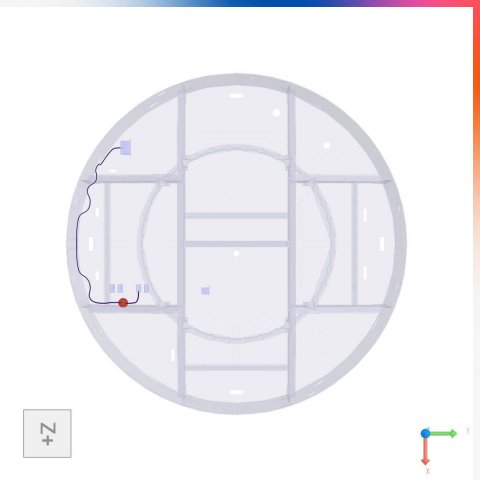


Synera

Harness Name: EC-RYSAT-07-03-008. Connection: AOCS-GYRO-0001-J1 — CDH-OBC-J301.

Points where the mounting distance are not fulfilled.

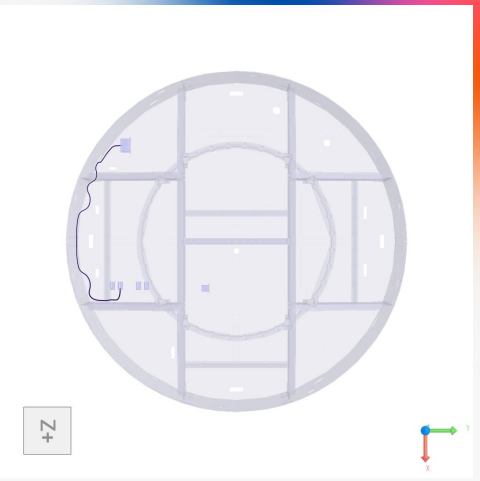
Points: Point(X = 178.836, Y = -344.656, Z = 129.2)



Synera

Harness Name: EC-RYSAT-07-03-008. Connection: AOCS-GYRO-0001-J1 — CDH-OBC-J302.

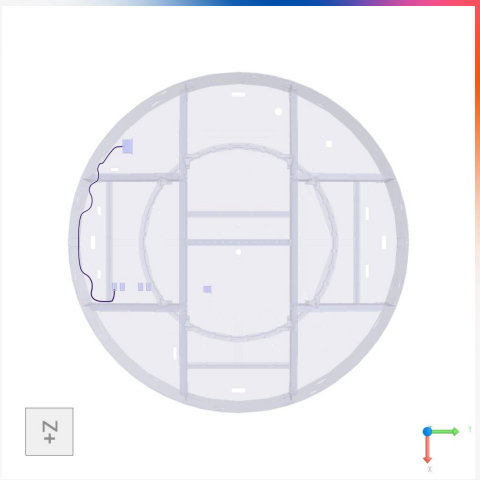
Mounting distance were fulfilled.



Synera

Harness Name: EC-RYSAT-07-03-008. Connection: AOCS-GYRO-0001-J1 — CDH-OBC-J303.

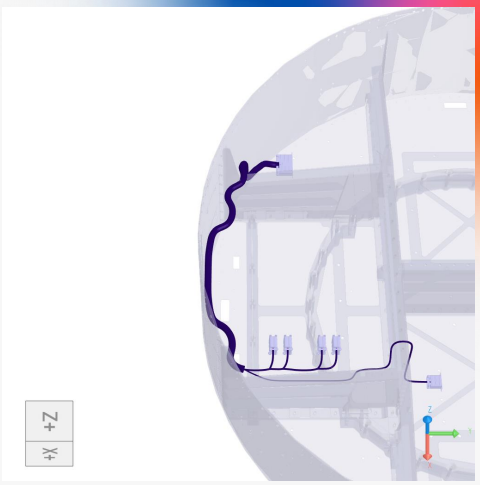
Mounting distance were fulfilled.



Synera

EC-RYSAT-07-03-008

Distance constraint for harnesses with different categories was fulfilled.



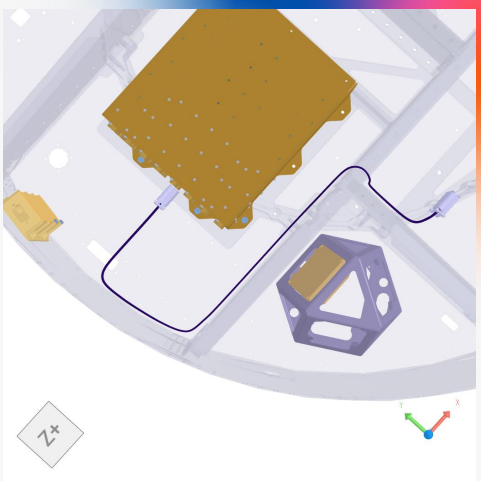
Synera

Harness overview for EC-RYSAT-07-03-000 (single cable)

Min. bend radius: 9 mm.

Priority: None.

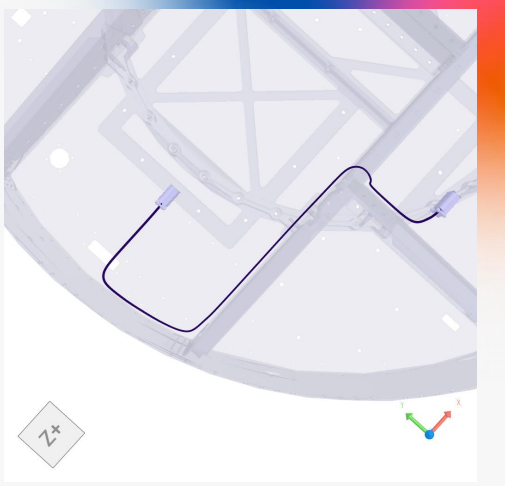
Connectors	Diameter [mm]	Category	Min. bend radis [mm]
CDH-OBC-J101 — EPS-PCDU-J122	3	2	9



Synera

Harness Name: EC-RYSAT-07-03-000. Connection: CDH-OBC-J101 — EPS-PCDU-J122.

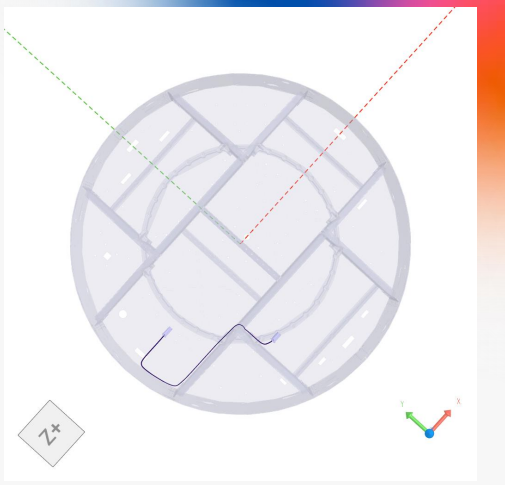
All points fulfilled the contraint for minimum bend radius 9.



Synera

Harness Name: EC-RYSAT-07-03-000. Connection: CDH-OBC-J101 — EPS-PCDU-J122.

No intersection with the structure/components.

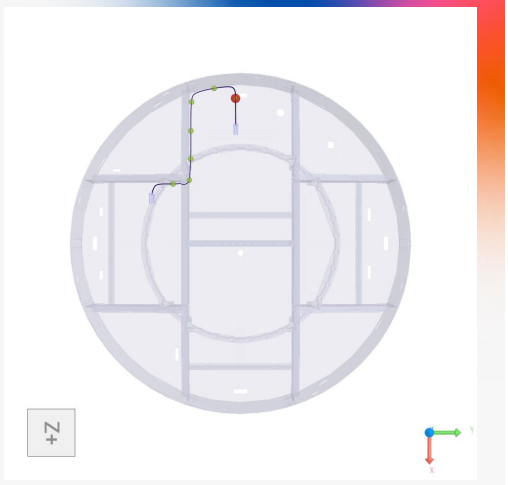


Synera

Harness Name: EC-RYSAT-07-03-000. Connection: CDH-OBC-J101 — EPS-PCDU-J122.

Points where the mounting distance are not fulfilled.

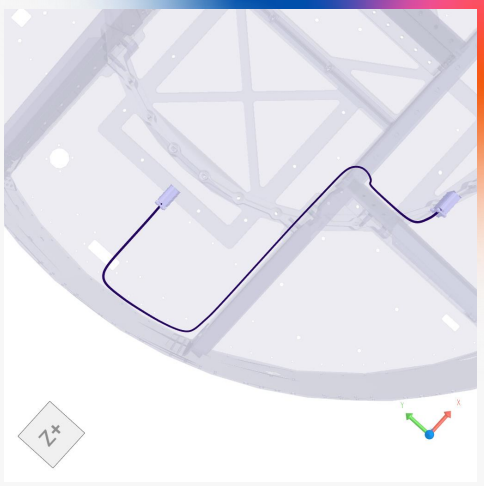
Points: Point(X = -451.322, Y = -15.3912, Z = 47.6)



Synera

EC-RYSAT-07-03-000

Distance constraint for harnesses with different categories was fulfilled.



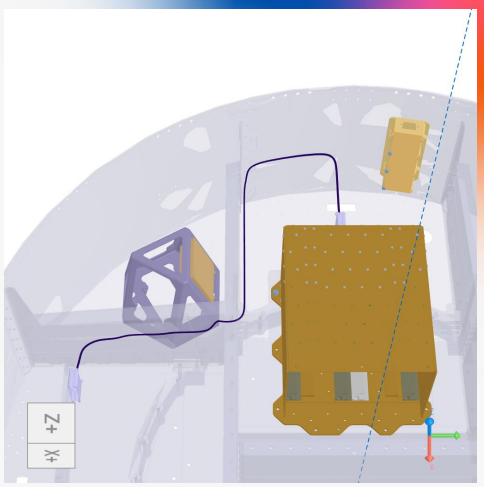
Synera

Harness overview for EC-RYSAT-07-03-001 (single cable)

Min. bend radius: 6.6 mm.

Priority: None.

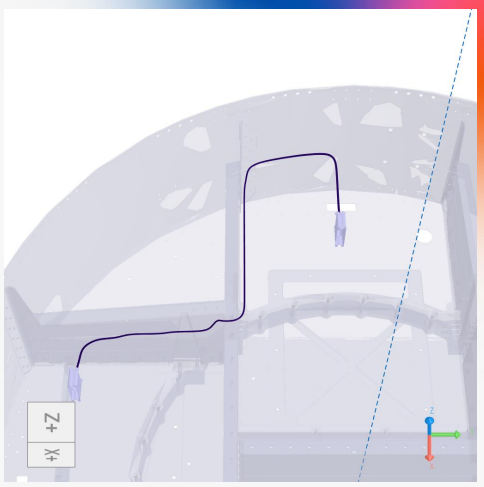
Connectors	Diameter [mm]	Category	Min. bend radis [mm]
CDH-OBC-J201 — EPS-PCDU-J121	3	2	9



Synera

Harness Name: EC-RYSAT-07-03-001. Connection: CDH-OBC-J201 — EPS-PCDU-J121.

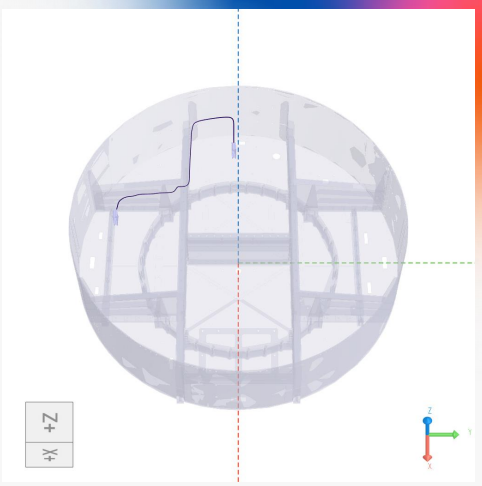
All points fulfilled the contraint for minimum bend radius 9.



Synera

Harness Name: EC-RYSAT-07-03-001. Connection: CDH-OBC-J201 — EPS-PCDU-J121.

No intersection with the structure/components.

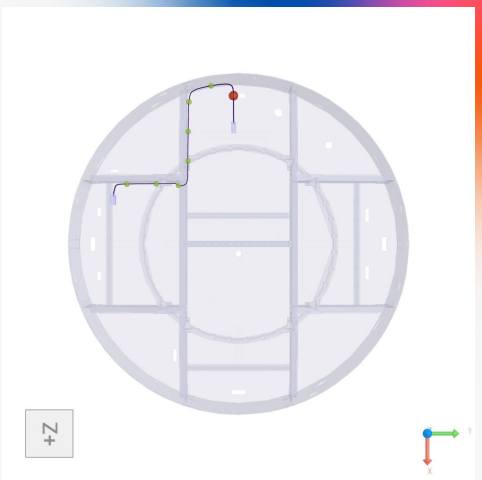


Synera

Harness Name: EC-RYSAT-07-03-001. Connection: CDH-OBC-J201 — EPS-PCDU-J121.

Points where the mounting distance are not fulfilled.

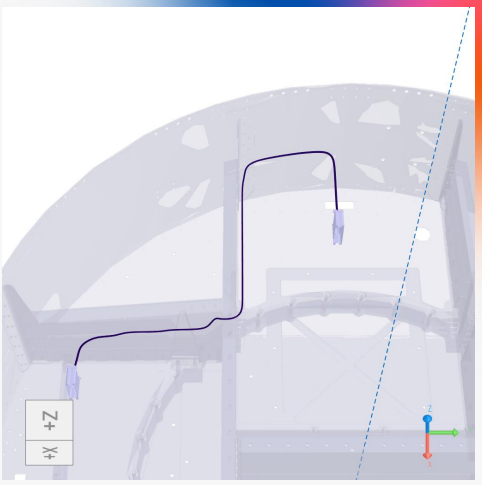
Points: Point(X = -452.014, Y = -15.3969, Z = 113.9)



Synera

EC-RYSAT-07-03-001

Distance constraint for harnesses with different categories was fulfilled.



Documentation of the Use of AI-based Applications and Tools

Universität Bremen

Dokumentation der Nutzung von KI-basierten Anwendungen und Werkzeugen

Documentation of the Use of AI-based Applications and Tools

	KI-basiertes Hilfsmittel AI-based Tool	Einsatzform Purpose	Betroffene Teile der Arbeit Aspect of the Work Affected	Beschreibung der Eingabe (Prompt) Prompt (Entry)	Bemerkung Comment
1	*ChatGPT*	*Thema eingrenzen* *Narrowing down a topic*	*Themenfindung* *Finding a topic*	*Wurde die Automatisierung der Kabelführung in Satelliten bereits untersucht?* *Has automation of cable routing in satellites been already studied?*	*Die Antwort hat mir geholfen, eine Motivation für meine Abschlussarbeit zu finden.* *The answer helped me find a motivation for my thesis.*
2	*ChatGPT*	*Thema eingrenzen* *Narrowing down a topic*	*Themenfindung* *Finding a topic*	*Was sind die relevantesten Forschungsergebnisse zum Thema Kabelführung für die Konstruktion von Kabelbäumen?* *What are the most relevant researches about cable routing for harness design?*	*Die Antwort half mir dabei, relevante Forschungsthemen für meine Abschlussarbeit zu finden.* *The answer helped me find relevant research topics to my thesis.*
3	*ChatGPT*	*Literaturrecherche* *Searching literature*	*Relevante Artikel finden* *Finding relevant papers*	*Was sind die neuesten Forschungsarbeiten zum Thema Kabelverlegung für die Konstruktion von Kabelbäumen für Satelliten?* *What are the latest research papers about cable routing for harness design for satellites?*	*Die Antwort hat mir geholfen, relevante Artikel für meine Abschlussarbeit zu finden.* *The answer helped me find relevant papers to my thesis.*
4	*ChatGPT*	*Literaturrecherche*	*Relevante Artikel finden*	*Welche ECSS- und NASA-Normen sind für die Kabelführung relevant?*	*Die Antwort half mir dabei, relevante technische Standards für*

		Searching literature	*Finding relevant technical documents*	*What are the relevant ECSS and NASA Standards in cable routing?*	*meine Abschlussarbeit zu finden.* *The answer helped me find relevant space technical standards to my thesis.*
5	*NotebookLM*	*Literaturüberblick zur Forschungsfrage* *Overview of the relevant literature*	*Literaturübersicht* *Literature review*	*Erstellen Sie einen Zeitplan und legen Sie die Hauptthemen der bereitgestellten Quellen fest:* *Prepare a timeline and main topics of the provided sources:*	*Das Ergebnis lieferte mir einen guten Zeitplan für meine Literaturrecherche.* *The result provided me with a good timeline of my literature research.*
6	*NotebookLM*	*Literaturüberblick zur Forschungsfrage* *Overview of the relevant literature*	*Literaturübersicht* *Literature review*	*Fassen Sie die in jeder Arbeit der angegebenen Quellen verwendete Hauptmethode zusammen:* *Summarize the main method used in each paper of the provided sources:*	*Das Ergebnis lieferte mir eine gute Zusammenfassung jeder Arbeit meiner Literaturrecherche.* *The result provided me with a good summary of each paper of my literature research.*
7	*NotebookLM*	*Literaturüberblick zur Forschungsfrage* *Overview of the relevant literature*	*Literaturübersicht* *Literature review*	*Was sind die Einschränkungen des genetischen Algorithmus und anderer Optimierungsalgorithmen für die Kabelführung, die in den bereitgestellten Quellen vorgestellt werden?* *What are the limitations of Genetic Algorithm and other optimization algorithms in cable routing presented in the provided sources?*	*Die Antwort half mir, die Grenzen anderer in der Literatur vorgestellter Methoden zu erkennen und zu verstehen.* *The answer helped me to identify and understand the limitations in other methodologies presented in the literature.*
8	*ChatGPT*	*Grammatik überprüfen* *Checking grammar*	*Textüberprüfung* *Text review*	*Überprüfen Sie den Text und suchen Sie nach grammatikalischen Fehlern.* *Review the text and look for grammar issues.*	*Die Antwort hat mir geholfen, Fehler in meiner Abschlussarbeit zu finden.* *The answer helped me to identify errors in my thesis.*