POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Model-based offline planning for dual-arm robotic manipulation of deformable linear objects

Tesi di Laurea Magistrale in
Automation and Control Engineering - Ingegneria dell'Automazione

Author: **Tommaso Dotti**

Student ID: 964654
Advisor: Prof. Paolo Rocco
Co-advisors: Andrea Monguzzi
Academic Year: 2021-22

Abstract

The increasing presence of automated systems in industrial frameworks makes the development of fully automatized processes more and more necessary, in order to improve cycle time of single task and optimize the global production. The manipulation of deformable linear objects (DLOs), such as wires, cables and tubes, represents the most difficult aspect to be automatized, due to the complex behaviours introduced by this kind of objects. For these reasons, the manipulation of DLOs, required in several industrial applications, is today entirely performed by human operators, involving alienating and time consuming operations.

This thesis proposes a model-based offline planning for a dual-arm robotic manipulation of a deformable linear object. The aim is to bring the DLO from an initial configuration, to a final one. A mass-spring dynamical model is exploited in order to perform optimization and simulation, and an additional Young's modulus estimation stage provides information related to the stiffness of the cable, managing to properly handle several deformable linear objects, with different mechanical properties.

Experiments are performed to validate the proposed method, obtaining good results that prove robustness of the planner to several DLOs brought in different final configuration. In addition a wire harness assembly operation is performed by exploiting the planning algorithm developed in this work.

Keywords: Model-based planning, Optimization, Dual-arm manipulation, Deformable linear objects, Robotics.

Abstract in lingua italiana

L'aumento della presenza di sistemi automatizzati nel settore industriale porta all'esigenza sempre più spinta di sviluppare processi totalmente automatici, in modo da diminuire i tempi di ciclo delle singole operazioni e di ottimizzare la produzione totale. La manipolazione di oggetti deformabili lineari (DLOs), come cavi e tubi, rappresenta l'aspetto più difficile in questo contesto, a causa delle complesse proprietà associate a questo tipo di oggetti. Per questo motivo, la manipolazione di DLO, necessaria in diverse realtà industriali, è al giorno d'oggi interamente eseguita da operatori, spezzando la catena di operazioni automatiche e introducendo tempi di attesa.
Questa tesi propone un metodo per pianificare la manipolazione di un oggetto deformabile lineare tramite un robot a due braccia. L'obiettivo è di portare un cavo da una configurazione iniziale ad una finale. Si utilizza un modello dinamico massa-molla integrato nella fase di ottimizzazione e simulazione, inoltre si implementa una fase di stima del modulo di Young, in modo da ottenere informazioni riguardo la rigidezza associata al cavo. In questo modo è possibile eseguire la pianificazione per vari DLO con diverse proprietà meccaniche.
Per convalidare la metodologia vengono eseguiti degli esperimenti finali , che mostrano buoni risultati e robustezza per diversi tipi di DLO portati in diverse configurazioni. Inoltre viene eseguita un'operazione di cablaggio, sfruttando l'algoritmo di pianificazione sviluppato in questo lavoro.

Parole chiave: Pianificazione , Ottimizzazione, Manipolazione dual-arm, Oggetto deformabile lineare, Robotica.

Ringraziamenti

Vorrei dedicare queste righe a tutti coloro che mi sono stati vicini in questo percorso di crescita personale e professionale.

Ringrazio i Professori Paolo Rocco e Andrea Zanchettin, che mi hanno permesso di far parte del loro gruppo di ricerca e lavorare nel MeRLIn lab.

Un ringraziamento speciale ad Andrea Monguzzi, che è stato capace di guidarmi quando ne avevo bisogno, sapendo dare spazio alla mia creatività e alle mie intuizioni.

Ringrazio i miei amici e compagni di questi cinque anni di Politecnico: Riccardo, Andrea, Gianluca, Roi e Alessandro, per tutti i momenti di spensieratezza tra una lezione e l'altra.

Ringrazio gli amici delle serate al tetto, e la loro semplicità con la quale mi hanno sempre tranquillizzato e ricaricato.
Un grazie anche a tutti i miei amici di Moggio, che mi hanno trascinato in un mondo più leggero quando ne avevo bisogno.

Un grande ringraziamento ad Alessandra, che mi è stata vicina, e mi ha sostenuto nei momenti più difficili.

Per ultima, ma non per importanza, ringrazio tutta la mia famiglia: Valeria e Flavio, che stanno trovando la loro strada insieme a me, e i miei genitori, senza i quali non avrei mai potuto arrivare fino a qui e diventare quello che sono oggi.

Contents

1 | Introduction

This thesis focuses on robotic manipulation of deformable linear objects (DLOs), like cables, wires and tubes. In particular, the objective is to develop an off-line planning strategy, aimed to computing the trajectories for a dual-arm robot, allowing a manipulation of a cable in order to bring it in a desired configuration.

1.1. Robotic manipulation of deformable linear objects

Deformable Linear Objects (DLOs) are elements, like wires, pipes and ropes, with one dimension that is bigger than the other two, for this reason they are highly prone to shape deformations.
Interest in robotic manipulation of deformable linear objects is growing rapidly, especially in the automotive and the aerospace fields, in which numerous applications involving DLOs, can be found, such as for example, wiring operations, wire harness manufacturing or switchgear assembly. In many industrial applications, manipulation of cables, wires or tubes is needed but while industrial manipulation of rigid objects has been automatized for a long time, the handling of deformable linear objects is usually performed manually. The sequence of automated operations must be interrupted to allow a human operator to manipulate the DLO, affecting the operation cycle-time and making the whole process time-consuming.
For this reason the DLO manipulation can be considered a bottleneck in industry frameworks, and this is due to the several issues encountered in automating such tasks.
The main challenges are related to the highly non-linear behaviours and the large number of degrees of freedom and parameters to deal with.
Those issues lead to complex mathematical models describing the behaviours of the cable subjected to manipulation constraints. Moreover also the usage of sensor can be a huge problem. Generally vision sensors are mostly exploited, because they are efficient and intuitive to use. However although a vision system can recognize an element, for example by inspecting its color and shape, it's an hard task to recognize and track a

deformable linear object, since they are highly subjected to deformation, leading to huge variation of their shape during the operations, introducing the necessity of detailed and complex tracking algorithms. Moreover, vision sensors can fail in presence of sub-optimal environmental conditions, for example in case of presence of obstacles, occlusions and poor-lighting. Also the presence of a complex back-ground may be an issue, especially if many similar deformable linear objects are placed together. In addition, the most common industrial cables and tubes are often composed by transparent or highly reflective materials, increasing the difficulty of the usage of vision sensors.
Force sensors can be exploited, but despite they are very informative to be used with rigid objects, some issues may arise with deformable linear objects. In fact the information collected by a force sensor usually is not enough to understand the global behaviour of a cable. More sensor can be exploited in order to inspect the object in many points, leading to a very expensive set-up. For this reason force sensors are generally used to inspect local behaviours of deformable linear objects, and the application is limited on stiff-cables.
To face those issues, a model-based solution can be exploited: all the available information about the deformable linear object and the working environment should be used, for a global understanding of the operation behaviour. For this reason a model-based planning phase is fundamental, as to use in an optimal way all the available information. In the scientific literature some model-based planning applications can be found [11, 13, 17], that proposed planning algorithms involving a dual-arm robot that manipulate a cable gripped at the two ends. The usage of a dual-arm robot exploits the maximum number of degree of freedom, in order to face the complexity introduced by the cable, and reproducing the human behaviour.

1.2. Thesis purposes, achievements and methodology

The main purpose of this thesis, is the development of a model-based planning algorithm of a dual-arm robot for a deformable linear object manipulation, for various kinds of industrial cables, composed by different materials and with different stiffness. The methods proposed in the thesis have potential industrial relevance, since they are easily adaptable to practical applications, facing the need of manipulating cables, wires and tubes.
More in detail, an optimization-based planning is implemented to find intermediate shapes for a deformable linear object, to let it evolve from an initial condition to a target one. From the planned intermediate configurations of the cable the intermediate positions and orientations of the robot grippers can be extracted. Additionally a simulation environment is set up, to better analyse the cable behaviours and the collision with modeled-obstacles.

Moreover a re-optimization procedure is performed for obstacle avoidance purposes.
An algorithm for the Young's modulus identification of a generic deformable linear object is implemented, that estimates the stiffness related to a cable, even if it is a composite one. This information is provided to the optimizator and the simulator, that must behave accordingly.
Finally, several tests are carried out on a real robotic setup to validate the proposed strategy. The methodology is also applied to a use case, for a wire harness assembly.

The flowchart of the implemented methodology is summarised in Figure 1.1. Different steps are exploited, each one is fundamental for a different aspect of the work. The different phases are explained one by one along the thesis, but they are highly intertwined in the developed planning algorithm.

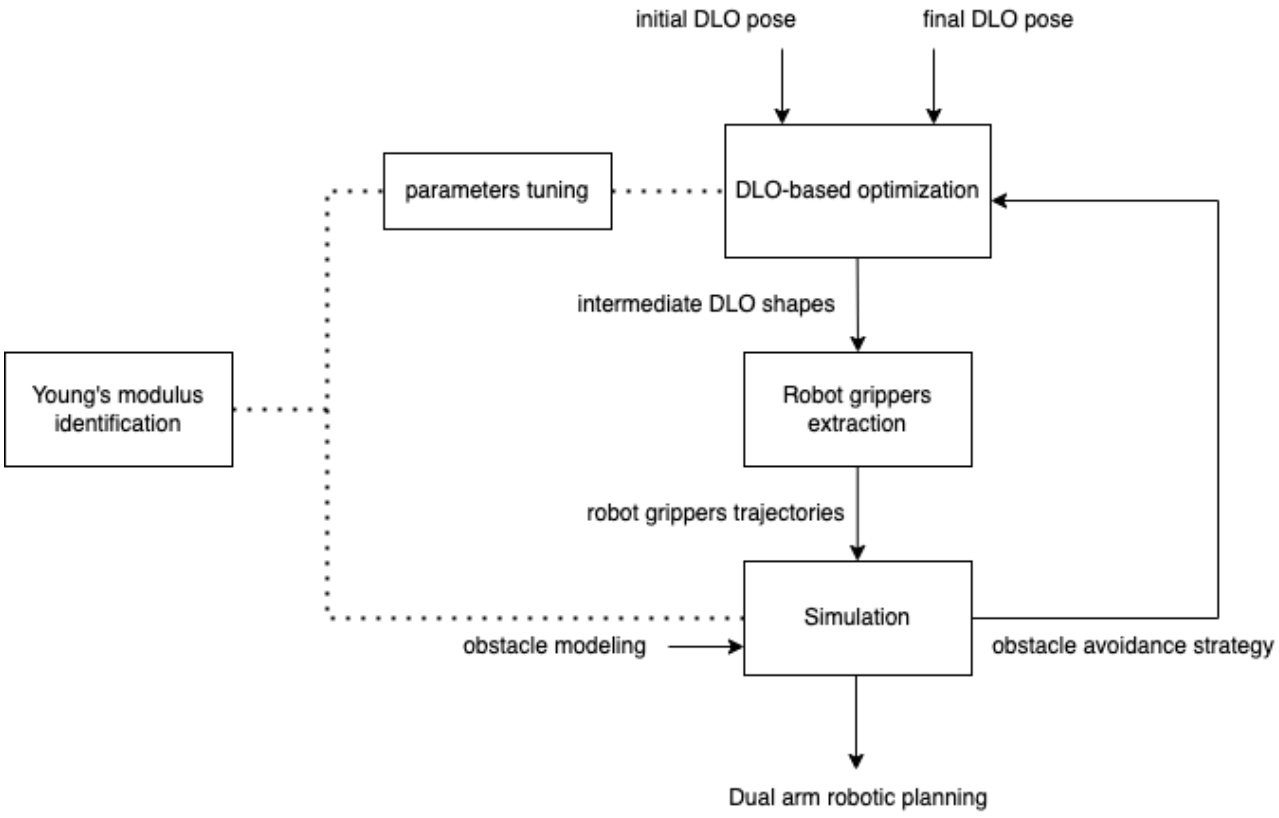


Figure 1.1: The work can be divided in many phases intertwined one with the other. This scheme is provided to have a better understanding and a global comprehension of the methodology developed in this work.

The achievements of this thesis can be summarized in:

- A planning algorithm for a dual arm robotic manipulation of a deformable linear object, in two-dimensional and three-dimensional space, generating trajectories for the robot grippers, and performing obstacle avoidance.
- An identification algorithm for the Young's modulus of deformable linear objects (in particular for composite cables), leading to a realistic and reliable estimation.
- An industrial application for the proposed methodology, involving a wire harness assembly operation.

1.3. Thesis structure

The remaining of this Thesis is articulated into seven main chapters, each one highlighting a key element in the development of the work.

- **Chapter 2** aims to summarize the main past works and contributions, relevant for modeling and planning for deformable linear objects;
- **Chapter 3** describes in detail the deformable linear object model that is exploited in this work;
- **Chapter 4** describes the optimization phase applied to the deformable linear object, and the strategies used to extract the robot grippers poses for the generation of the robot trajectories;
- **Chapter 5** reports the integration of a simulation environment that follows the optimization phase, and describes the obstacle avoidance strategy;
- **Chapter 6** provides a method for the Young's modulus identification of a generic deformable linear object, and describes a stiffness-based adaptation procedure involving the optimization phase and the simulation environment;
- **Chapter 7** reports the experimental validation and an industrial use-case application for the methodology;
- **Chapter 8** describes the conclusions and possible future works to improve the proposed method.

2 | State of the art

This Chapter aims to analyse some modeling and planning strategies involved with the deformable linear objects, highlighting the contribution of the thesis.

2.1. Models for Deformable Linear Objects

Modeling is central in automation and models are widely used for different tasks, such as simulation, planning, control and estimation. It is important to highlight that the best model is the most informative one, that allows us to get all the information we need, without overloading the computations and the communications. That's why sometimes simple but informative models are prefereed with respect to computationally heavy and more accurate ones.
As stated in [6] due to high-dynamic behaviour of deformable objects, correctly modelling the deformations of a deformable linear object is one of the main concerns when handling these kinds of objects. That is the reason why different kinds of models have been studied. In the following a brief description of different models is proposed.

2.1.1. Finite element method applied to DLO dynamical model

The finite element method (FEM) [8] allows to create a very accurate model, that can express in a realistic way the deformation of the cable. However generally FEM models are computationally heavy to be solved, ending in a too complex and time-consuming procedure. An exemple can be found in [15], where a FEM strategy is used to perform a detailed analysis for some structural cables (see Figure 2.1).

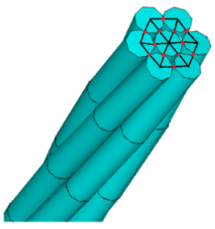


Figure 2.1: FEM model for a structural cable [15]

2.1.2. Elastic rod based DLO model

An other way to model the DLOs is to regard them as tiny elastic rods. Two common methods are used: The *Kirchoff elastic rod* and the *Casserat model*.
The *Kirchoff elastic rod* is exploited in [3] where the outhors used this model to compute the static equilibrium of the cable, considering the grippers placed at the two ends. This model allows the computation of the bending and the twisting energy. the *Cosserat rod model*, used in [7] is instead a more complete model, allowing also the computation of stretching and shearing forces. Both those strategies model the cable as a smooth and continuous elastic rod, by using a framed curve $\Gamma = \{\gamma(s); t(s), m_1(s), m_2(s)\}$, where $\gamma(s)$ is an arc length parameterized curve in $\mathbb{R}^3$ describing the rod's center-line, and $(t(s), m_1(s), m_2(s))$ is a frame on each point of the center-line, useful to express the twisting and the bending behaviour of the cable.

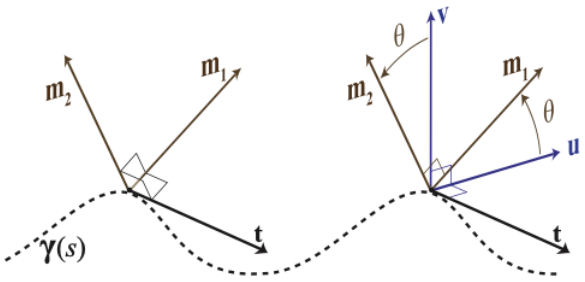


Figure 2.2: Elastic rod representation [2]

This theory can be extended toward a discrete picture, an example can be the *discrete Kirchoff elastic rod*, described in [2].
The elastic rod models involves a very reliable mathematical model, introducing comprehensive physical properties. However using those models for simulation or optimization can be challenging, because of the over complexity introduced. Moreover the introduction of constraints can be complex.

2.1.3. Local models

On the other side some works in the literature focus on the creation of local simple models for deformations, exploiting some learning procedures in order to find parameters that well approximate the system in the point of interest. An example can be found in [14], where a simple Jacobian matrix is used to model the interaction between the grippers of the robot and the deformation of the DLO, in the form $\dot{\boldsymbol{x}} = J(\boldsymbol{x}, \boldsymbol{r})\dot{\boldsymbol{r}}$, where $\boldsymbol{x}$ are the position

vectors of some points on the cable, and $\boldsymbol{r}$ are the position vectors of the grippers. An other method is proposed by Navarro et al. [16] where the DLO placed on a 2D surface is represented with Fourier series, while its physical properties are unknown, a local linear deformation model is estimated by learning some parameters from the series.
Those strategies give the possibility to handle very simple models, but they express only local behaviours, without considering the global physical behaviour of the cable: they are effective only for small deformation control problems, and there is no way to use them for a wider task, such as path optimization or simulation.

2.1.4. Mass-spring based DLO dynamical model

As a simple but informative model, the *mass-spring model* is widely used for deformable objects: it is composed by mass points and various springs, managing to provide the deformation behaviours of the cable in an intuitive way, and considering the entire cable. The model was firstly introduced by Humann and Parant [5], representing deformable objects with particles of mass attached by linear springs. A contribuition to the model is given by Look et al. [9] by using some "torsional" springs attached to the masses in order to capture the bending behaviours of the cable.

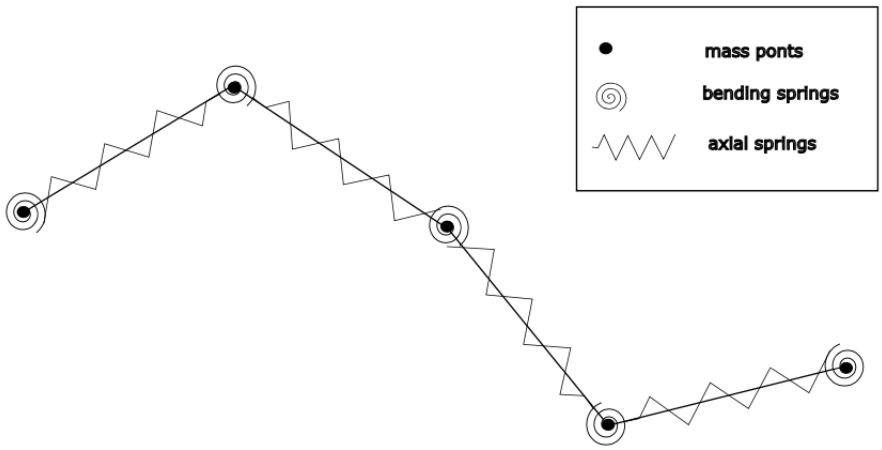


Figure 2.3: Mass-spring model with "torsional" bending springs, and linear axial springs

Finally Lv et al. [10] added some other torsional springs, in order to express also the twisting and torsional behaviours of the DLO, bringing the model to be more realistic and complete, involving the major phisical behaviours of a deformable linear object.

As previously stated, models can be exploited in different frameworks: in this work, we focus on model-based planning strategy for robotic DLO manipulation.

2.2. Planning for Deformable Linear Objects

The problem of planning a path for robots, and manipulating rigid objects, has been studied extensively in the past. Different planner algorithms exist that can generate in an efficent way a path for a robot in a complex environment.
On the other side a limited number of works in the literature deal with the planning of paths for robot manipulating deformable objects and several challenges have to be addressed. In particular the space describing the shape of the DLO has infinite dimensions, and hence an high number of configurations of the DLO exist for a given pose of the robot. These challenges have made the motion planning of a DLO an hard problem. That is the reason why different solutions can be found, exploiting different models and different algorithms.

In [13], Sintov, Macenski et al. developed a strategy to create a planned path for two robotic arms, in order to manipulate a cable by modelling it as an elastic rod. The strategy can be resumed mainly in two steps:

- the first is the exploration phase, in which stable configurations for the elastic rod are computed.
- the second is the path planning phase, in which with a suitable algorithm the shortest path connecting the goal and the target configuration is found.

The stable configuration is computed exploiting the *Kirchoff discrete elastic rod model*, based on the fact that the equilibrium configurations for an elastic rod is a six-dimensional smooth manifold, as shown by Bretl and Mc Carthy [3], which provided also a mathematical test to determine if an equilibrium configuration is stable or not.

The curve of the cable is parameterized by $t \in [0, L]$, where L is the fixed length of the cable. The rod shape is described by a map $q : [0, L] \to SE(3)$. Some functions $u : [0, L] \to \mathbb{R}^3$ are used to describe the strain, in particular u_1 describe the twisting strain, while u_2 and u_3 the bending one. Finally some functions $\mu : [0, L] \to \mathbb{R}^6$ are exploited to describe the internal forces and torques along the cable. Force and strain are related such that $u_i = c_i^{-1}\mu_i$ for $i = \{1, 2, 3\}$, where $c_1 > 0$ is the torsional stiffness, and $c_2, c_3 > 0$ are the bending stiffness. It is important to highlight that $a = \mu(0)$ describe the forces applied by the gripper on the first end of the cable: this value is known and it can be used to solve a system of differential equations described in [13] in order to find the equilibrium shape of the rod. It can be demonstrated that the choice of the point $a \in \mathbb{R}^6$ uniquely defines an equilibrium condition of the rod.
The authors used this result to map equilibrium conditions into a six-dimensional mani-

fold $A \subset \mathbb{R}^6$, and then derive the subset of stable equilibium conditions $A_{stable} \subset A$. This subset is finally approximated with a graph A_G that is used to perform a path search from the initial configuration to the final one through a Constrained Bidirectional RRT algorithm.
This approach involves mainly two problems: first a heavy computation for the solutions of the model of the cable related to the search of equilibrium conditions and stable configurations. The other problem is that the planning strategy is based on the force applied on the grippers on the cable, this requires some very reliable force actuators. Moreover in this project the cable is considered inextensible, focusing only on stiff cables, also neglecting the gravity force.

An other example of model-based planning can be found in [11], where the geometrical side of the discrete elastic rod model was used to created a path planner algorithm for deformable linear objects manipulated by two robotic arms. In particular they focused on the computation of minimal energy shapes by an adapting subdivision of the curve.
They consider the curvature and the torsion of the curve to compute the energy (q):

$$q = \sum_{i=1}^{n} (k_1^2 + \tau_i^2) s_i \tag{2.1}$$

where k_i and τ_i are the curvature and torsion associated to the segment i, and s_i is its length.

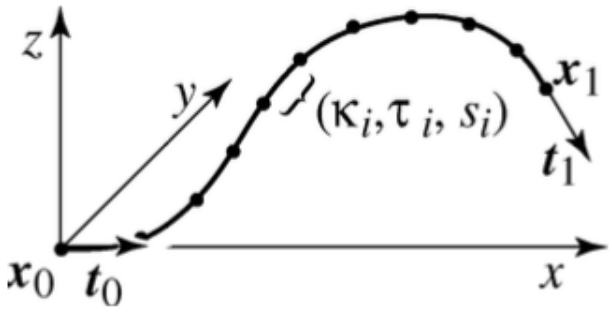


Figure 2.4: Example of a discretized curve

Then depending on the "complexity of the curve" they change its discretization (the number of segments that represent the curve) through an optimization problem. The idea is that a subdivision step will minimize the energy by smoothing the difference in curvature and torsion between all the segments.
The path planning problem can be stated as:

- given end-points of the start and goal configuration, find the minimal energy curves that satisfies the constraints.
- find a deformation of the curve (a path) from the start to the end such that all the intermediate curves are minimal energy curves.

The path is obtained through a recursive algorithm: each iteration consists of an interpolation of curves plus the application of the subdivision strategy (if the energy associated is higher than a certain threshold).
This example uses a simpler representation for the deformable linear object (only geometrical), and avoids the need of having reliable force actuators on the grippers of the robot, providing a path expressed by positions and orientations. As said by the authors an additional work must be done in order to relax the assumption of constant length of the curve, in order to add gravity, and also in order to extend the computation of the energy terms to a more accurate model, since in this solution those are computed only by looking at the geometrical curve. Moreover this solution introduces an additional degree of complexity: with the adaptive segmentation of the cable the problem of finding the minimal energy curve expands into the problem of finding the minimal energy curve plus the optimal representation for such curve, increasing the computational load.

Roussel et al. [12] provided a solution for the planning problem of an extensible elastic rod in free or contact space by working with the integration of two models:

- The discrete elastic rod model, used for the quasi-static configurations computation.
- The physical engine eXtended Dynamic Engine (XDE) that offers a realistic multi-body dynamic simulation with contact for deformable bodies.

Thanks to the dynamical simulation of the cable, the planning algorithm can take advantage of contacts, by allowing sliding motions in order to plan trajectories in narrow passages, or in enviroments with many obstacles.

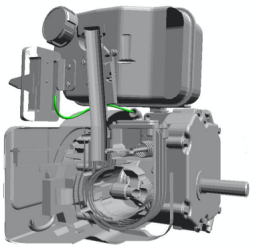


Figure 2.5: Example of planning with sliding on obstacles [12]

The strategy consists in using the elastic rod model in order to sample quasi-static free-

contacts configurations of the cable in an efficent way, and then to use the dynamic simulation to extend the exploration of the state space, including also obstacles.
The problem is that the dynamic multi-bodies simulation uses the Finite Element Method (FEM) in order to discretize the elastic rod, leading to a very massive and heavy computation: a good dynamic simulation is provided, at the cost of a very high computational time. The other issue is related to the usage of two different models, and in particular to the fitting of parameters, to ensure their convergence in quasi-static configurations.

In [17], Zhu et al. proposed a framework that allows robot to use contacts with the environment in order to shape deformable linear objects, such as cables.

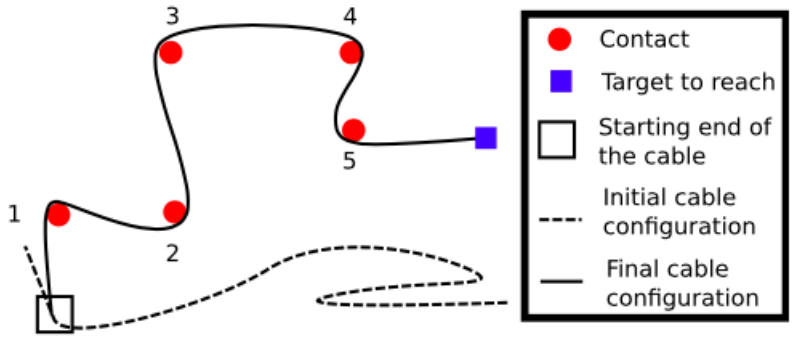


Figure 2.6: Example of contact-based manipulation [17]. The cable should go from the starting point to the end touching all the obstacle in the right order.

The strategy uses a model-free approach to represent the cable, instead focus on contacts between the cable and circular obstacles, modelling them through an index: the Angular Contact Mobility Index (ACMI), that represent the angle around the obstacle for which the object can move to break free from the contact.
The planning consists in a sequence of "primitive operations":

- the *pull* operation: it's used to move the cable between obstacles.
- the *rotate* operation: it's used to create contact between cable and obstacles and in order to shape the cable around them.

Those primitives are used during the entire task, passing from a phase to an other: The pre-contact phase, in which the cable is free of contact and it is rotated around its fixed end; the contact detection phase, in which with a vision system the contact with an obstacle is detected; The post-contact phase, in which the cable is moved on the obstacle to change the contact area and its ACMI index. This pipeline is repeated for each obstacle until the end is reached.

However, this work manages only 2D problems, and focuses only on not-rigid DLO, such as ropes or wires. Moreover this method involves a model-free solution to represent the cable, a physical model of the cable should be introduced, in order to prevent cables from excessive stress due to pull actions or deflections around obstacles.

In Table 2.1 a summary of different planning strategies analysed in section 2.2 is reported.

Method	**Model**	**Output**	**Physical behaviours**	**Gravity**	**Types of DLOs**	**E identification**
[13]	Kirchoff elastic rod	gripper forces	bending, torsion	neglected	stiff cables	no
[11]	Geometrical model	gripper poses	bending, torsion	neglected	geometrical curves	no
[12]	Discrete elastic rod + FEM	cable path	bending, torsion, axial	yes	not specified	no
[17]	Geometrical 2D cable	primitive operations	neglected	neglected	soft cables	no
[14]	local models	grippers motions	neglected	neglected	not specified	no
[16]	local models	grippers motions	neglected	neglected	not specified	no

Table 2.1: Analysis of different strategies in literature of planning for deformable linear objects.

2.3. Remarks and thesis contributions

The main limitation found in literature, concerns the difficulty to model a deformable linear object, looking for an accurate and reliable model that is not computationally heavy, in order to be easily integrated in a planning algorithm involving optimization. Some solutions managed to solve this issue with an elastic rod model, ending in a too detailed model and focusing on restricted situations.

The objective of this work is to design a planning strategy for a dual-arm robot, that brings a generic cable from a starting pose to a final one, obtaining the gripper poses that will lead the cable through some intermediate shapes. Since an explicit planning is much difficult to be addressed for deformable linear objects, the purpose of this work is to perform an implicit planning algorithm, getting a minimal energy shortest path for the cable, focusing on bending and axial forces, and avoiding obstacle collisions. For this

reason we adopted a mass-spring model, that is much more flexible and intuitive, and in particular is much less computationally intensive. Moreover, unlike the solution found in [13], we work for a position-based planning, avoiding the need of accurate force actuators and sensors for the robot.

In an industrial framework several kinds of cables need to be manipulated, depending on the operation. An example can be a wire-routing task with electrical cables, that are very soft, or another example can be an assembling operation that concerns a oil hose in a motorbike braking system. Hence, one of the goals of this work is to propose a general methodology, without focusing on a restricted stiffness for the cable to be manipulated. In particular we implemented a least-square algorithm for the identification of the Young's Modulus (E) of the cable, and provide a set of different tuning parameters for the optimization algorithms, that depend on the cable stiffness identified, managing to handle very different types of deformable linear objects.

Our work exploits a single model (the mass-spring model) in two different phases:

- the optimization phase, where the model is used to find the shortest path and the minimal energy curves from the start to the end configuration, focusing on the axial and bending forces of the cable, analysing the planning under a quasi-static point of view.
- the simulation phase, where the dynamical nature of the model is exploited, so as to integrate the planning algorithm with the torsional forces, the gravity and the obstacles in the environment.

This solution allows to have a complete domain of the cable behaviour, without increasing the complexity by the introduction of an additional complex model for the simulation.
Moreover, in order to prevent the optimization computations to be too heavy we divide the geometrical (shortest path) and physical (minimal energy) optimization in two steps. In particular the intermediate shapes from the initial to the final one are physically optimized in a decoupled way. In this way we prevent the optimization from being overloaded, managing to get better results, and a better understandability of the algorithm computations.

In Table 2.2 a comparison between the proposed methods and the methods encountered in literature can be found.

Method	Model	Output	Physical behaviours	Gravity	Types of DLOs	E identification
[13]	Kirchoff elastic rod	gripper forces	bending, torsion	neglected	stiff cables	no
[11]	Geometrical model	gripper poses	bending, torsion	neglected	geometrical curves	no
[12]	Discrete elastic rod + FEM	cable path	bending, torsion, axial	yes	not specified	no
[17]	Geometrical 2D cable	primitive operations	neglected	neglected	soft cables	no
[14]	local models	grippers motions	neglected	neglected	not specified	no
[16]	local models	grippers motions	neglected	neglected	not specified	no
Proposed strategy	Mass-spring model	gripper poses	bending, torsion, axial	yes	stiff cables, medium cables, soft cables	yes

Table 2.2: Contributions of the proposed method.

The main contributions related to this work are: (1) The exploitation of an easier but still informative model such as the mass-spring model, for an efficient integration of a model-based optimization and a simulation environment, including axial and bending forces, torsional behaviours and gravity effects; (2) The combination of a decoupled geometrical optimization and a physical one, based on the internal forces of the DLO; (3) The implementation of a Young's modulus identification for composites cables, and an adaptation mechanism that allows to work with DLOs of various kinds of stiffness.

3 | Model description

3.1. Introduction to the model

The model we used in this work is a mass-spring model, inspired by the Lv's work, described in detail in [10]. This chapter presents a general description, in order to introduce the relevant aspects that were used in this work.

Geometrically the cable is represented with a finite number of mass-points connected by some straight segments called links. The cable is composed by **n** links and **n+1** mass-points.
The total mass is distributed along the cable through the discrete mass-points. m^0 is the total mass of the cable, m_i is the mass of a single point, and they are related such that:

$$m_i = \frac{m^0}{n+1} \tag{3.1}$$

The initial total length of the cable is l^0, and the initial length of each single link is l_i^0, such that:

$$l_i^0 = \frac{l_0}{n} \tag{3.2}$$

Links are used to represent 3 kinds of springs, connecting pairs of masses as shown in figure 3.1.
Different types of spring describe different physical properties of the cable, in particular:

- Linear springs are used to describe the response of cable under stretching and compressing conditions, and they connect every adjacent mass points.
- Bending springs are used to describe the bending behaviour, and they are attached to each mass point, except for the first and the last.
- Torsion spring are used to describe the geometrical torsion and twisting behaviour of the cable

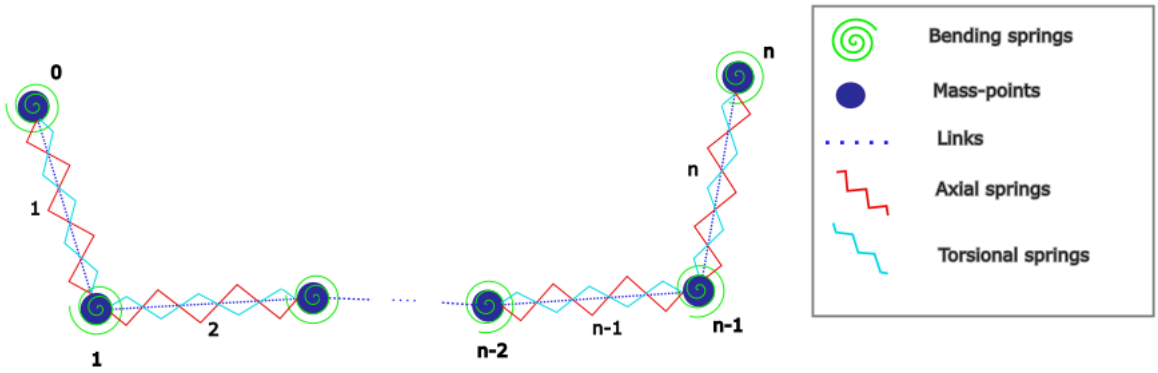


Figure 3.1: Dynamic mass-spring model.

3.2. State representation

Intuitively the state of the model can be assigned to the collection of positions of the mass points plus the torsion angle of each link:

$\boldsymbol{x_i} = (x_i, y_i, z_i)^T$ for i = 1...n+1

ψ_i for i = 1...n.

3.3. Equations of motion

The motion law of the mass points can be computed by applying the Newton's second law:

$$m_i \frac{\delta^2 \boldsymbol{x_i}}{\delta t^2} + k^d \frac{\delta \boldsymbol{x_i}}{\delta t} = \boldsymbol{F_i} = -\frac{\delta U}{\delta \boldsymbol{x_i}} + \boldsymbol{F_i^e} \tag{3.3}$$

where k^d is a fictitious damping coefficient, introduced in order to prevent oscillating behaviour of the cable. For simplicity this term is not physically modeled through dampers, but it's introduced directly in the equations of motions. $\boldsymbol{F_i}$ is the force acting on the mass point i, and it is composed by the internal forces and the external forces:

$$\boldsymbol{F_i} = \boldsymbol{F_i^i} + \boldsymbol{F_i^e} \tag{3.4}$$

external forces are known because they are the gravity force plus some forces externally applied on mass points, instead internal forces are computed exploiting the internal energy of the cable

$$\boldsymbol{F_i^i} = -\frac{\delta U}{\delta \boldsymbol{x_i}} \tag{3.5}$$

3.4. Springs and energy

U is the total potential energy of the spring system:

$$U = \sum_{i=1}^{n} U_i \tag{3.6}$$

Each term U_i takes account about the energy introduced by each spring: the linear, the bending and the torsional spring.

$$U_i = U_i^s + U_i^b + U_i^t \tag{3.7}$$

3.4.1. Linear spring

Considering the link connecting the points i-1 and i, the potential energy of the associated linear spring is:

$$U_i^s = \frac{1}{2} k_s (l_i - l_i^0)^2 \tag{3.8}$$

where l_i^0 is the length of the link i at rest condition, as shown in equation 3.2, and l_i is the actual length of the link i, computed as $|\boldsymbol{x_i} - \boldsymbol{x_{i-1}}|$. k_s is the elastic coefficient of the linear spring.

A linear spring can be considered equivalent to a generic circular rod, and by exploiting its mechanical properties we know that when it is subjected to an external axial force F_s then the axial deformation on the length is given by:

$$\Delta l = \frac{F_s l_i}{EA} \tag{3.9}$$

where E is the Young's modulus of the material in [Pa], A is the section area of the rod, $A = \pi d^2/4$, d is the diameter of the section. EA is the stretching stiffness of the rod.

The force introduced by a linear spring with the same deformation Δl is $F_s = k_s \Delta l$, and hence it follows:

$$k_s = \frac{EA}{l_i} \tag{3.10}$$

3.4.2. Bending spring

Considering the point i, the potential energy of the associated bending spring is:

$$U_i^b = \frac{1}{2} k_b \beta_i^2 \tag{3.11}$$

where β_i is the angle between the link i and i+1 and k_b is the elastic coefficient of the bending spring.

As before, a deflecting spring can be considered equivalent to a generic cantilever beam, and by exploiting its mechanical properties, we know that when the free end is subjected to a force F_b, its deflection is given by ω, such that:

$$\omega = l_i \beta_{i-1} = \frac{F^b l_i^3}{3EI} \tag{3.12}$$

where I is the moment of inertia $I = \pi d^4/64$, and EI is the bending stiffness.

The force introduced on point i by a bending spring attached on point i-1 is $F_b = k_b \beta_{i-1}/l_i$, hence we can derive k_b, exploiting the relationship:

$$k_b = \frac{3EI}{l_i} \tag{3.13}$$

Figure 3.2 shows the equivalence between a link of the model and a cantilever beam:

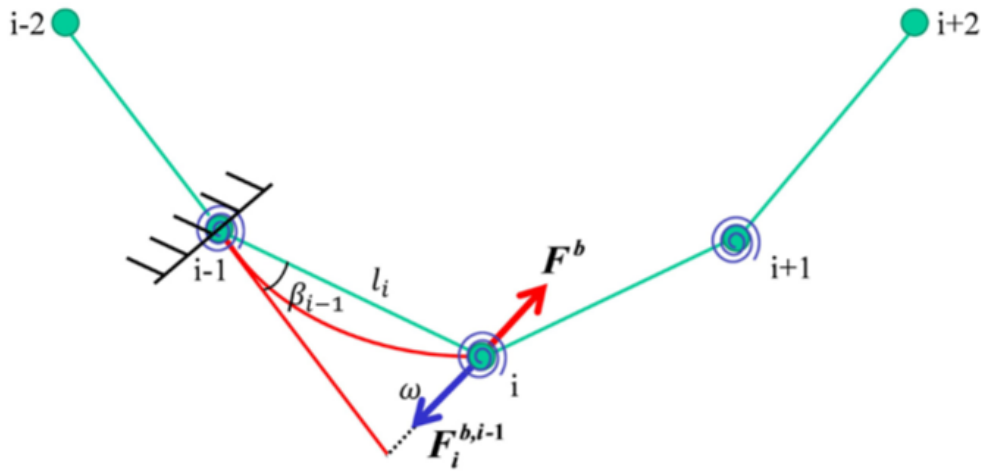


Figure 3.2: Link deflection as a cantilever beam

3.4.3. Torsional spring

The torsional behaviours of the cable can be divided in two main aspects:

- The geometrical torsion of the cable, due to an extra rotation of the links on their own axis. It's defined once for each link and denoted as φ_i.
- The original material twisting angle, denoted as θ and uniformly distributed through the links such that $\theta_i = \frac{\theta}{n}$.

The total torsion on link i is the sum of the two contributions: $\psi_i = \varphi_i + \theta_i$.

The potential energy of the spring attached to the link i can be obtained as:

$$U_i^t = \frac{1}{2} k_t \psi_i^2 = \frac{1}{2} k_t (\varphi_i + \theta_i)^2 \tag{3.14}$$

where k_t is the elastic coefficient of the torsional spring

Considering the mechanical properties of a generic circular rod subjected to a torque T, the relative torsion angle at the free end is:

$$\psi = \frac{T l_i}{G I_p} \tag{3.15}$$

where G is the shear modulus of the of the rod material, $G = E/2(1+v)$, v is the Poisson's ratio of the material, and I_p is the polar moment of inertia, $I_p = \pi d^4/32$.

The torsion introduced by the torsion spring attached to the link i is $T = k_t \psi$, and hence we can deduce that:

$$k_t = \frac{G I_p}{l_i} \tag{3.16}$$

3.5. Forces computation

As shown in equation 3.5 the forces acting on each mass can be computed from the energy. In our model we compute the forces in a more direct way, following the analysis carried out by Lv et al. in [10]:

- The axial force on each point can be computed as:

$$\begin{aligned}\boldsymbol{F}_i^s &= -\frac{\delta U^s}{\boldsymbol{\delta x_i}} = -(\frac{\delta U_i^s}{\boldsymbol{\delta x_i}} + \frac{\delta U_{i+1}^s}{\boldsymbol{\delta x_i}}) = \\ &= -k_s(l_i - l_i^0)\frac{\boldsymbol{x_i} - \boldsymbol{x_{i-1}}}{|\boldsymbol{x_i} - \boldsymbol{x_{i-1}}|} + k_s(l_{i+1} - l_{i+1}^0)\frac{\boldsymbol{x_{i+1}} - \boldsymbol{x_i}}{|\boldsymbol{x_{i+1}} - \boldsymbol{x_i}|} = \\ &= -k_s(l_i - l_i^0)\boldsymbol{u_i} + k_s(l_{i+1} - l_{i+1}^0)\boldsymbol{u_{i+1}} = \\ &= \boldsymbol{F_i^{s,i-1}} + \boldsymbol{F_i^{s,i+1}}\end{aligned} \tag{3.17}$$

As shown in Figure 3.3 the axial force on a mass-point i can be divided in two terms: $\boldsymbol{F_i^{s,i-1}}$ and $\boldsymbol{F_i^{s,i+1}}$, that are forces from the two linear springs connected to the mass-point i.

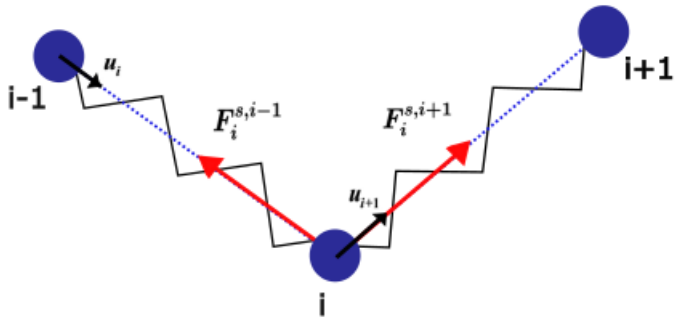


Figure 3.3: Axial force decomposition on mass point i

where $\boldsymbol{u_i}$ is the unit vector that defines the direction of the link i in the space.

- the bending force on each point can be computed as:

$$\begin{aligned}\boldsymbol{F}_i^b &= -\frac{\delta U^b}{\boldsymbol{\delta x_i}} = -(\frac{\delta U_{i-1}^b}{\boldsymbol{\delta x_i}} + \frac{\delta U_i^b}{\boldsymbol{\delta x_i}} + \frac{\delta U_{i+1}^b}{\boldsymbol{\delta x_i}}) = \\ &= \frac{k_b\beta_{i-1}}{l_i}\frac{\boldsymbol{u_i} \times (\boldsymbol{u_{i-1}} \times \boldsymbol{u_i})}{\sin(\beta_{i-1})} + \frac{k_b\beta_i}{l_i}\frac{\boldsymbol{u_i} \times (\boldsymbol{u_i} \times \boldsymbol{u_{i+1}})}{\sin(\beta_i)} + \\ &+ \frac{k_b\beta_i}{l_{i+1}}\frac{\boldsymbol{u_{i+1}} \times (\boldsymbol{u_i} \times \boldsymbol{u_{i+1}})}{\sin(\beta_i)} + \frac{k_b\beta_{i+1}}{l_{i+1}}\frac{\boldsymbol{u_{i+1}} \times (\boldsymbol{u_{i+1}} \times \boldsymbol{u_{i+2}})}{\sin(\beta_{i+1})} = \\ &= \boldsymbol{F_i^{b,i-1}} + \boldsymbol{F_i^{b,i,1}} + \boldsymbol{F_i^{b,i+1}} + \boldsymbol{F_i^{b,i,2}}\end{aligned} \tag{3.18}$$

The bending force on mass-point i can be divided in 4 terms: $\boldsymbol{F_i^{b,i-1}}$ and $\boldsymbol{F_i^{b,i+1}}$ are the bending forces acting on point i, due to the bending springs attached to point $i+1$ and $i-1$, instead $\boldsymbol{F_i^{b,i,1}}$ and $\boldsymbol{F_i^{b,i,2}}$ are the forces on mass i introduced by the bending spring on point i. Note that recursively we have $\boldsymbol{F_i^{b,i,2}} = \boldsymbol{F_{i+1}^{b,i}}$ considering

the next mass point, and that $\boldsymbol{F}_i^{b,i,1} = \boldsymbol{F}_{i-1}^{b,i}$ considering the previous one.

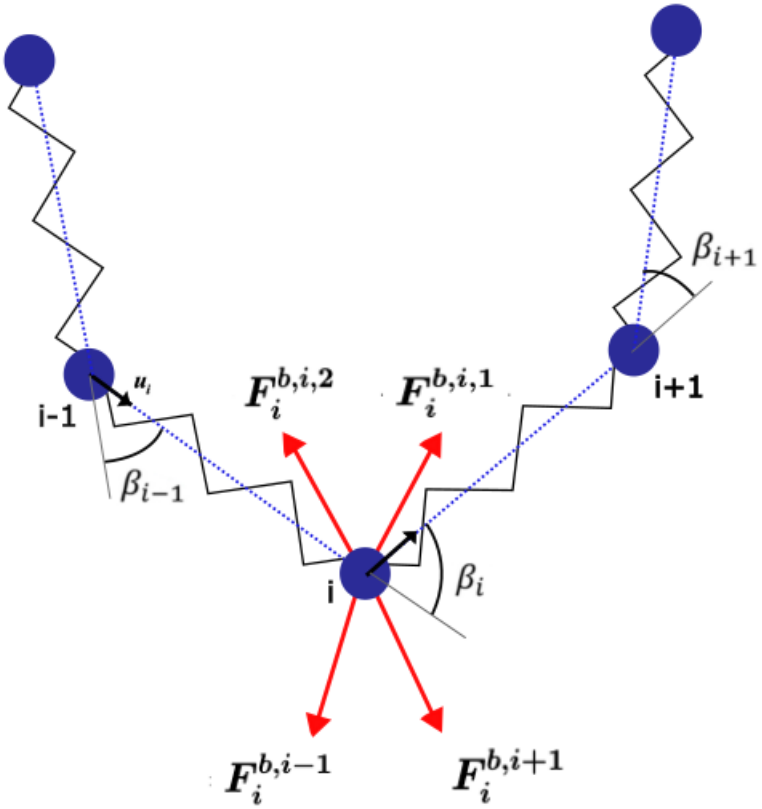


Figure 3.4: Bending force decomposition on mass point i

- the twisting force on each point can be computed as:

$$\boldsymbol{F}_i^t = -\frac{\delta U^t}{\boldsymbol{\delta x_i}} \tag{3.19}$$

The twisting force on a mass-point i can be divided into 8 terms, taking account about the torsional spring attached between mass i and $i-1$ and mass i and $i+1$.

3.6. Gravity

Since the mass is distributed along the cable through the points, each point will be affected by gravity:

$$\boldsymbol{G_i} = m_i \boldsymbol{g}, \tag{3.20}$$

Where $\boldsymbol{g}$ is the gravity acceleration, and m_i is the mass of the single mass-point.

3.7. Damping coefficient

As said in 3.3, the k_d coefficient is introduced in order to prevent the system to excessive vibrations.

Theoretically, in order to estimate this parameter a dynamical experiment should be performed, however, since it is very hard to measure the vibrations of a cable, we decide to find a rough estimation, to be used as an upper bound.

For our model the value of k_d depends on the cable material, the springs coefficients k_s, k_b k_t, and the number of mass points. To compute it we can refer to the critical damping ξ of a generic mechanical system (Figure 3.5):

$$\xi = \frac{k_d}{2\sqrt{mk}} = \frac{k_d}{2m\omega_0} \tag{3.21}$$

where m is the mass of the system, k is the elastic coefficient of the spring and ω_0 is the natural circular frequency.

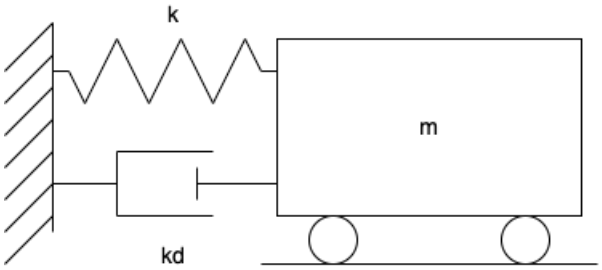


Figure 3.5: Generic mass-spring-damper system

The mass m can be set to be a single mass-point m_i. The values of ξ can be found tabulated in [1] for different materials, and we can choose an upper bound value referring to rubber material, that is $\xi_{max} = 0.05$. The value of k can be found by choosing the maximum value between k_s k_t and k_b. Generally for DLOs the following relationships are true:

$$\begin{cases} k_s >> k_b \\ k_s >> k_t \end{cases} \tag{3.22}$$

hence we can set $k = k_s$.

Finally the relationship 3.21 can be used to determine an upper bound for k_d:

$$k_d = 2\xi\sqrt{mk_s} \tag{3.23}$$

3.8. Time discretization and iterations

3.8.1. Iterations

The time is discretized into a sequence of small steps $\Delta\tau$, at each iteration, according to the Newton's second law (3.3) the following equations are solved:

$$\begin{cases} \boldsymbol{a}_i^{\tau+\Delta\tau} = (\boldsymbol{F}_i^{\tau} - k_d \boldsymbol{v}_i^{\tau})/m_i \\ \boldsymbol{v}_i^{\tau+\Delta\tau} = \boldsymbol{v}_i^{\tau} + \boldsymbol{a}_i^{\tau+\Delta\tau}\Delta\tau \\ \boldsymbol{x}_i^{\tau+\Delta\tau} = \boldsymbol{x}_i^{\tau} + \boldsymbol{v}_i^{\tau+\Delta\tau}\Delta\tau \end{cases} \tag{3.24}$$

Where $\boldsymbol{x}_i^{\tau}$, $\boldsymbol{v}_i^{\tau}$, $\boldsymbol{a}_i^{\tau}$ are respectively the position, velocity and acceleration of the mass-point i at instant τ. F_i^{τ} is the total force applied on such mass-point at instant τ.

3.8.2. Time discretization

By choosing the value $\Delta\tau$ we can regulate the convergence iterations and the number of steps in which the model must be solved. Very small time steps $\Delta\tau$ will bring the computation to be very heavy, hence a safe value $\Delta\tau = \sqrt{(m_i l_i)/k_s}$ is chosen. This value can be slightly modified, for example in order to speed up the computation by introducing less iterations.

3.9. Enhanced DLO modeling through mass-spring model

This section aims to describe an additional contribution related to the modeling of a generic deformable object through mass-spring model, developed in the Matlab model provided by the MeRLIn lab. and implemented in this work.

The objective was to create a model for deformable linear objects for which the rest condition is not straight. In industry applications often rigid cables are involved, as shown in Figure 3.6, and many times their rest configuration is curved. The implementation of the mass-spring model described in [10] does not consider this situation.

Figure 3.6: Braking hose of a motorcycle.

A refinement on the bending forces is necessary, and it can be computed focusing on two different aspects: the magnitude and the direction of the forces.

3.9.1. Bending forces magnitude refinement

The magnitude of the force, as shown in equation 3.18 depends on the length of the link l_i, on the bending stiffness k_b and on the bending angles β_i associated to the link i. This last term is the one to be modified in order to take in account about the original curvature of the cable.

Some examples of magnitude refinement of bending angles are provided in Figures 3.7 (rest condition), 3.8 (refinement case-1), and 3.9 (refinement case-2).

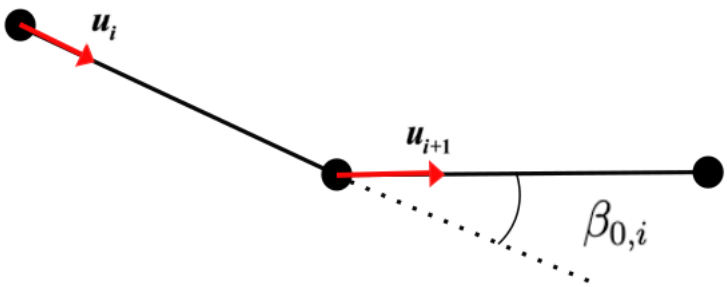


Figure 3.7: Rest condition for two links of a curved cable.

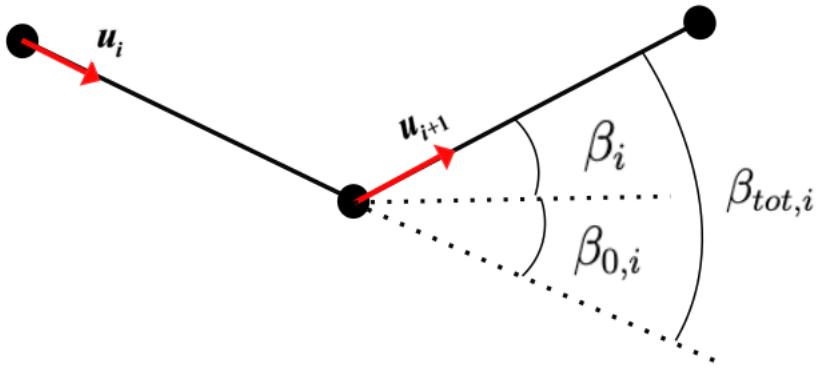


Figure 3.8: Case in which the considered bending angle β_i must be smaller than the total one.

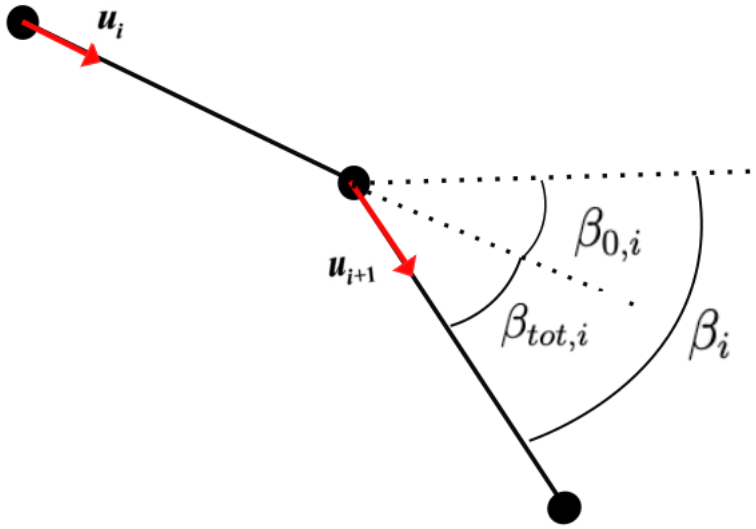


Figure 3.9: Case in which the considered bending angle β_i must be greater than the total one.

In the case reported in Figure 3.8 the refinement consist in:

$$\beta_i = \beta_{tot,i} - \beta_{0,i} \tag{3.25}$$

Instead in the case of Figure 3.9 the refinement is:

$$\beta_i = \beta_{tot,i} + \beta_{0,i} \tag{3.26}$$

3.9.2. Bending forces direction refinement

The direction of the force should be refined such that the cable will return in its rest curvature condition regardless of the motion in the space that has been applied on it.

In other words: if the cable is rotated on a plane and then is allowed to rest, the bending forces should try to drive the cable to a rotated rest condition, and not to the initial one.

In order to do that, the direction of the forces described in equation 3.18 is modified: the direction of the force will be refined by subtracting at the direction, a term, consisting in the original forces direction processed with a rotation matrix that takes into account the global motion of the cable.

For each mass-point the bending force will be :

$$\boldsymbol{F_i^b} = \frac{k_b(\beta - \beta_0)}{l_i}(\boldsymbol{u_{\beta,\tau}} - R_{i,\tau}\boldsymbol{u_{\beta,0}}) \tag{3.27}$$

where $\boldsymbol{u_{\beta,\tau}}$ is the direction of the force computed in equation 3.18, $\boldsymbol{u_{\beta,0}}$ is the direction of the form computed in equation 3.18 in the rest condition, and $R_{i,\tau}$ is the rotation matrix to be applied at time τ.

3.9.3. Energy refinement

As shown in equation 3.11 also the bending energy is related to the bending angles, and since energy is a scalar quantity it can be refined by acting on its magnitude. The potential energy associated to the mass-point i can be computed as:

$$E_i^b = \frac{1}{2}k_b(\beta_i - \beta_{0,i})^2 \tag{3.28}$$

where β_i is the bending angle associated to the bending spring, and $\beta_{i,0}$ is the bending angle under rest condition.

4 | Optimal trajectory planning for dual arm DLO manipulation

This chapter aims to describe the optimization procedure involved in the work. The final objective is to plan the gripper trajectories for a dual arm robot in order to bring a cable from an initial pose to the target one.

4.1. Introduction to optimization problems

An optimization problem can be described as a decision-making process, a process where one needs to make a number of choices trying to minimize a certain value, while satisfying a number of criteria.

- The choices are called *decision variables*
- The value to be minimized is the *cost function*, it can be composed by a number of additive terms, resulting in a multi-objective optimization problem.
- The criteria to be satisfied are the *constraints*, they can be divided in equality or inequality constraints.

Many engineering problems can be described in this way, and especially today it's important to find solutions and achieve goals in optimal manner by introducing strategies to maximize results and minimize costs. This is valid for many sectors in engineering, involving different levels and timescales, spacing in many applications and different fields. An example can be a generic industrial case, where optimization can be used in many layers, from the higher level of managing, to the more restricted one involving a single manufacturing operation.

Sometimes an optimization problem is not trivial to be solved, due to the presence of many decision variables or multi-objective cost functions, or simply due to limiting constraints, such that experience and intuitions are not enough to get an optimal solution. We are able to find optimal solutions in an efficient way, regardless of the nature of the

problem and its complexity, by looking at the optimization problem in a systematic way: The *mathematical modeling* is used to build an abstraction layer, where whatever optimization problem is described in the same way, with mathematical symbols.

- Decision variables are modeled as a vector of unknown variables:

$$\boldsymbol{x} \in \mathbb{R}^n \tag{4.1}$$

  where n is the number of unknowns.

- The cost is represented as a multivariate scalar function:

$$f : \mathbb{R}^n \to \mathbb{R} \tag{4.2}$$

- The constraints are represented by some multivariate vector functions:

$$g : \mathbb{R}^n \to \mathbb{R}^p, \; h : \mathbb{R}^n \to \mathbb{R}^q \tag{4.3}$$

  where p is the number of equality and q is the number of inequality constraints.

Functions f, g, h are provided by mathematically modeling the problem. In engineering applications this often entails knowledge from collected data, physics laws and so on.
The process of modeling produces a representation of the problem in a mathematical standard form:

$$\min_x f(x) \tag{4.4}$$

subject to

$$g(\boldsymbol{x}) = 0 \tag{4.5}$$

$$h(\boldsymbol{x}) \geq 0 \tag{4.6}$$

Note that the same problem can be posed in many different ways, for example by looking at it under other perspective or by changing the definitions of the decision variables. Accordingly the results can change a lot, in terms of accuracy, efficiency, and sometimes also in terms of "be able or not be able" to solve the problem at hand.

After the problem is set in the standard form, the approach consists in finding the decision variables such that the cost is minimized and the constraints are satisfied. The *differential calculus* is a rigorous mathematical tool: minima (or maxima) of a function are related

to stationary points, hence the solution can be found by inspecting the derivative of the function. Since optimization often deals with multi-variable problems, this results in inspecting the gradient.
Assuming f to be differentiable, the stationary points $\bar{\boldsymbol{x}}$ of $f(x)$ can be found by solving the following set of equations:

$$\nabla_x f(\bar{\boldsymbol{x}}) = 0 \tag{4.7}$$

Those equations are generally non linear, and are composed by n equations in n unknowns. The introduction of constraints is managed by the Lagrange functions, bringing in the problem an additional set of equations (generally non linear) to be solved.
The set of equations 4.7 can provide an explicit expression for the stationary points, but only in simple cases. Also, an additional analysis on higher-order derivatives is necessary to evaluate and classify the minima and maxima. This inspection is impossible to be performed analytically in many cases, for example with some non-linearities in the model, or when the problem deals with many decision variables.

To face this complexity problem the *numerical analysis* is used, giving the possibility to solve in an approximated and numerical way a set of non-linear equations, and to overcome complexity in a very efficient way, keeping high accuracy in the results. The idea is to solve the problem starting from an initial guess $\boldsymbol{x_0}$ and iteratively search by varying $\boldsymbol{x}$ according to a certain strategy, as shown in Figure 4.1.

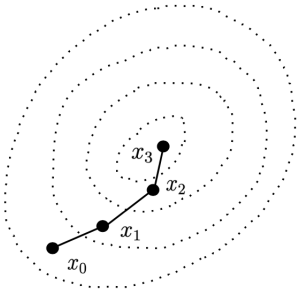


Figure 4.1: Example of iterative search, starting from x_0, where the dashed lines represent some level curves for the function f.

The decision variables $\boldsymbol{x}$ are gradually driven to the stationary points, after a certain number of iterations.

4.2. Optimization pipeline

This section aims to describe the optimization strategy to be followed in order plan a path for a robot dual arm that grasps a cable at the two ends, to bring the DLO to a target pose, starting from an initial one.

As described in Figure 4.3 the strategy consists into:

- Apply an optimization procedure considering the deformable linear object to be manipulated in the space, in order to find the path of the cable consisting into a number of intermediate cable shapes.
- Define the robot grippers poses from the computed intermediate shapes (a detailed description is provided in section 4.6): by knowing the intermediate configurations of the cable it's possible to assign the positions and orientations of the grippers holding the cable. Those poses are used to generate the trajectory of the grippers, connecting them through linear motions.

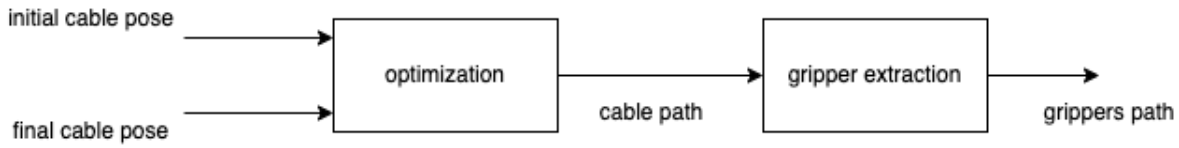


Figure 4.2: The grippers poses are extracted from the cable poses, computed by the optimization procedure.

The complexity of modeling a cable leads to the presence of a very high number of variables, increasing the difficulty for an optimization problem to be solved. This may lead to problems associated to a time-consuming procedure, or worse, to the sub-optimality of the solution. For this reason the resolution of the optimization problem is divided in different steps, differentiating several aspects that together will lead to an optimal solution, subdividing the computational complexity in many layers and making the process more efficient and understandable.

As described in Chapter 3, the mass-spring model is composed by a number of mass-points, connected through links. This geometrical model is then completed with the physical meaning of such structure, associating masses to the points, and springs to the links. This concept is the main key for the tasks subdivision, splitting the resolution in more steps, involving the geometrical side of the model and the physical one.

The optimization strategy involves three steps: the *basic geometrical optimization*, the

advanced geometrical optimization, and finally the *physical minimization*, as described in Figure 4.3:

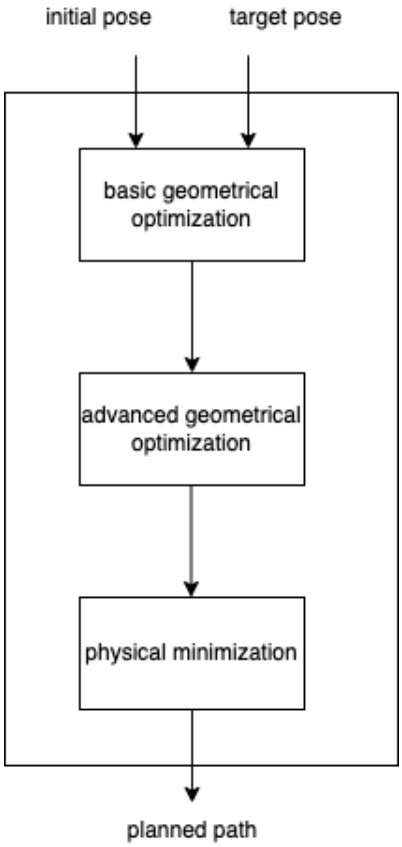


Figure 4.3: Optimization pipeline for the cable path. The total optimization problem is divided in different steps.

- The *basic geometrical optimization* generates a gradual and smooth path for the cable, from the starting pose to the target one, consisting into a number S of intermediate shapes.
- The *advanced geometrical optimization* prepares the previous optimized path in order to be physically minimized, preserving the length of the cable and avoiding eventual compression that leads to a planning that is smooth under geometrical point of view, but unstable under energy aspects.
- The *physical minimization* aims to integrate the physical model in the geometrical planned shapes, in order to minimize the forces (and hence the energy) associated to the DLO, providing minimal energy curves.

4.3. Planning through basic geometrical optimization

4.3.1. Problem description

Often in industrial sector, many actions such as assembly tasks or similar manufacturing operations, suffer the need of working by remaining in a limited area, due to the presence of obstacles or other stations in the neighbourhood. At the same time it is necessary to complete the operation with the least possible cost, in terms of motions and time cycle. This is the purpose that inspired the design of the first optimization step: move the deformable linear object from the start configuration to the target one in the most efficient way, including the minimal amount of motion and generating the shortest path.

4.3.2. Optimization formalism

In order to formalize the problem into a standard optimization one, the geometrical mass-spring model for the DLO is considered, and some data structures are created.

A single cable in a certain pose is called *shape*, and it is discretized into n links, and $n+1$ nodes (corresponding to the mass-points of the model), as shown in Figure 4.4.

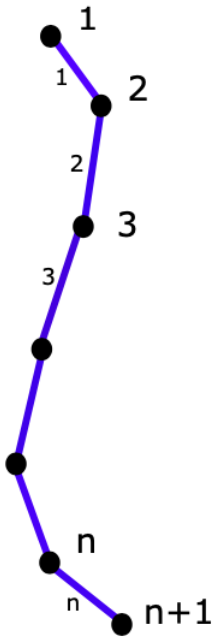


Figure 4.4: Geometrical discretization of a DLO.

Each node has its own spatial coordinates (x, y, z). A single shape i, called $\xi_i \in \mathbb{R}^{3(n+1)}$,

is defined by the collection of the positions of its nodes. So each shape will be defined by a number $3(n+1)$ of variables.

$$\xi_i = \begin{pmatrix} \begin{pmatrix} x_{i,1} \\ y_{i,1} \\ z_{i,1} \end{pmatrix} \\ \begin{pmatrix} x_{i,2} \\ y_{i,2} \\ z_{i,2} \end{pmatrix} \\ \cdots \\ \cdots \\ \begin{pmatrix} x_{i,n+1} \\ y_{i,n+1} \\ z_{i,n+1} \end{pmatrix} \end{pmatrix} \tag{4.8}$$

The grippers hold the cable with custom designed fingertips, denoted as "clips" in this work. In particular the first and second clips of the robot are placed respectively in the intermediate points of the first and last link of the cable, as shown in Figure 4.5.

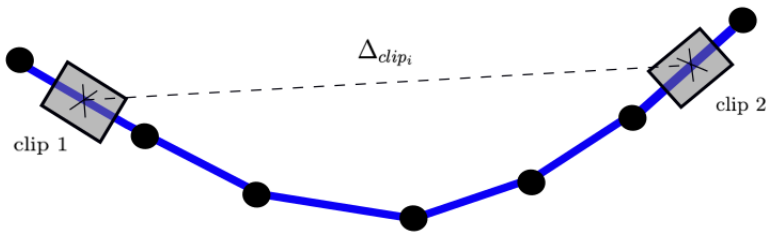


Figure 4.5: Clips of the robot applied on a shape i, and their distance $\Delta_{clip,i}$.

The optimization problem aims to find a number S of intermediate shapes starting from the initial pose ξ_o, to the final one ξ_f, producing in total $S+2$ shapes, as shown in Figure 4.6. Then it's possible to extract intermediate poses for the robot grippers holding the cable, referring to each intermediate shape.

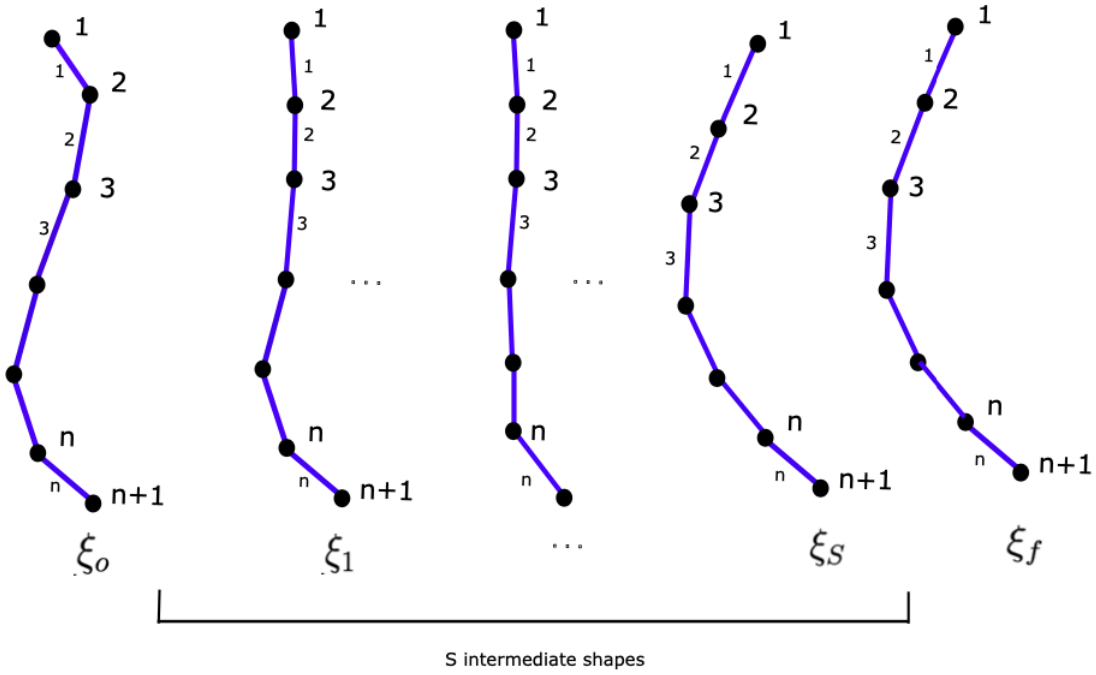


Figure 4.6: Initial and final shapes with S planned shapes.

By collecting all the subsequent intermediate shapes into a single structure it's possible to define an intermediate distribution of cable shapes, by a matrix $\xi \in \mathbb{R}^{3(n+1)\times S}$, such that:

$$\xi = [\xi_1, \xi_2, \cdots, \xi_{S-1}, \xi_S] \tag{4.9}$$

It's important to specify that that each intermediate shape is discretized by the same number off mass-points $n+1$. For this reason the problem of finding an intermediate shapes distribution for the cable can be interpreted as finding an intermediate distribution of each mass-points of the cable, from the initial configuration to the final one. Therefore a number S of intermediate distributions for $n+1$ mass points in the space will lead to a number S of intermediate distribution for the cable shape in the space.

Finally the total path $\xi_{tot} \in \mathbb{R}^{(3(n+1)\times(S+2)}$ can be addressed by inserting also the original and the final shape.

$$\xi_{tot} = [\xi_o, \quad \xi, \quad \xi_f] \tag{4.10}$$

4.3.3. Decision variables

Since ξ_o and ξ_f are given and user-defined, the optimization problem will work on the matrix ξ, where the intermediate distribution of shapes (and hence the path) are defined.

The optimization problem will have a number of $3(n+1)S$ decision variables that coincides with the definition of the intermediate shapes.

Decision variables:

$$\xi = [\ \xi_1,\ \xi_2,\ \cdots,\ \xi_{S-1},\ \xi_S\ ] \ \in\ \mathbb{R}^{3(n+1)\times S} \tag{4.11}$$

4.3.4. Cost function

The cost is related to the relative distribution of the intermediate shapes with respect to the initial and final pose of the cable. To express this distribution it's necessary to define the distances between adjacent shapes, as shown in Figure 4.7 , this is possible by considering pairs of nodes between a shape and an other.

Generically we can say that the variation of coordinates between two shapes can be defined as $\gamma \in \mathbb{R}^{3(n+1)}$, and considering the initial and final pose, and all the intermediate shapes we have that:

$$\gamma_1 = \xi_1 - \xi_o \tag{4.12}$$

$$\gamma_i = \xi_i - \xi_{i-1} \quad for\ i = 1\ \cdots\ S \tag{4.13}$$

$$\gamma_f = (\xi_f - \xi_o) - \sum_{s=1}^{S} \gamma_s \tag{4.14}$$

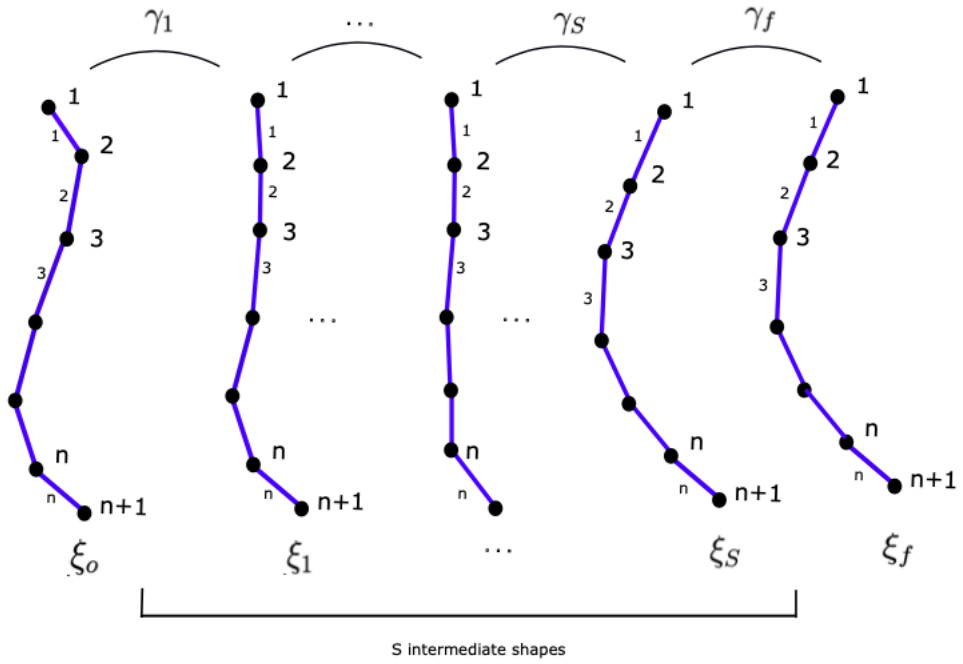


Figure 4.7: Coordinate variation computation between adjacent shapes.

In this way the decision variables are used to compute a new set of variables. Note that γ_f depends on the previous choices and hence it's not an additional variable, keeping unchanged the total number of free variables, that still are $3(n+1)S$.

The coordinate variations are then transformed into distances by the computation of the 2-norm. The idea is to create the cost function by summing all the distances between adjacent shapes,and introducing a tunable weight w_i for each term. Those weights can be set to equal values in order to have equally spaced poses of the cable, or can be increased in some areas where a thicker planning is required.

Cost function to be minimized:

$$\min_{\xi_1,\xi_2,\cdots,\xi_S} \sum_{i=1}^{S} w_i ||\gamma_i(\xi_i, \xi_{i-1})||^2 \quad + w_f ||\gamma_f(\xi_o, \xi_1, \cdots, \xi_f)||^2 \tag{4.15}$$

The optimization problem is unconstrained, and model-free.

4.3.5. Numerical results

For a better understanding of the problem, some results are reported below in Figure 4.8 and 4.9.

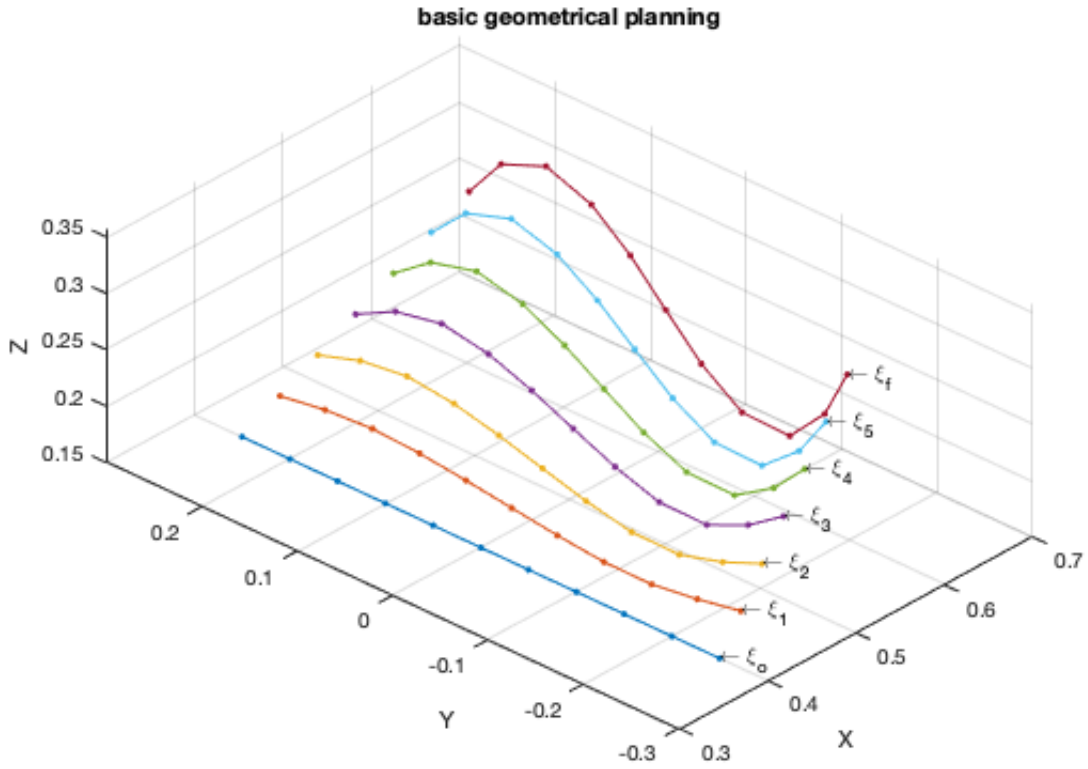
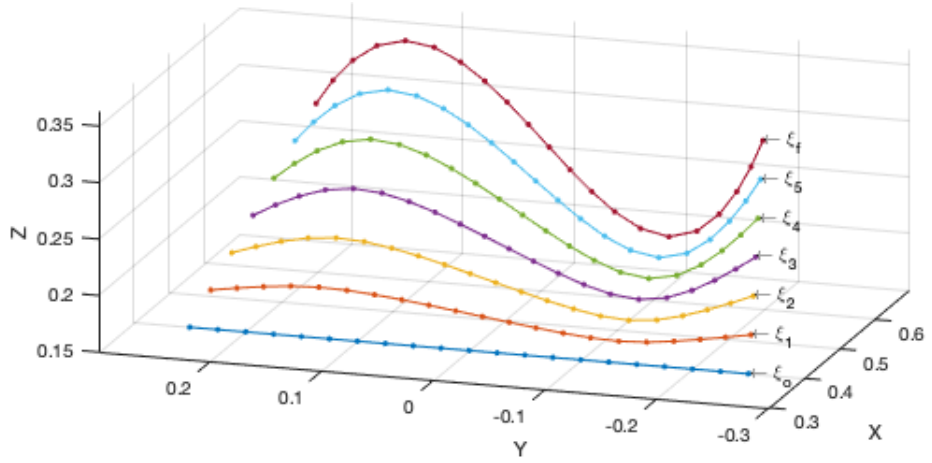


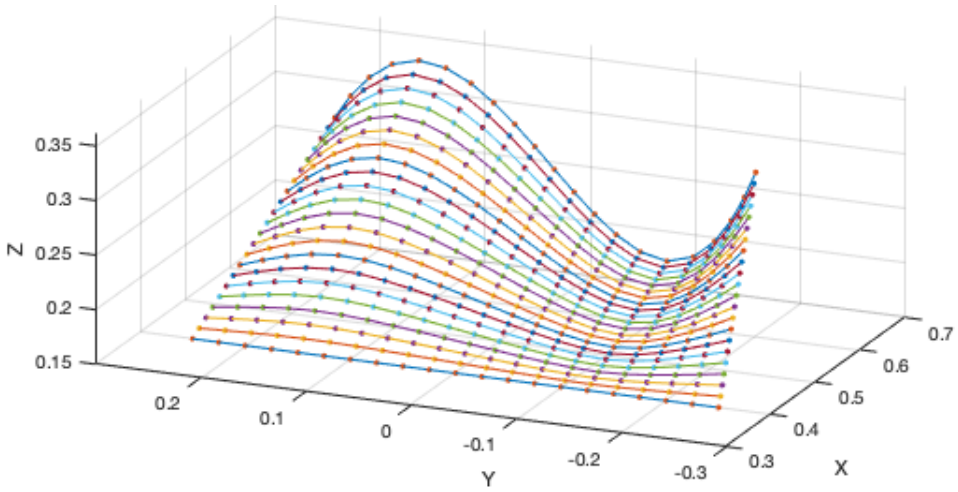
Figure 4.8: Example of basic geometrical optimization in order to plan the distribution of a cable shape from a straight pose to a deformed and translated one. The initial length of the cable is 0.5 m. The number of intermediate shapes is set to $S = 5$. The number of discretization links is $n = 10$.

The accuracy and the complexity of the problem can be modified by addressing parameters n and S, related to the number of links and the number of intermediate shapes, some examples of different tuning of those parameters are provided in Figure 4.9. Like in every optimization problem it's important to find a right trade-off, because a very accurate solution may lead to a very complex problem to be solved, instead very small values of S and n may lead to a very coarse path. Note that the modeling of a deformable linear object is a very hard task, and it always provides some inaccuracies, moreover the mass-spring model is an informative model, but not the most accurate one. For this reason it's

useless to increase too much n and S parameters, leading to an unnecessary accuracy and risking to overload the computations.



(a) Planning with $n = 20$.



(b) Planning with $n = 20$ and $S = 20$.

Figure 4.9: Examples of the same optimization problem in Figure 4.8, with variations on DLO discretization (n) and path discretization (S).

The *basic geometrical optimization* provides the shortest path: the distribution of the intermediate shapes that allows to pass from ξ_o to ξ_f with the least possible motion of each mass-point of the cable.

4.4. Planning through advanced geometrical optimization

As said in section 4.2, the optimization problem exploits the division between geometrical and physical concepts of the model (described in section 3.1) in order to subdivide the complexity of the planning procedure in more focused steps: the geometrical planning aims to find the shortest path for the cable, instead the physical planning aims to compute the minimal energy curves during the path.

The *Advanced Geometrical Planning* aims to link those two concepts in a smooth and safe way, by processing the geometrical shortest path to be ready for the physical minimization. It can be seen as a post-processing procedure for the geometrical planning, and as a pre-processing procedure for the physical planning.

4.4.1. Problem description

Sometimes the shortest geometrical path is not the best path for a deformable linear object.

Remark The shortest geometrical path is defined as solution of the optimization problem described in section 4.3, by considering only "transversal" distances between adjacent shapes. But the lengths and distances between the mass-points along single shapes are not considered.

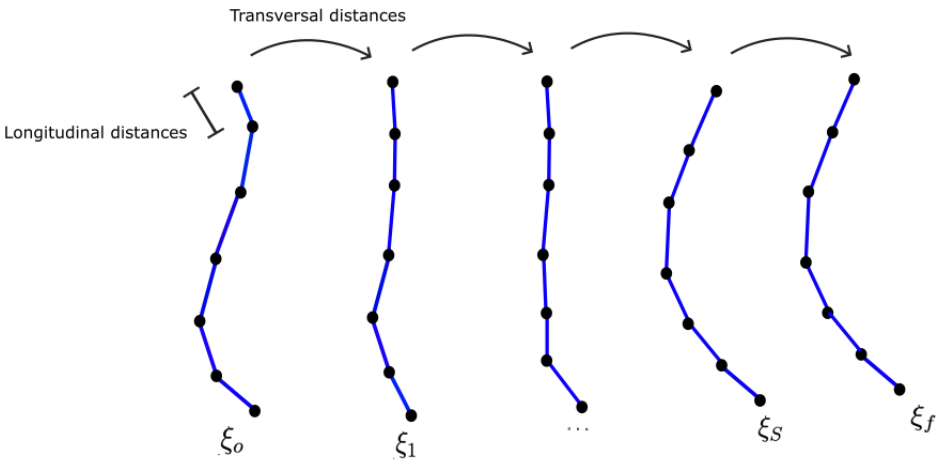


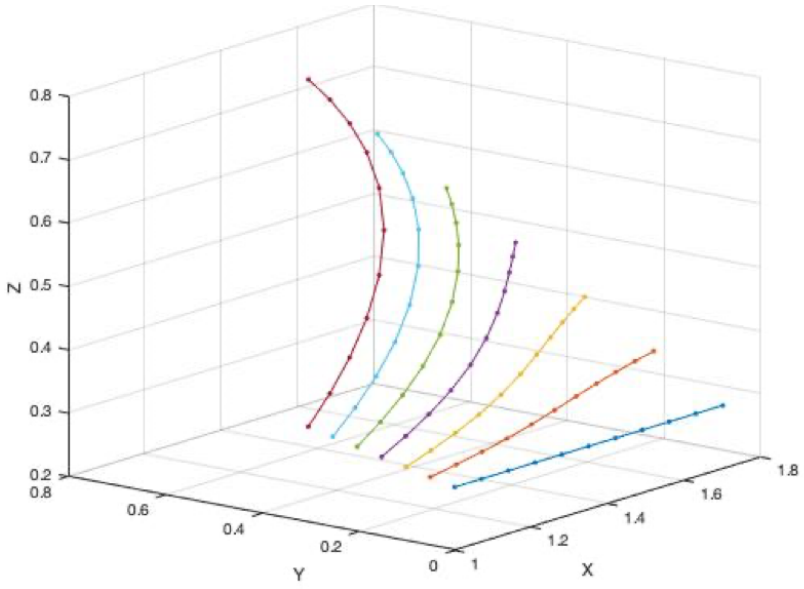
Figure 4.10: "Transversal distances" are represented through arrows, they coincide with γ_i computations. Instead we will call "longitudinal distances" the distances between mass-points along a single shape.

In some cases a conflict may occur between geometrical goals and physical goals:

- The geometrical optimization aims to find the shortest path, that is a compact path from the start to the end. Of course the geometry of the cable can be modified a lot during the path, leading to stressed configurations.
- The physical optimization aims to minimize the forces on the intermediate shapes, this equals to avoid stressed configurations during the path.

In particular, a problem arises when the cable is compressed during the geometrical optimization. In fact since the optimal geometrical path is the shortest one, the cable may result in compression deformations due to the compact motions in the path, as shown in Figure 4.11 and in Table 4.1.

Figure 4.11: Example of compression of intermediate shapes in the basic geometrical planning. The initial length of the cable is 0.7m.



	shape 1	shape 2	shape 3	shape 4	shape 5	shape 6	shape 7
Length [m]	0.70	0.59	0.50	0.46	0.50	0.59	0.70

Table 4.1: Length of each shape in Figure 4.11.

This effect may lead to problems in the physical minimization, having to work with very unstable shapes, involving very high axial forces and bringing often to bad results. For this reason an additional geometrical optimization procedure has been designed, in order

to keep the cable stretched along the shortest geometrical path, without compression of longitudinal distances, and without pull it in an excessive way.

4.4.2. Decision variables

The decision variables of the optimization problem are the configurations of the S intermediate shapes.

Decision variables:

$$\xi = [\ \xi_1,\ \xi_2,\ \cdots,\ \xi_{S-1},\ \xi_S\ ] \ \in\ \mathbb{R}^{3(n+1)\times S} \tag{4.16}$$

4.4.3. Cost function

Through the cost function we have to formalize the purpose described above.

This can be done with a multi-objective function composed by 5 terms. Each term can be computed starting from the expression of the shapes configurations $\xi_i \in \mathbb{R}^{3(n+1)\times 1}, \quad i = [1,\ S]$, and it's related to a different aspect of the problem.

Cost function to be minimized:

$$\begin{aligned}\min_{\xi_1,\xi_2,\cdots,\xi_S} \sum_{i=1}^{S} w_i||\gamma_i(\xi_i,\xi_{i-1})||^2 \quad + w_f||\gamma_f(\xi_o,\xi_1,\cdots,\xi_f)||^2 \\ + \sum_{i=1}^{S} w_\delta||\delta_i(\xi_i)||^2 + w_{clip}\frac{1}{\Delta_{clip,i}} + w_\beta||\beta_i(\xi_i) - \beta_{o,i}||^2\end{aligned} \tag{4.17}$$

- The first and second terms aim to find the shortest geometrical path, by introducing the cost terms used in the *basic geometrical planning*. These terms takes care about the "transversal" distances on the path.

- The third term aims to prevent the cable to be too stretched, with a cost involving the variation of length of the links in the shape, with respect to their length in the initial condition.
  The length of all the links composing the initial shape are saved in a structure $length_o \in \mathbb{R}^{n\times 1}$ where n is the number of links. For each intermediate shape it's computed the variable $length_i \in \mathbb{R}^{n\times 1}$, containing the length of all the links related to the i^{th} shape, with $i \in [1,\ S]$. Finally the variables $\delta_i \in \mathbb{R}^{n\times 1}$ can be computed, for each intermediate shape $i \in [1,\ S]$, such that $\delta_i = |\ length_i - length_o\ |$, describing the length variation during the planned path.

- The fourth term aims to prevent the compression of the cable, with a cost term

used for the maximisation of the distances between the two grippers of the robot that are holding the cable at the two ends (Figure 4.5). For this purpose some variables are created: $\Delta_{clip,i} \in \mathbb{R}$ describing the distance between the two clips, for each intermediate shape $i \in [1,\ S]$. It's important to note that:

$$\max\ \Delta_{clip,i} = \min\ \frac{1}{\Delta_{clip,i}} \tag{4.18}$$

with $\Delta_{clip_i} \neq 0 \quad \forall i$, since the position of the clips can't be the same during the manipulation.

- The last cost term is related to the curvature of the shapes. The previous terms introduced for the "longitudinal" aspects of the planning may bring the cable to have a flat geometry (like in Figure 4.13a), risking to introduce the curvature required for the final pose only in the final part of the planning. This is an effect that should be avoided because it is preferred to have a motion equally distributed along the path.
  On the other side, as we can see in Figure 4.8, the *basic geometrical optimization* provides a very smooth and gradual curvature deformation of the cable along the planning. The idea is to extract the information related to the curvature of the intermediate shapes in the *basic geometrical optimization*, in order to impose them in the *advanced geometrical planning*, trying to recreate a gradual curvature deformation. The basic geometrical optimization is thus necessary, in order to obtain curvature data to be used in the advanced one (Figure 4.12).

  The curvature information is resumed in the bending angles associated to the links along the shapes, some vectors are used to collect them in an efficient way: $\beta_{o,i} \in \mathbb{R}^{(n-1),1}$, $for\ i \in [1,\ S]$ are the bending angles of the intermediate shapes in the *basic geometrical planning*. $\beta_i(\xi_i) \in \mathbb{R}^{(n-1),1}$, $for\ i \in [1,\ S]$ are the bending angles associated to an intermediate shapes in the *advanced geometrical planning*. Their difference $|\beta_{o,i} - \beta_i|$ can be used in a cost term, to impose similar curvature deformation in the two solutions. An example of a result with and without curvature correction can be seen in Figure 4.27.

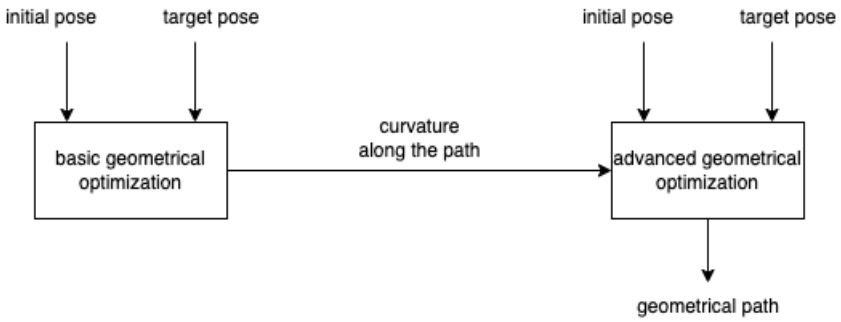
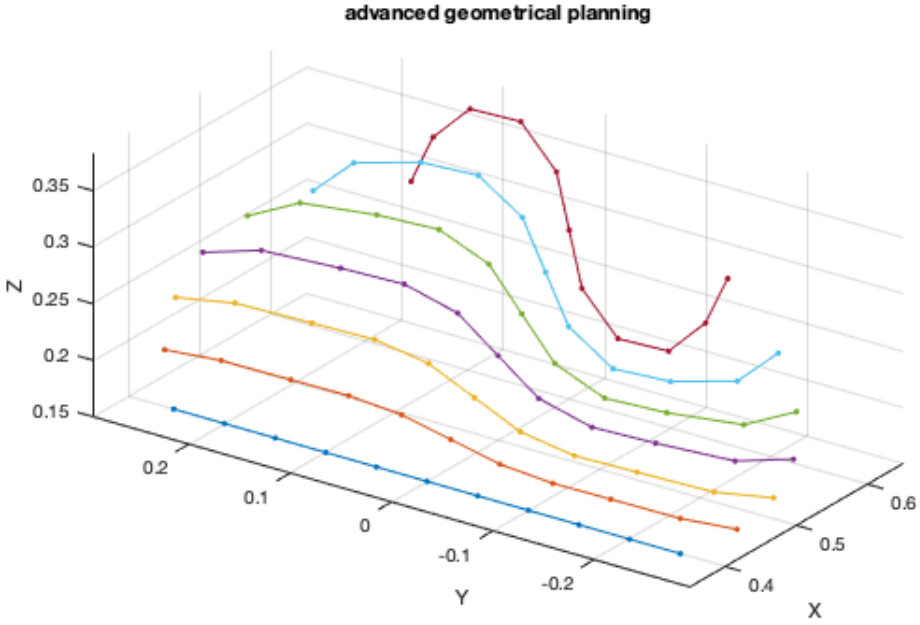
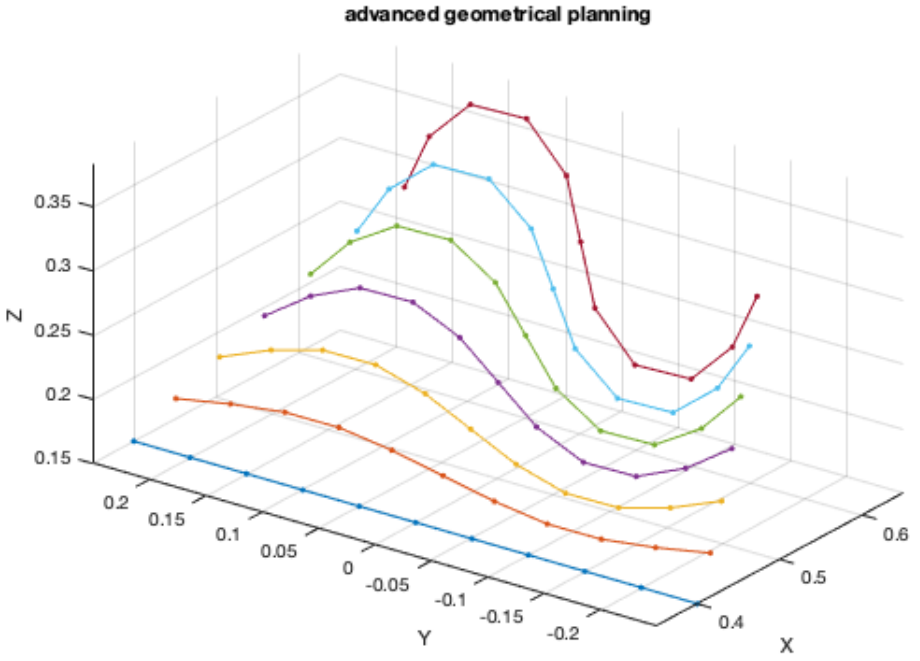


Figure 4.12: Curvature extraction and final geometrical path.



(a) Planning without the curvature cost term.



(b) Planning with the curvature cost.

Figure 4.13: Examples of *advanced geometrical planning*, with and without curvature correction.

Some weights are introduced in the cost function, in order to tune the cost by varying the importance in the problem of each single term.
The problem is unconstrained and model-free.

4.4.4. Numerical results

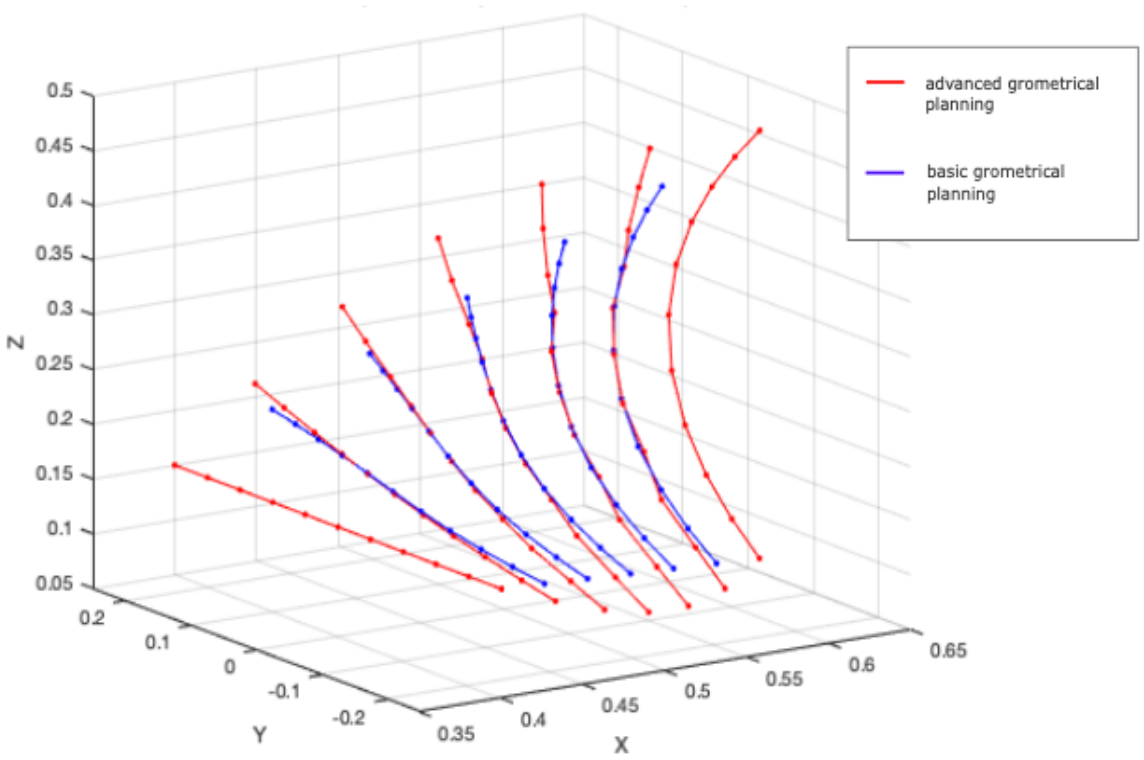


Figure 4.14: Comparison between the solution of the two geometrical planning optimizations.

	shape 1	shape 2	shape 3	shape 4	shape 5	shape 6	shape 7
Basic	0.50	0.50	0.42	0.35	0.33	0.36	0.50
Advanced	0.50	0.50	0.48	0.46	0.45	0.45	0.50

Table 4.2: Length of each shape in $[m]$, comparison between the basic geometrical planning and the advanced geometrical one.

As shown in Figure 4.14 and in Table 4.2, the *basic geometrical optimization* provides compressed shapes, instead the *advanced geometrical optimization* keeps the cable length to be in the neighborhood of its initial length (Table 4.2) , by introducing larger movements

around the shortest path. This avoids the presence of very stressed configuration of the cable.

4.5. Planning through physical optimization

The last step aims to process the geometrical planned shapes, in order to physically minimize the forces associated to the cable. Emphasis has been placed to the bending and axial forces in the cable, that can be considered the most relevant terms in the contribution of the energy associated to the cable. Moreover since the orientations of the grippers holding the cable are not considered yet, torsional forces can't be inspected in this phase. Torsional behaviours will be treated during the gripper poses definition.
As it's described in Chapter 3, energy and forces are strictly related. The minimization of the energy associated with a shape can be translated into a minimization of the forces along the cable, and vice versa. Moreover the mass-spring model implementation allows to extract those forces in a simple way. For this reason the optimization algorithm is related to the forces computed on the single shapes.

4.5.1. Physical optimization pipeline

The *physical optimization* is implemented in a decoupled way on the single shapes, and it's applied in different steps of the global optimization algorithm:

- For the pre-processing of the target configuration;
- For the energy minimization of the advanced geometrical path.

Pre-processing of the target configuration

It is important to highlight that one objective of this work is the manipulation of deformable linear objects following some minimal energy configurations. To this aim the target pose has to take a minimal energy curve. For this reason, before starting the *basic geometrical optimization* and the *advanced geometrical optimization*, a physical minimization is computed on the target final pose. To make sure to bring the cable in a stable final configuration. Moreover if the minimal energy target pose is much different from the proposed one a warning is provided by the user.

Decoupled physical minimization of intermediate shapes

After the geometrical optimizations described in Sections 4.3 and 4.4 have been computed, the path is ready to be processed: all the intermediate shapes ξ_i, $i \in [1, S]$ are physically minimized one by one, as shown in Figure 4.15.

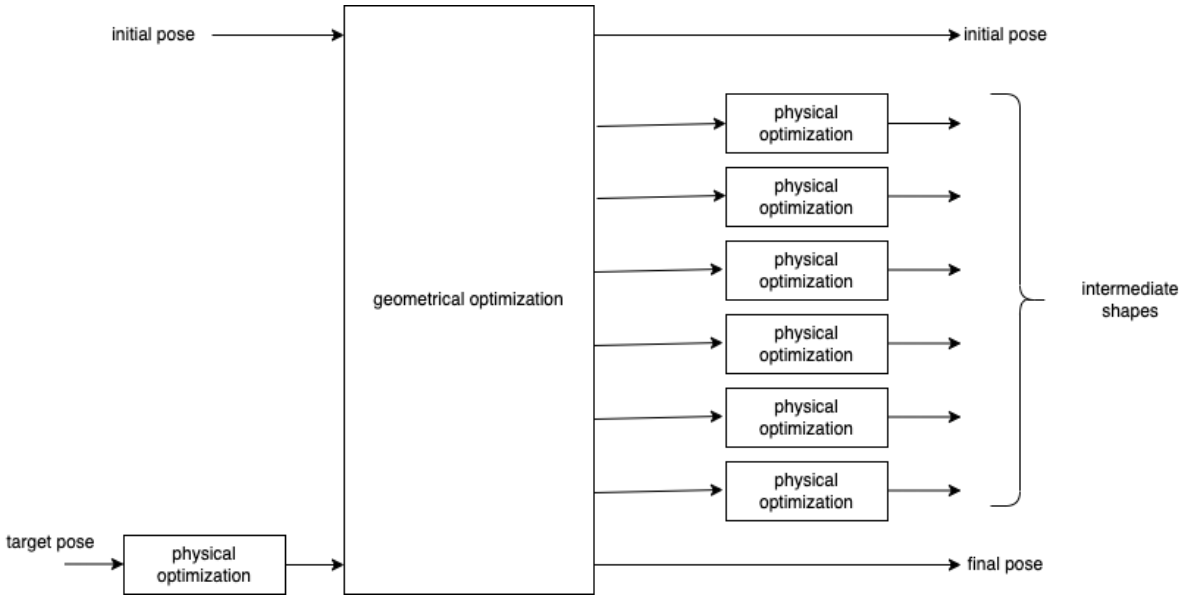


Figure 4.15: Decoupled application of physical minimization on target pose and intermediate shapes.

The decoupled strategy implemented in this step manages to obtain a more efficient algorithm, that provides better results, avoiding saturation, leading to a more controlled and scalable process.
In order to explain how the physical minimization works we need to focus on a single shape minimization. As shown in Figure 4.15, the same algorithm is indeed applied on each shape.

4.5.2. Decision variables

The strategy consists in slightly moving the mass-points of the shape in order to minimize the axial and bending forces. Intuitively the decision variables are imposed to be the positions of the mass-points along the shape and hence the vector ξ_i associated to the shape at hand.
The initial condition of the shape is the one contained in the planned geometrical path and the optimization algorithm drives the associated geometry in order to find the nearest

minimal energy configuration.

4.5.3. Cost function

The cost function is related to the internal forces of the cable that are generated by the geometrical pose and it is composed by three terms.

Cost function to be minimized:

$$\begin{aligned} &\min_{\xi_{min}} = w_\sigma ||\sigma||^2 + w_b ||F_b||^2 + w_s ||F_s||^2 \\ &\quad subject\ to \\ &\text{mass-spring model} \end{aligned} \tag{4.19}$$

- The first term is related to the geometrical variation of the shape: in order to prevent the cable to assume a shape too different from the geometrical planning one, a cost term involving the variation σ between the geometrical configuration and the minimal-energy one is introduced.

$$\sigma = \xi_{geom} - \xi_{min} \in \mathbb{R}^{3(n+1)} \tag{4.20}$$

  where ξ_{geom} is the shape proposed in the advanced geometrical shortest path, and ξ_{min} is the physically minimized shape.

- The second term is related to the bending forces associated to the shape. By exploiting the mass-spring model it's possible to extract the bending forces on each mass-point $F_{b,i} \in \mathbb{R}^3$,as described in equation 3.18. The bending forces are then arranged in a column vector F_b of the form:

$$F_b = \begin{pmatrix} \begin{pmatrix} F_{b,1} \end{pmatrix} \\ \begin{pmatrix} F_{b,2} \end{pmatrix} \\ \dots \\ \dots \\ \begin{pmatrix} F_{b,n+1} \end{pmatrix} \end{pmatrix} \tag{4.21}$$

  – The third term is related to the axial forces on the shape. As described in equation 3.17 the axial force $F_{s,i} \in \mathbb{R}^3$ on each mass-point can be computed through the

mass-spring model, and then it is arranged into a column vector F_s.

$$F_s = \begin{pmatrix} \begin{pmatrix} F_{s,1} \end{pmatrix} \\ \begin{pmatrix} F_{s,2} \end{pmatrix} \\ \dots \\ \dots \\ \begin{pmatrix} F_{s,n+1} \end{pmatrix} \end{pmatrix} \tag{4.22}$$

Notice that σ, F_b and F_s are functions of ξ_{min}, such that:

$$\sigma = \sigma(\xi_{min}), \qquad F_s = F_s(\xi_{min}), \qquad F_b = F_b(\xi_{min}) \tag{4.23}$$

Some weights w_σ, w_b and w_s are introduced for the terms, in order to appropriately tune the total cost function.

4.5.4. Numerical results

This section aims to analyse some effects of the energy minimization on different cables. In Figure 4.16 two cables are shown, they are very similar and it could seem that they are overlapping.

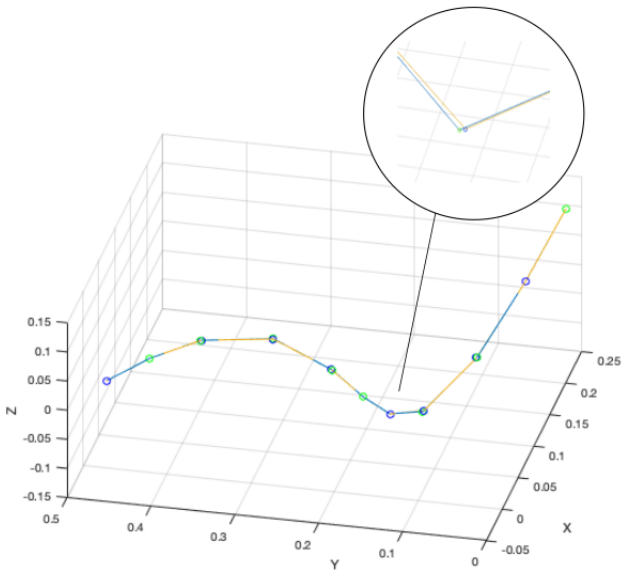


Figure 4.16: In the figure are reported two shapes, an original geometrical shape (blue), and the respective energy minimized one (yellow). Length = 1[m], n = 10, E = $1 \cdot 10^6[Pa]$.

The difference is that one is the provided geometrical shape, and the other is the minimal energy one. Since the cable is very soft and the curvature is smooth, the bending forces are not a problem. The interesting fact relies on the axial forces in the cable.

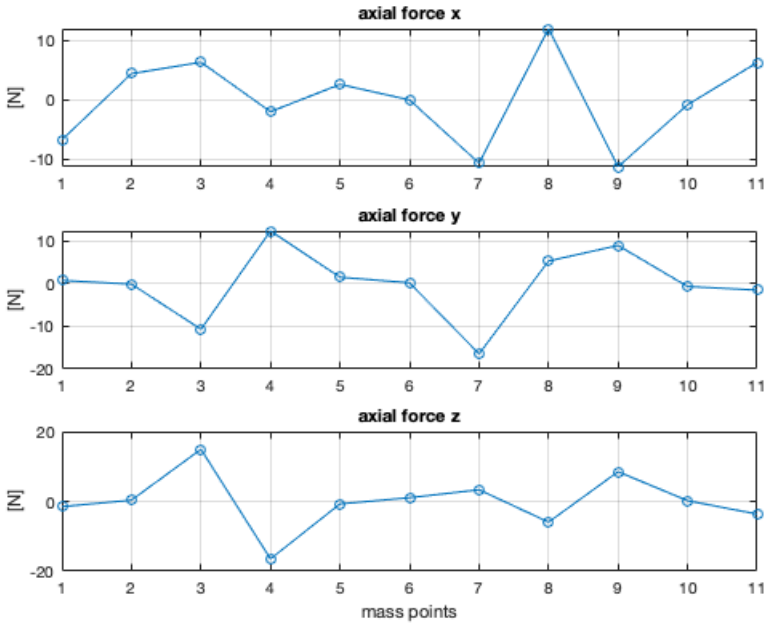


Figure 4.17: Axial forces on geometrical cable.

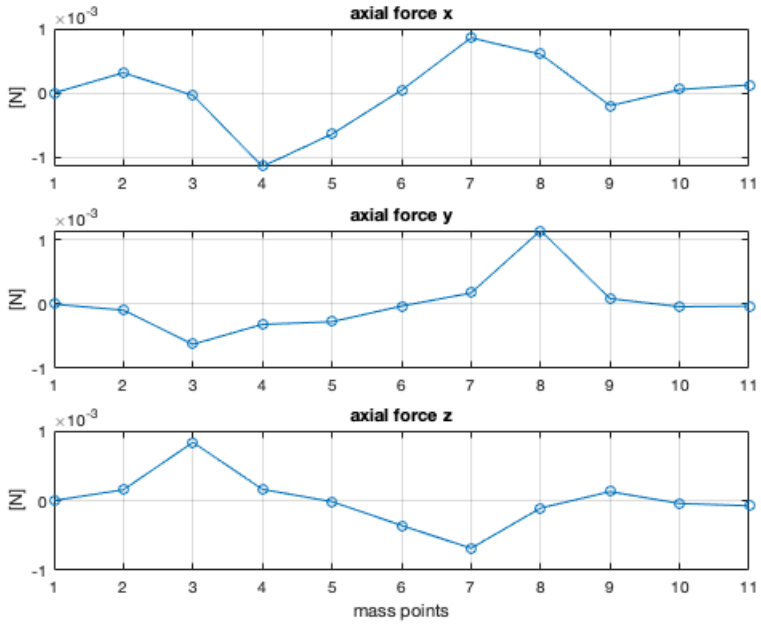


Figure 4.18: Axial forces on minimal energy cable.

In Figure 4.17 the components of the axial forces along the original geometrical cable are shown, instead in Figure 4.18 the minimal ones are reported, that are much smaller. The effect of the physical minimization on the geometry of the cable can be inspected with an analysis on the lengths of the links composing the shapes.

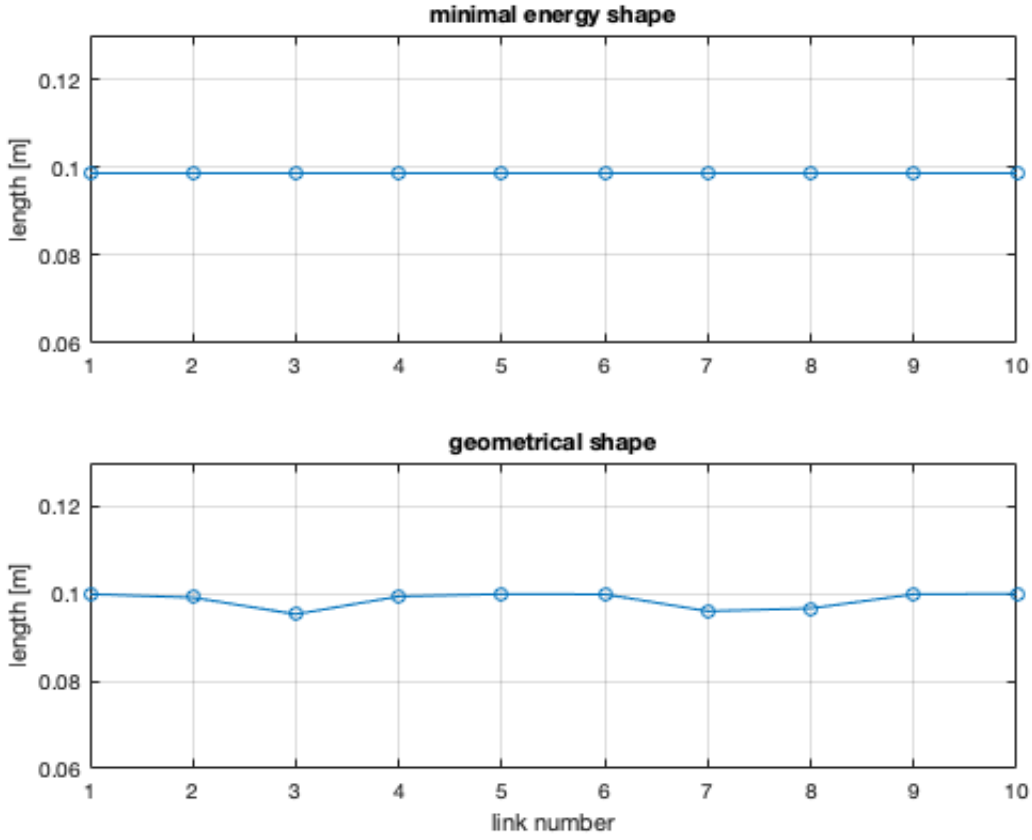


Figure 4.19: Length of the links.

As shown in Figure 4.19, in the minimal energy shape the total length is distributed uniformly along the links, unlike in the proposed geometrical shape in which the variation of length is evident, leading to high compression/stretching forces that may bring into model instability during a simulation.

Moreover, the geometrical effect of the physical minimization on the bending angles can be seen in the example of Figure 4.20: while the geometrical shapes accentuate the curvature in the two deflection areas keeping the cable quite straight in the middle part, the minimal energy shapes aims to distribute the curvature along the cable, providing more smooth and uniform bending angles.

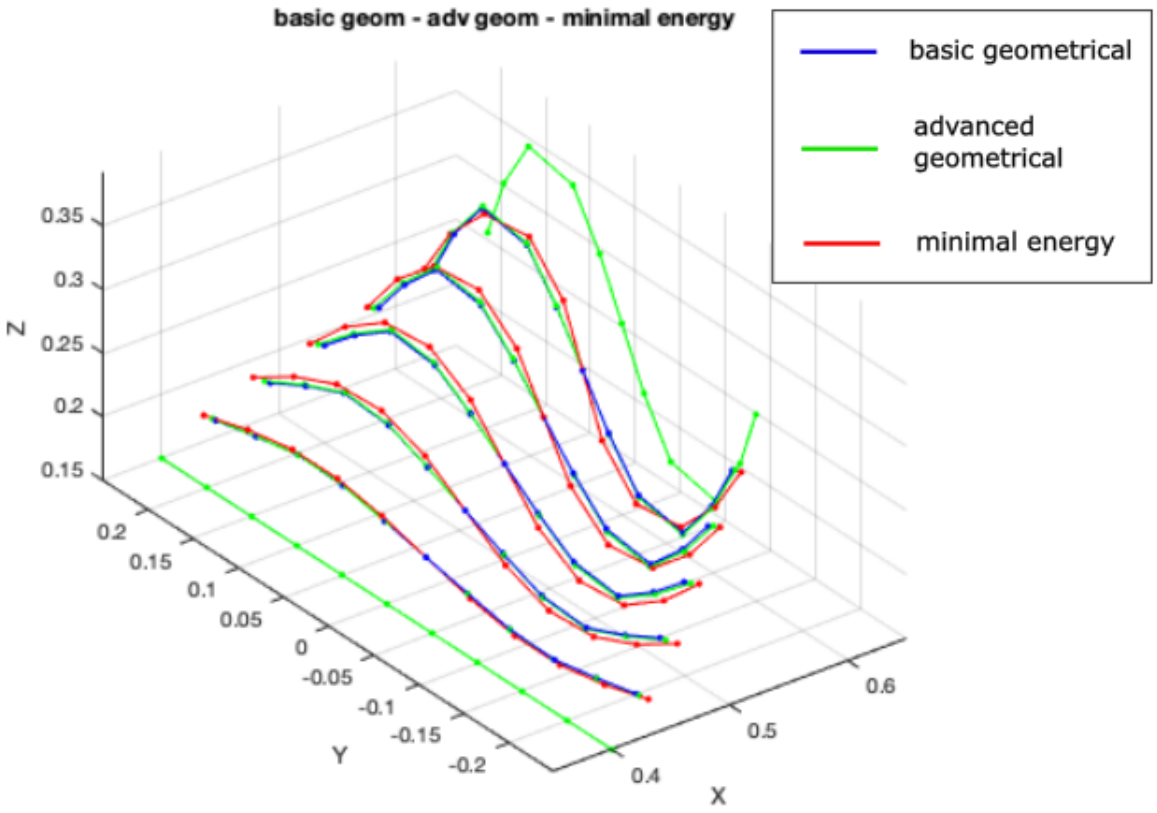


Figure 4.20: Optimization steps for a cable of length $l_0 = 0.5[m]$, $E = 1e7[Pa]$, discretized into $n = 10$ links and $n + 1 = 11$ mass-points.

In Figure 4.21 and 4.22 an analysis on the forces related to the geometrical path and the minimal energy path shown in Figure 4.20 is provided.

- The axial forces are well minimized: the associated mean and variance is decreased a lot after the physical minimization, leading to very low and uniform forces along the path, with a benefit on the stability of the planned path.
- Also the bending forces are minimized: The two peaks on mass-points 3 and 9 are smoothed, leading to an uniform distribution of the bending stress along the shape.

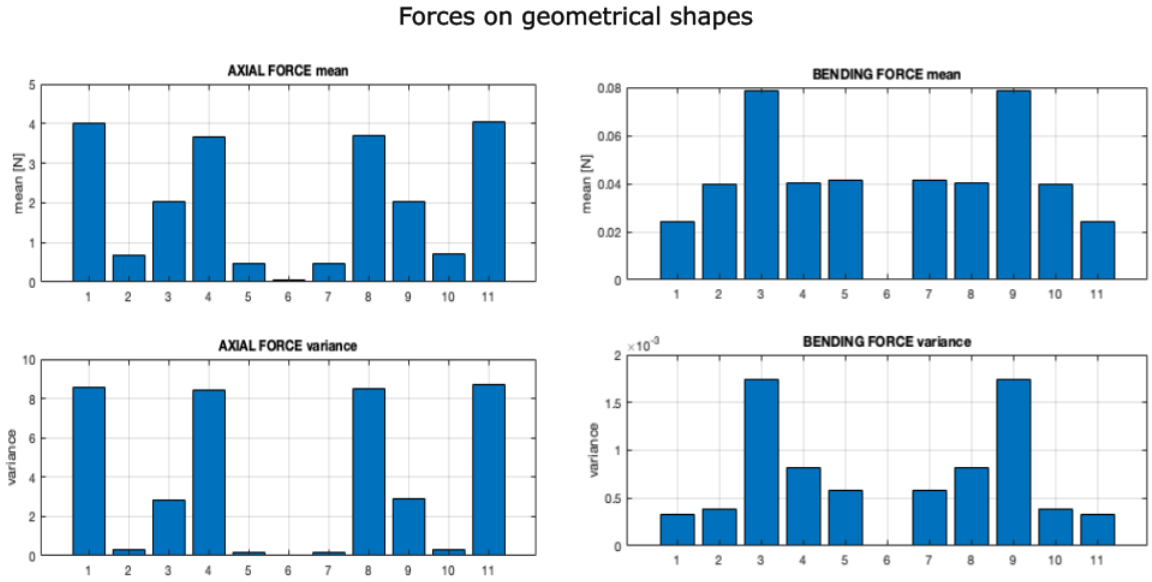


Figure 4.21: Mean and Variance of the mass-points forces along the shapes, considering the geometrical path of the example shown in Figure 4.20.

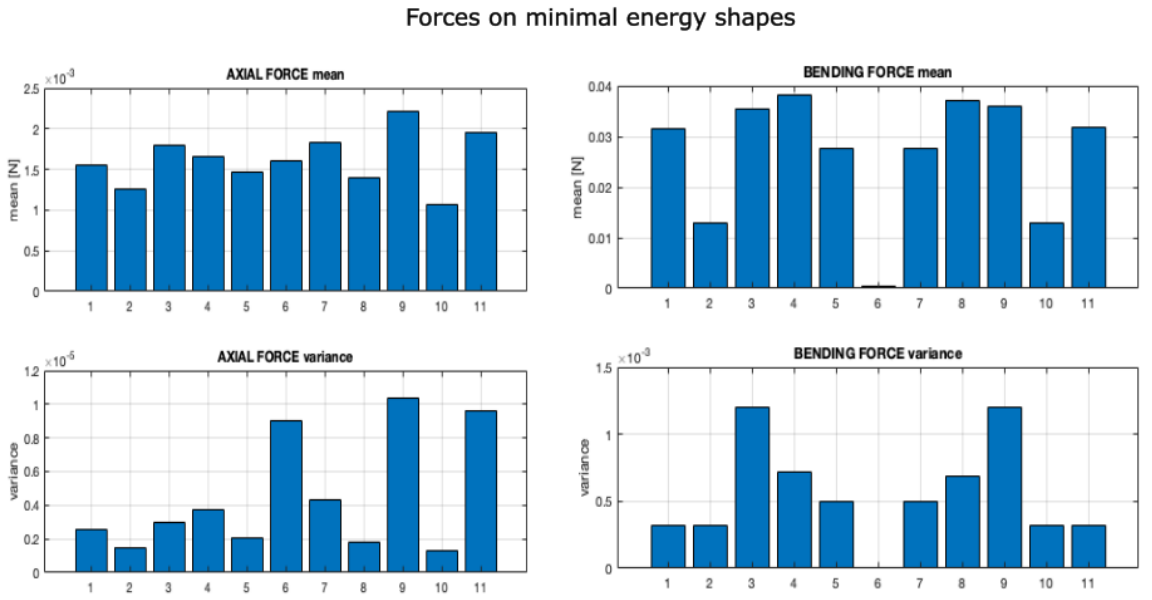


Figure 4.22: Mean and Variance of the mass-points forces along the shapes, considering the minimal energy path of the example shown in Figure 4.20.

4.6. TCPs trajectories definition

Once the intermediate shapes for the cable are planned, it is necessary to plan the path for the grippers of the robot, which have to hold and move the cable during the manipulation. The goal is to extract the positions and orientations for the robot grippers, by exploiting the information contained in the planned path $\xi_o, \xi_1, \cdots, \xi_S, \xi_f$. Moreover particular attention must be posed in order to do not involve torsional behaviours with wrong grippers orientations.
The robot has two clips that grasp the end of the manipulated DLO. Each clip is identified with a frame placed on it, in this way it's possible to specify their positions and orientation. The frames are placed on the robot clips in the exact point where the cable will be gripped (in the center of the end effector), as shown in Figure 4.23.

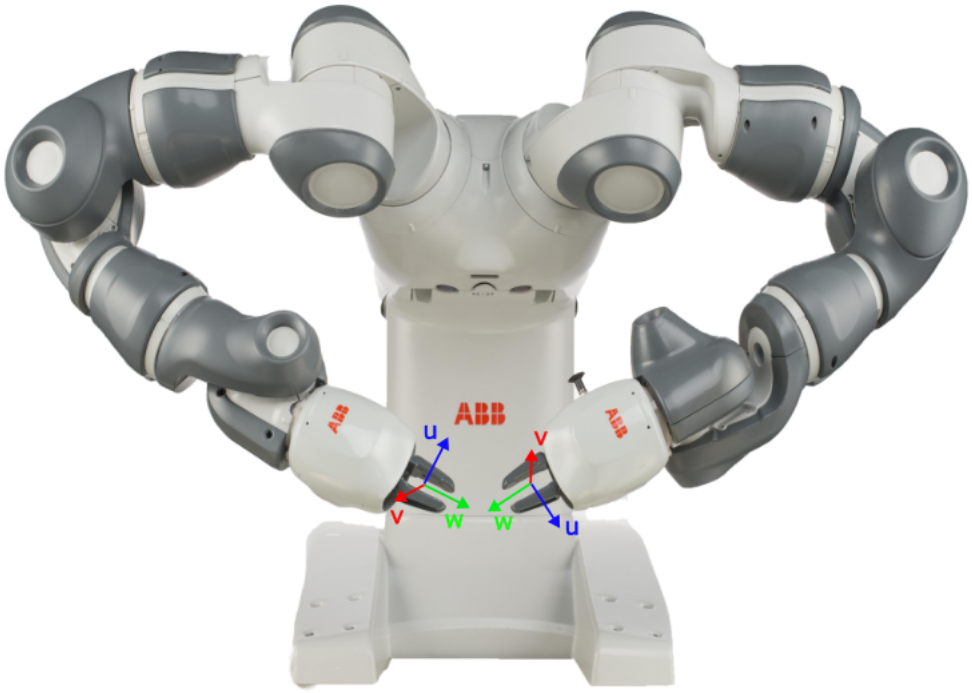


Figure 4.23: Frames placed on the two grippers of the robot. The orientations are user-defined, each frame has a u-axis, a v-axis and a w-axis, creating a right-handed frame.

The positions to be assumed by the clips are extracted from the shapes by exploiting the middle point on the first and last links of the cable, as described in Figure 4.24. This because some custom grippers are used, and the size of each gripper on the cable can be considered equal to the size of a single link.
The u-axis of the clip frames are forced to coincide with the axis of the first link and last link of the cable, as shown in Figure 4.24.

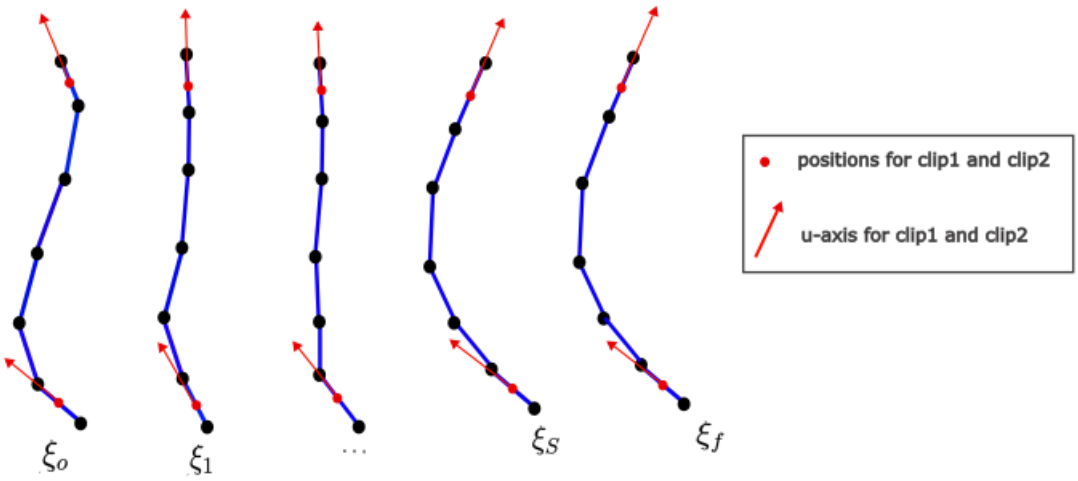


Figure 4.24: Extraction of positions and u-axes for the clips of the robot, from the planned shapes $\xi_o, \xi_1, \cdots, \xi_S, \xi_f$.

In order to define the other two axes (v-axes and w-axes) of the clip frames some algorithms have been designed, since the planning and the mass-spring model do not provide the information to compute them in an unique way.
Three methodologies have been developed: the *auxiliary vector method*, the *minimal rotation identification method* and the *equivalent rigid rotation method*. Each methodolgy has some pros and some cons.

4.6.1. Auxiliary vector method

This solution is the simplest one. The idea is to define an user-chosen auxiliary vector, to be used for the computation of the v-vector of the clips:

- The u-axis vectors of the clips are provided as mentioned above, from each planned shape.
- The v-axis vectors are obtained by computing the cross product between the u-axis and a constant user-defined vector.
- Finally the w-axis vector is obtained by computing the cross product between the u-axis and the v-axis (that are perpendicular), in order to define a right handed frame.

Due to the constant behaviour of the auxiliary vector, a common component is maintained along the shapes, resulting in a smooth and gradual rotation of the frames along the path, as shown in Figure 4.25.

Some issues may occur, related to the cross-product computation. To overcome those problems it's enough to change the auxiliary vector, for example by keeping attention to do not choose a vector parallel to an u-axes. Note that a change on the auxiliary vector will change the v-axes and w-axes computed in the algorithm, for this reason sometime an additional rotation matrix should be designed in order to match the initial frame with the initial condition of the robot.

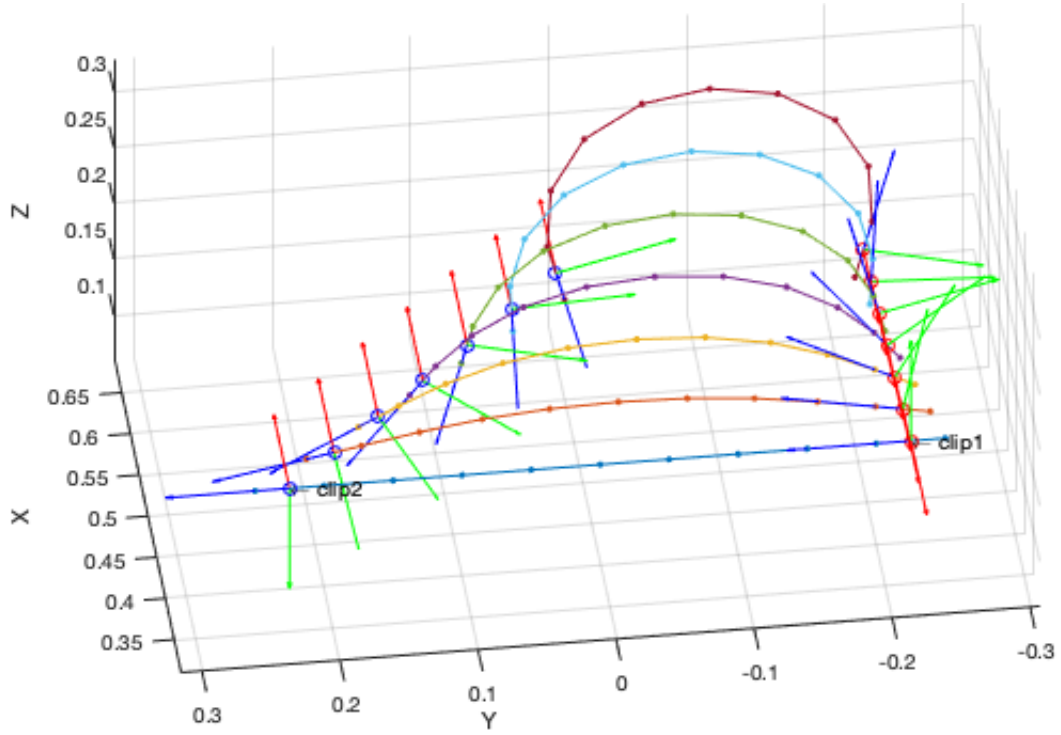


Figure 4.25: Example of clips computation by using the auxiliary vector method.

4.6.2. Minimial rotation identification method

This solution aims to gradually rotate the clip frames of the robot, starting from some initial frames, and applying on each planned step a minimal rotation in order to follow the u-axes along the planned shapes. The initial orientations of the clips are user-defined, hence an initial v-axis and w-axis are provided by the user for both clips, depending on the robot initial configuration. The same procedure is applied on both clips of the robot.

- The u-axes of the robot are known (as described in Figure 4.24), hence we can process them in order to find the *minimal rotation* that relates pairs of subsequent u-axis between adjacent shapes, as shown in Figure 4.26.

$$R_i^{i-1} \in SO(3) \;\; \textit{such that} \;\; \boldsymbol{u_i} = R_i^{i-1}\boldsymbol{u_{i-1}} \tag{4.24}$$

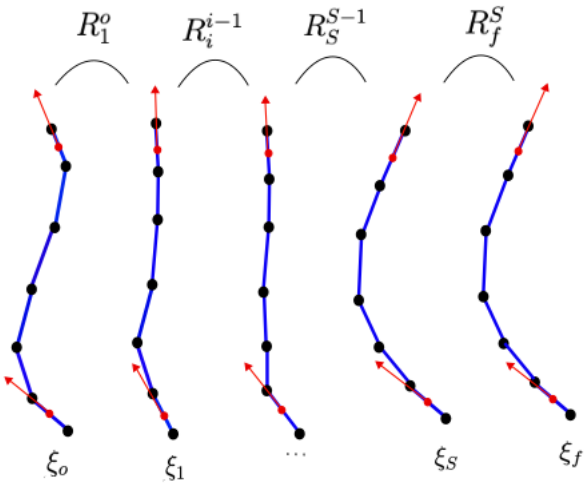


Figure 4.26: Example of rotations relating the u-axes for the clip1, the same is done for the clip2.

- After the computation of all the rotation matrices along the planned shapes, it's enough to apply the identified rotations to the v-axes and w-axes, in order to gradually rotate the initial frame.

Minimal Rotation identification

An optimization problem has been designed in order to find the matrix connecting two adjacent u-axes. The reason is that two vectors in a three-dimensional space can be connected by an infinite number of rotations, since some information are missing. This information is related to the rotation around the own axis of the vectors, that in the planning problem means including some torsional effects during the manipulation, and this should be avoided.
For this reason it's not enough to compute a rotation between adjacent vectors, but the *minimal rotation* is needed. For this purpose an optimization strategy solves the problem in an efficient way.

Decision variables:

The decision variables are the elements of the matrix $X_i^{i-1} \in \mathbb{R}^{3,3}$, related with the rotation to be identified $R_i^{i-1} \in SO(3)$ such that:

$$R_i^{i-1} = I + X_i^{i-1} \tag{4.25}$$

where $I \in \mathbb{R}^{3,3}$ is the identity matrix, such that if $X_i^{i-1} = 0$, then $R_i^{i-1} = I$, that means

a null rotation.
The cost function aims to minimize the variations of R_i^{i-1} with respect the null rotation I, and hence a minimization on the decision variables X_i^{i-1} is involved. This can be interpreted as finding the *minimal rotation*.

Cost function to be minimized:

$$\min_{X_i^{i-1}} ||X_i^{i-1}||^2 \tag{4.26}$$

The cost function must be minimized by satisfying some criteria: the matrix $R_i^{i-1} = X_i^{i-1} + I$ must be a rotation matrix (this can be traduced in imposing orthogonality and normality to the solution $X_i^{i-1} + I$) such that it rotates the vector u_{i-1} to coincide with u_i.

Constraints to be satisfied:

$$(X_i^{i-1} + I)^{-1} = (X_i^{i-1} + I)^T \tag{4.27}$$

$$\det(X_i^{i-1} + I) = 1 \tag{4.28}$$

$$u_i = (R_i^{i-1})u_{i-1} = (X_i^{i-1} + I)u_{i-1} \tag{4.29}$$

Through this optimization problem, the estimation of the minimal rotation between two vectors is possible. By applying the problem on each pair of vectors u_i and u_{i+1} for the clip-1 and the clip-2 along the shapes, it's possible to find all the clip rotations involved in the path, in order to smoothly rotate the robot.

This solution is more computationally heavy with respect to the *auxiliary vector method*, but it provides some improvements:

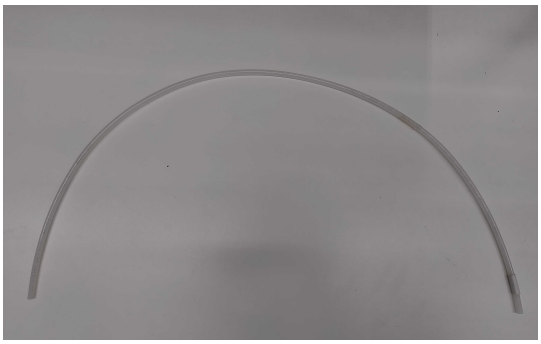
- the initial frames orientation are user defined, this means that there is no need of computing an additional rotation matrix.
- there is no need of creating an auxiliary vector for the algorithm, this overcomes all the problems related to the cross-product computation.
- the algorithm identifies the minimal rotation to move the clip frames form a shape the the following one, this prevents the inclusion of extra rotation components in the motion, that may lead to bring undesirable torsional behaviours in the cable.

As it is explained in subsection 4.6.3, the minimal rotation identification method does not work for cables with high stiffness, therefore another method, "equivalent rigid rotation method", is proposed.

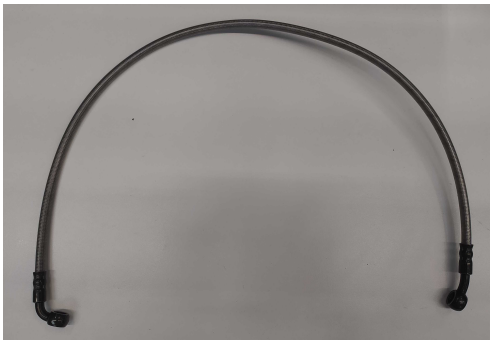
4.6.3. Equivalent rigid rotation method

In many cases in industry applications some cables characterized by high stiffness are involved in the task. An example can be a brake hose in automotive sector (Figure 4.27b), or a pipe for compressed air (Figure 4.27a). Often the elevated stiffness of the cables brings to have a rest configuration that is curved, associating to the cable a more complex geometry with respect to the straight equilibrium configuration.

Hence a new method called "equivalent rigid rotation method" has been designed for those cases, when the planning algorithm deals with cables for which the rest configuration is not straight.



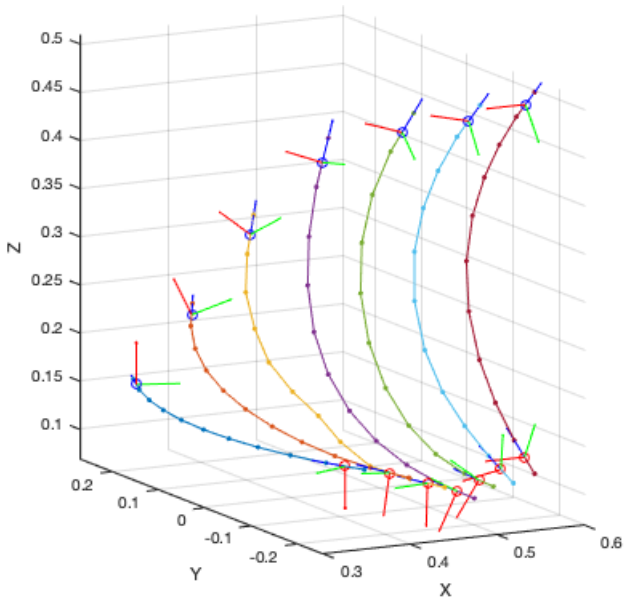
(a) Air compressed hose for industrial applications..



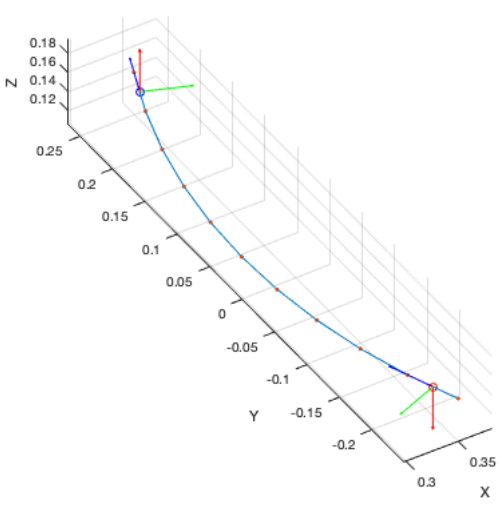
(b) Braking hose of a motorcycle..

Figure 4.27: Examples of DLOs with curved rest condition.

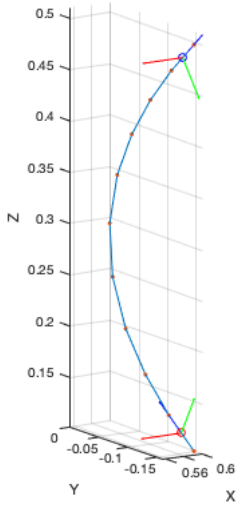
In those cases the minimal rotation is no more the right solution to prevent the introduction of torsional behaviours in the cable, since the cable has a complex geometry which must be considered in the extraction of the orientation of the clips. In Figure 4.28 it is shown that the *minimal rotation identification method* leads to wrong rotations applied on the two clips.



(a) Total planning with wrong clip extraction.



(b) Initial shape grippers.



(c) Final shape grippers.

Figure 4.28: Example of wrong clips extraction. The minimal rotation is no more the right solution. With a comparison between the clips on the initial and final shapes it can be noticed that the cable is subjected to a torsion about 180° that will compromise the target shape.

For this reason a more complex algorithm has been developed. The strategy consists into building an equivalent rigid body around the cable by inspecting its geometry, then calculate the rotation associated to the body along the path in order to be applied also to the clips. This procedure is suggested by the high stiffness characterizing this kind of DLO.

- The first step aims to create an equivalent rigid body associated to the cable. This can be done by using the chord of the arc identified by the shape, as shown in Figure 4.29. Then a right-handed frame is obtained by processing the axis of the first link of the cable, the chord and their perpendicular vector. This procedure is applied on each planned shape (Figure 4.30).

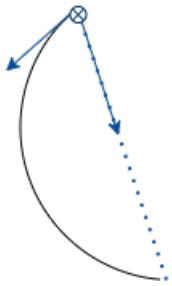


Figure 4.29: Rigid-body computation and vectors used for the creation of the right-handed frame.

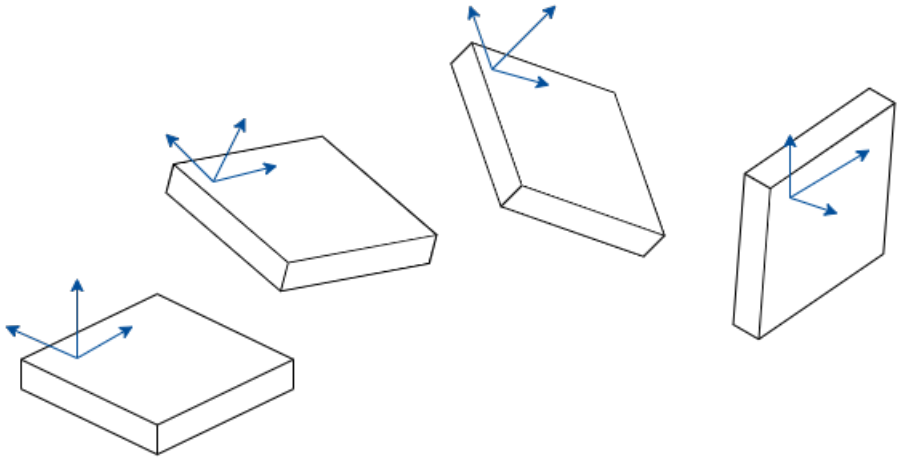


Figure 4.30: Equivalent rigid-bodies with the associated right-handed frames.

The algorithm is computed two times: the first time for the first clip, by placing the rigid-body frame in the point in which the first clip will hold the cable; Instead the second time it's done the same way but for the second clip. This because the rigid body is only an approximation of the cable and the links can slightly move in different ways, hence a focus in the two areas of interest are needed.

- The second step aims to compute the rotation matrices between adjacent shapes considering the equivalent rigid body frames. This time an optimization procedure

is not necessary, because all the information is available: in particular the right-handed frames provide the rotation matrices to be used in the computations, and the following relationship holds:

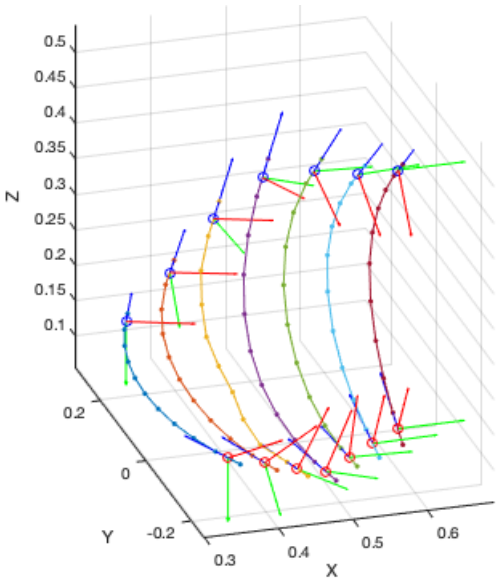
$$R_i^o = R_i^{i-1} R_{i-1}^o \tag{4.30}$$

where $R_i^o \in SO(3)$ is the rotation matrix related to the right-handed frame associated to the rigid-body on the shape i, for $i = [1, S+2]$, with respect to the base frame (the identity). $R_i^{i-1} \in SO(3)$ is the rotation matrix connecting two consecutive right-handed frames along the path, it can be easily computed as:

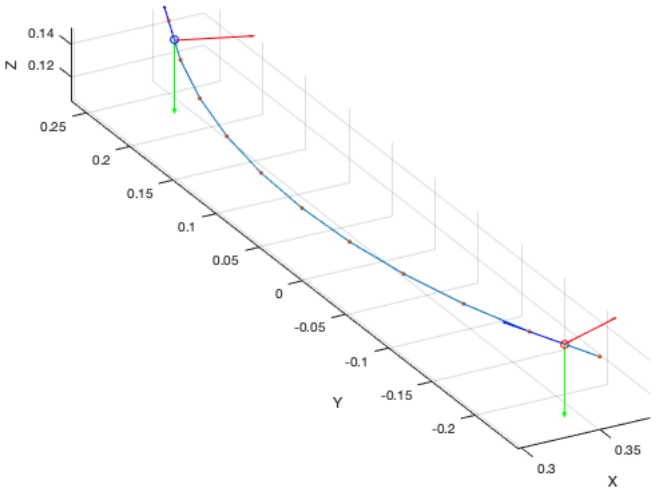
$$R_i^{i-1} = R_i^o (R_{i-1}^o)^T \tag{4.31}$$

- The last step aims to use the computed rotations in order to move the clips according to the planned path. The identified rotations, computed from the rigid-body frames are progressively applied to the clips of the robot, starting from their user-defined configuration, in order to rotate them accordingly.

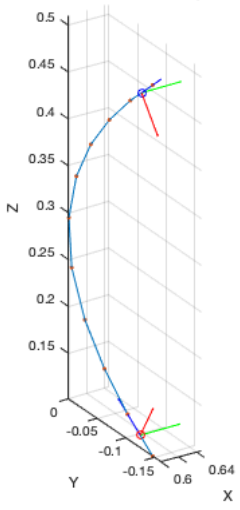
An example is shown in Figure 4.31, where the *equivalent rigid rotation method* has been applied in order to extract the orientations of the clips along the path. The obtained result can be compared to the one obtained in Figure 4.28.



(a) Total planning with clip extraction.



(b) Orientation of the clips on the starting shape.



(c) Orientation of the clips on the final shape.

Figure 4.31: Example of manipulation, by inspecting the initial and the final shape it can be noticed that the cable is not under torsional behaviours.

Table 4.3 summarizes when to use each methodology.

Method	When to be used
Auxiliary vector	when cross-product conditions can be easily checked
Minimal Rotation identification	when the auxiliary vector method fails, and it's necessary to avoid torsion introduction
Equivalent Rigid Rotation	when stiff and curved cables are involved

Table 4.3: Gripper extraction methodologies.

5 | Iterative planning through simulation

The optimization phase described in Chapter 4 provides the path for the end-effectors TCPs (tool center points) of a dual-arm robot to bring a cable from an initial pose to the final one: a serie of S intermediate distributions of the mass-points composing the cable is provided. Those mass-points configurations can be translated into a series of S intermediate shapes of the cable, that can be used to extract the subsequent poses for the robot grippers. Those poses are connected by linear motions of the TCPs in order to provide the trajectories for the dual arm robot.

The provided path has the following features:

- The planned path is based on the shortest geometrical one. This aspect is achieved by the basic and advanced geometrical minimization.
- The shapes of the cable along the path are minimal energy curves, thanks to the physical minimization step, which minimizes the bending and the axial forces of the cable.
- The orientations of the robot clips do not introduce torsional stress. This is achieved thanks to the minimal rotation computation or the equivalent rigid-body methodology introduced for the clips extraction from the shapes.

While the optimization phase cares about the geometric planning and the internal forces in the cable, some external aspects related to the environment are not considered as, for example, the gravity force acting on the cable and the obstacles involved in the working area. For this reason an additional phase involving a simulation of the cable exploiting the DLO physical model introduced in Chapter 3 has been implemented. In particular a static simulation is performed (explained in section 5.2) for the stabilization of the shapes and the obstacle avoidance.

5.1. Simulation environment

The simulation aims to move and simulate the cable under some user defined manipulation effects. Therefore the mass-spring model described in Chapter 3 is implemented in Matlab, where constraints related to the grippers of the robot are added.
It is worth mentioning that the mass-spring model describes the behaviour of a deformable linear object under different types of constraints, for our needs it is necessary to simulate a dual-arm robotic manipulation and hence those constraints fall to the robot grippers holding the cable at the two ends: the grippers of the robot are applied on the first and last link of the cable, providing the motion under which the cable must behave accordingly. As shown in figure 5.1, the inputs of the simulation are:

- The initial shape of the cable;
- The time history of the positions and orientations of the two grippers (clips), that will drive the cable ends during the simulation;
- The obstacles can be modeled and introduced as additional constraints.

The simulator will solve the motion equations at each time step $\Delta\tau$, as described in Section 3.8, and the output will be the cable behaviour during the simulation under the manipulation constraints.

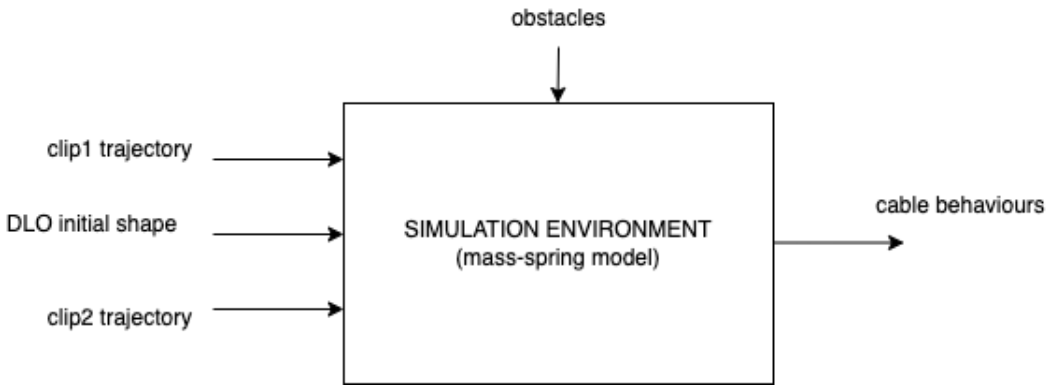


Figure 5.1: Simulation environment. Specifying the time history of the clips holding the cable, the time history of the cable behaviour is provided.

5.2. Static simulation

Since deformable linear objects are very difficult to be modeled and simulated (considering also non-idealities involved with the structure of the cable such as plastic deformations), we prefer to keep low speed while manipulating such objects, to cut off dynamic effects

and inertia contributes, that may increase the uncertainty of the predicted behaviour. Hence in this work the simulation phase aims to simulate the cable around points of equilibrium, managing quasi-static configurations.

The cable shapes provided by the optimization step described in Chapter 4 are processed with a static simulation: each shape ξ_o, ξ_1, $\xi_1 \cdots \xi_S$, ξ_f is simulated keeping the associated planned clips in their fixed position, in this way the cable is free to stabilize under the gravity effect, leading the shapes to a stabilized configuration. As described in Figure 5.2, the clips provided are stationary and the final shape of the simulation time history is the stabilized one.

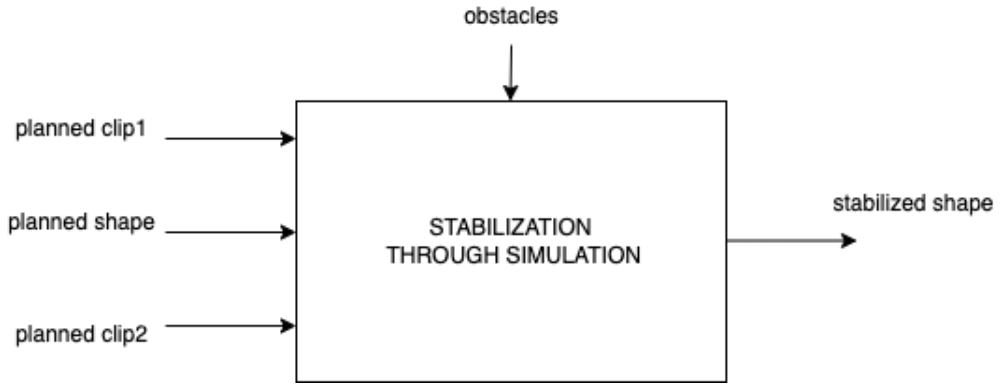


Figure 5.2: Static simulation of a planned shape.

The static simulations of the intermediate shapes are used for two purposes: A stability check for the target pose, and for obstacles avoidance along the planned path. In particular:

- For the target shape a check is necessary, to inspect if the actual desired configuration is a stable one. For this reason the static simulation is used to stabilize the shapes under the gravity effect, and a comparison with the planned one is carried on. This check is fundamental to validate the feasibility of the final user defined configuration.
- The stable configuration for each planned intermediate shape is also exploited to inspect collisions with known obstacles in the environment. The collided shapes are then re-defined, performing obstacle avoidance, and the optimization procedure is recomputed, in order to change the path, as shown in Figure 5.3.

Section 5.4 details the procedure applied to the intermediate shapes, while Section 5.3 deals with the stability check executed to the user defined final shape, that is the first check that must be executed to verify that the final configuration is feasible.

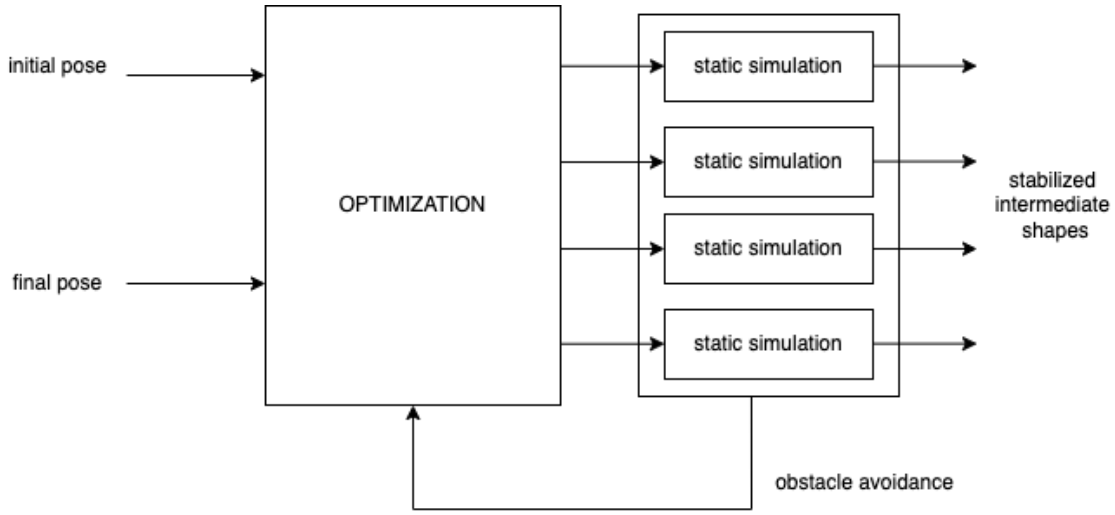


Figure 5.3: Optimization and simulation mechanism for intermediate shapes.

5.3. Stability check for target pose

To avoid creating a ill-posed manipulation it is necessary to perform a detailed inspection of the desired pose ξ_f. Such shape is processed by the static simulation procedure, obtaining the stabilized one $\xi_{f,stab}$.

The stability analysis is provided by computing the distances between relative mass-points of the ξ_f and $\xi_{f,stab}$: high distances mean instability of the desired shape, involving large amount of motions during the stabilization phase. In such cases some warnings to the user are provided, in order to re-design the target pose for the manipulation at hand, as described in Figure 5.4. In particular the warning is provided when the maximum distance between relative mass-points is greater than an user-defined threshold T_{cf}.

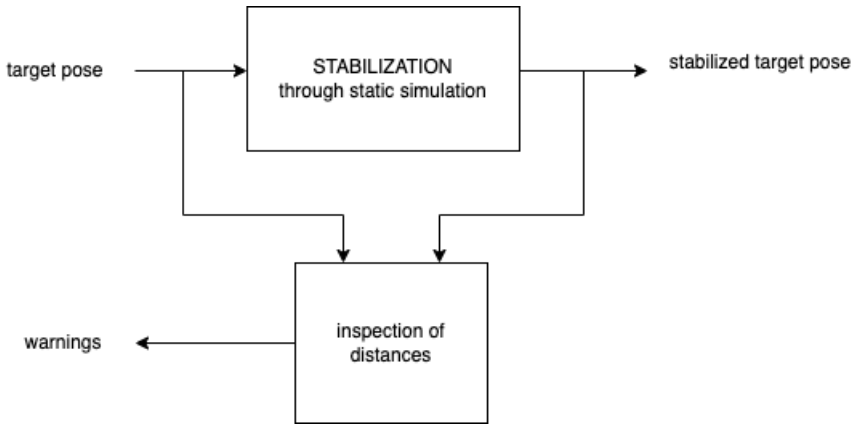


Figure 5.4: Stability check procedure for the target pose.

In Figure 5.5 an example of target pose stabilization is shown, for an Ethernet cable of initial length $l_0 = 0.57m$, $d = 6mm$ and Young's Modulus $E = 7.5 \cdot 10^6 Pa$. The stabilized shape is deformed due to the gravity effect.

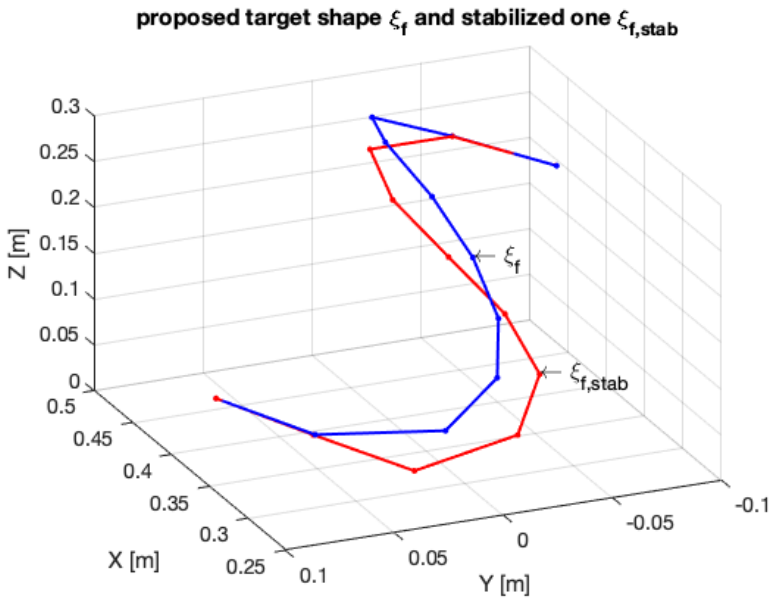


Figure 5.5: Proposed target shape ξ_f and stabilized one $\xi_{f,stab}$.

The distances are shown in Figure 5.6: a large motion is experienced in the middle of the cable, reaching up $6cm$ of magnitude. This behaviour is due to the low stiffness of the cable, because the internal forces produced in the shape are not enough to compensate the gravity effects.

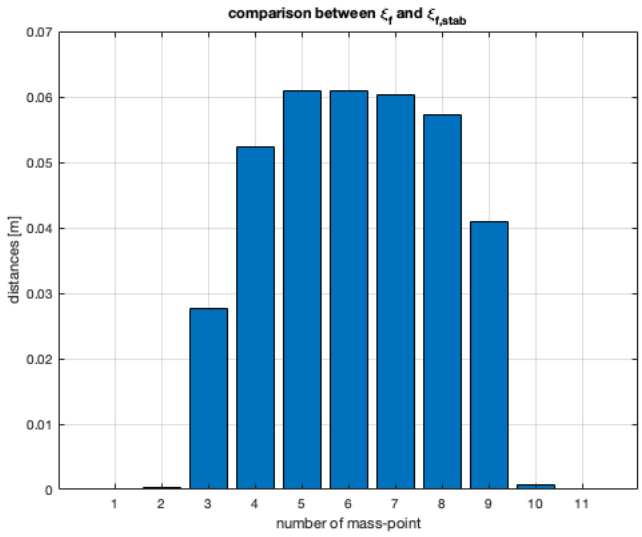


Figure 5.6: Distances between relative mass-points along the two shapes in Figure 5.5.

An other example in which the cable stabilizes under gravity effects is shown in Figure 5.7 and 5.8.

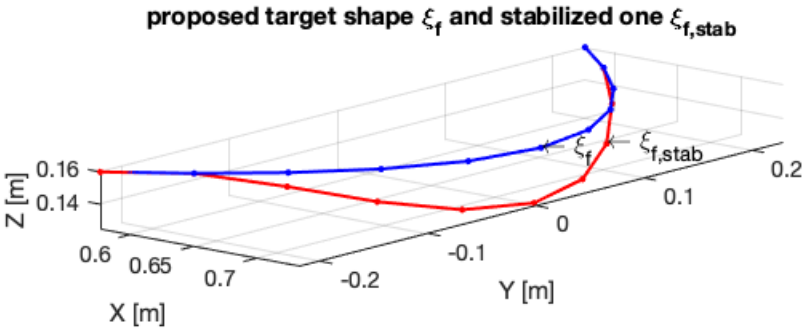


Figure 5.7: Proposed target shape and ξ_f and stabilized one $\xi_{f,stab}$.

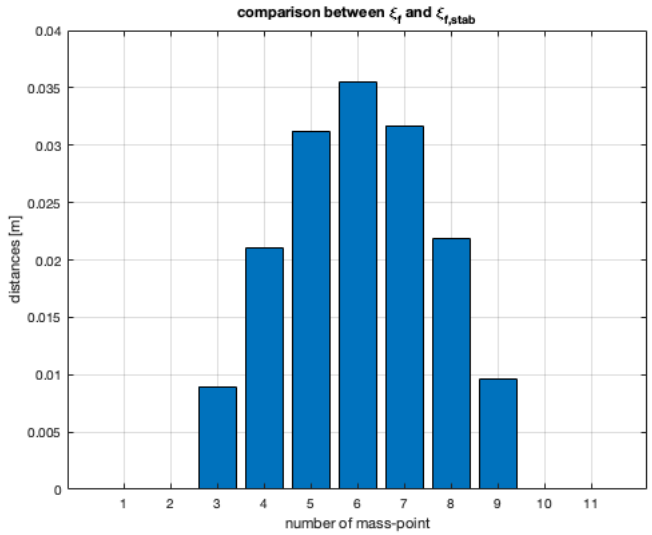


Figure 5.8: Distances between relative mass-points along the two shapes in Figure 5.7.

In Figure 5.9 and 5.10 a case is shown in which the provided target shape ξ_f and the stabilized one $\xi_{f,stab}$ are almost equal. The involved DLO is an hose with length $l_0 = 0.57m$, $d = 6mm$, mass $m = 10g$ and Young's modulus $E = 1 \cdot 10^8 Pa$. The stiffness of the cable allows to compensate the gravity effects, indeed the distances between the two

mass-points are very small (less than $1mm$), since the internal forces are minimized by the *physical minimization* procedure, leading to a stable target shape.

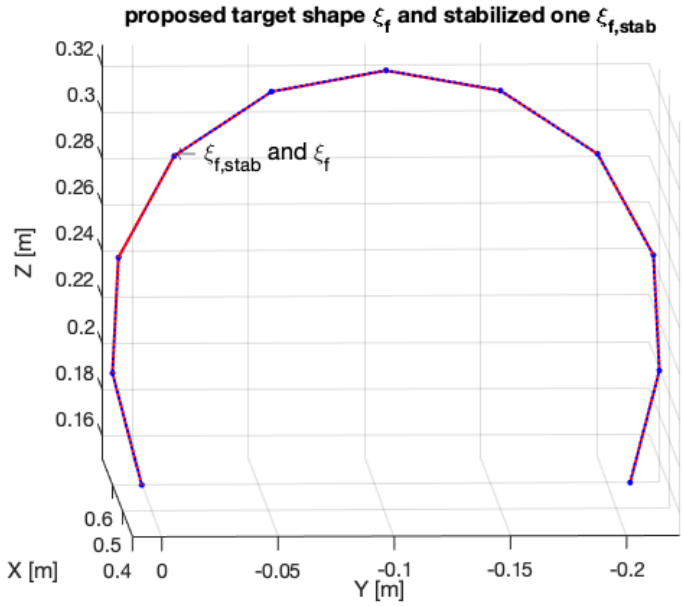


Figure 5.9: Proposed target shape ξ_f and stabilized one $\xi_{f,stab}$.

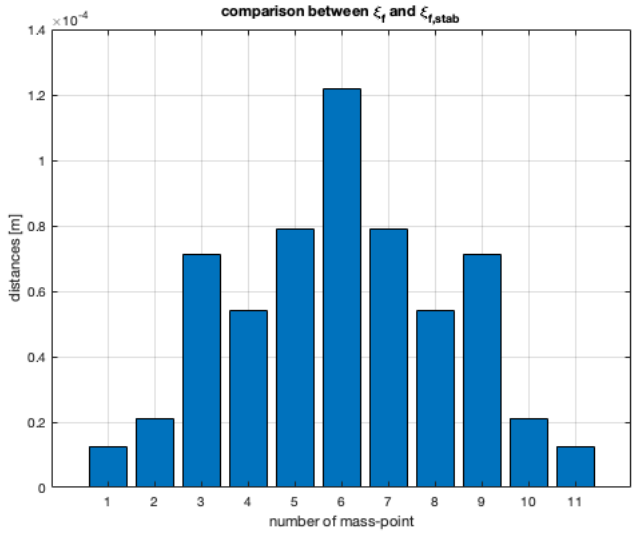


Figure 5.10: Distances between relative mass-points along the two shapes in Figure 5.9.

While on the final target pose a stabilization analysis is enough, the static simulation environment has to be applied also on intermediate shapes, in order to inspect gravity effects on the planned shapes and collision with obstacles, as previously mentioned.

5.4. Path stabilization and obstacle avoidance

This simulation step aims to inspect the cable configurations along the path, managing to avoid collisions with obstacles in the environment. As described in the previous section, the cable may undergo to gravity effects, providing some deformations. This phase computes such deformation on each intermediate planned shape ξ_i with $i \in [1,\ S]$, where S is the number of intermediate shapes. Then a collision check is computed, referring to obstacles modeled in the simulation environment. Finally a collision avoidance strategy and a re-planning through optimization procedure are implemented, leading to a collision free path, considering the cable during the manipulation.

5.4.1. Obstacle modeling

In assembly operations such as cable wiring operations, the obstacles are supposed to be lying on the working area. An example can be a component placed in the middle of the working table. For this reason obstacles are modeled as some prohibitive areas attached on the ground, as shown in Figure 5.11.

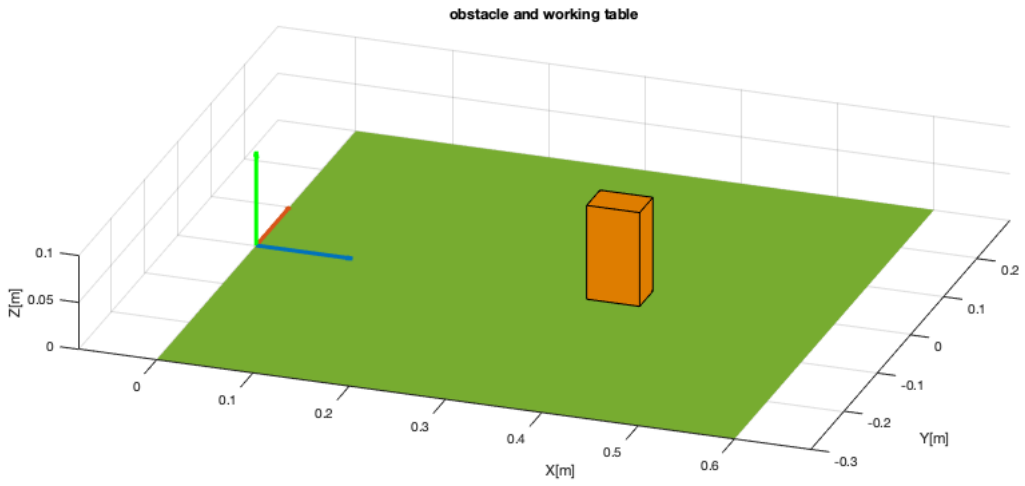


Figure 5.11: The obstacle is modeled as a volume resting on the table.

5.4.2. Gravity deformation of intermediate shapes and collision detection

Once the obstacle is modeled, a collision check between the intermediate shapes of the path is necessary. To this aim each intermediate shape is processed by a static simulation, then

the collision with obstacles is inspected considering the stabilized shapes. As described in section 5.3, often shapes stabilized under gravity effect provide a downward deformation. For this reason they are used for collision detection purposes, providing a worst-case situation, leading to a safer path correction.

In Figure 5.12 an example of stabilized total path is provided: each intermediate shape is stabilized by the static simulation.

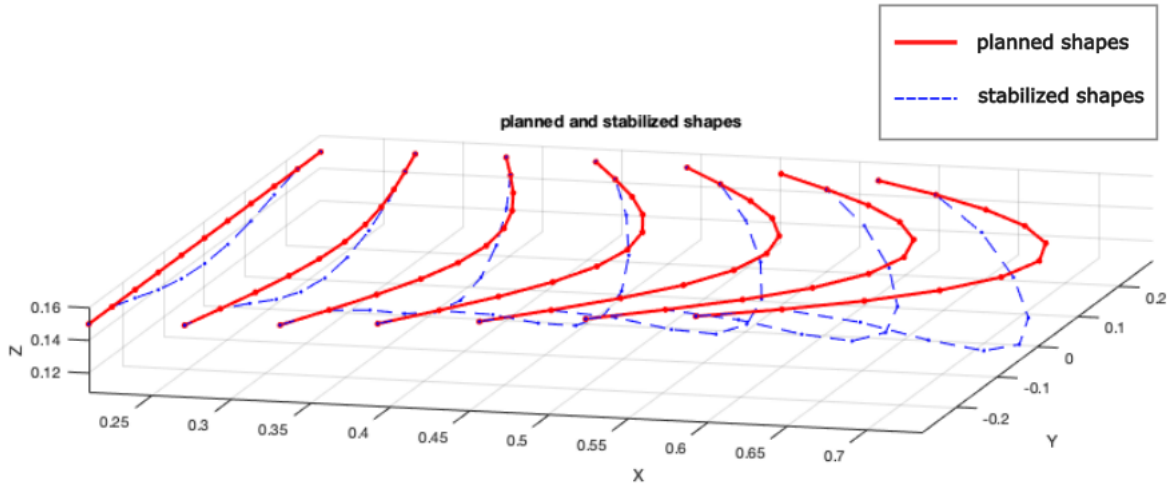


Figure 5.12: Stabilized path for an USB cable with $l_0 = 0.5m$, $d = 5mm$, $m = 10g$ and equivalent Young's modulus $E = 2.5 \cdot 10^6 Pa$.

It can be noticed that a deformation due to gravity effects is not a problem, since shapes are stabilized and minimal energy ones. Hence even if the manipulated cable is deformed, it is not expected to be in stressed configurations. However, problems arise when a planned shape (or a stabilized one) collides with an obstacle. For this reason, it is necessary to spot and correct possible collisions of the manipulated cable with the environment.

The collision detection is performed by comparing the modeled obstacles and each intermediate stabilized shape, as shown in Figure 5.13. In particular, mass-points positions are verified not to be in the prohibitive volume: if this holds, a flag is assigned to the relative shape, and the maximum collision depth z_{depth} along z-axis is computed, considering all the mass-points of the stabilized cable shape at hand.
The maximum collision depth z_{depth} is the maximum depth for which a mass-point of a shape lies in the space occupied by the obstacle, along the z-axis, as shown in Figure 5.14. It is important to notice that more than one mass-point may undergo to a collision, so to consider the worst-case is necessary, inspecting all the mass-points composing the shape.

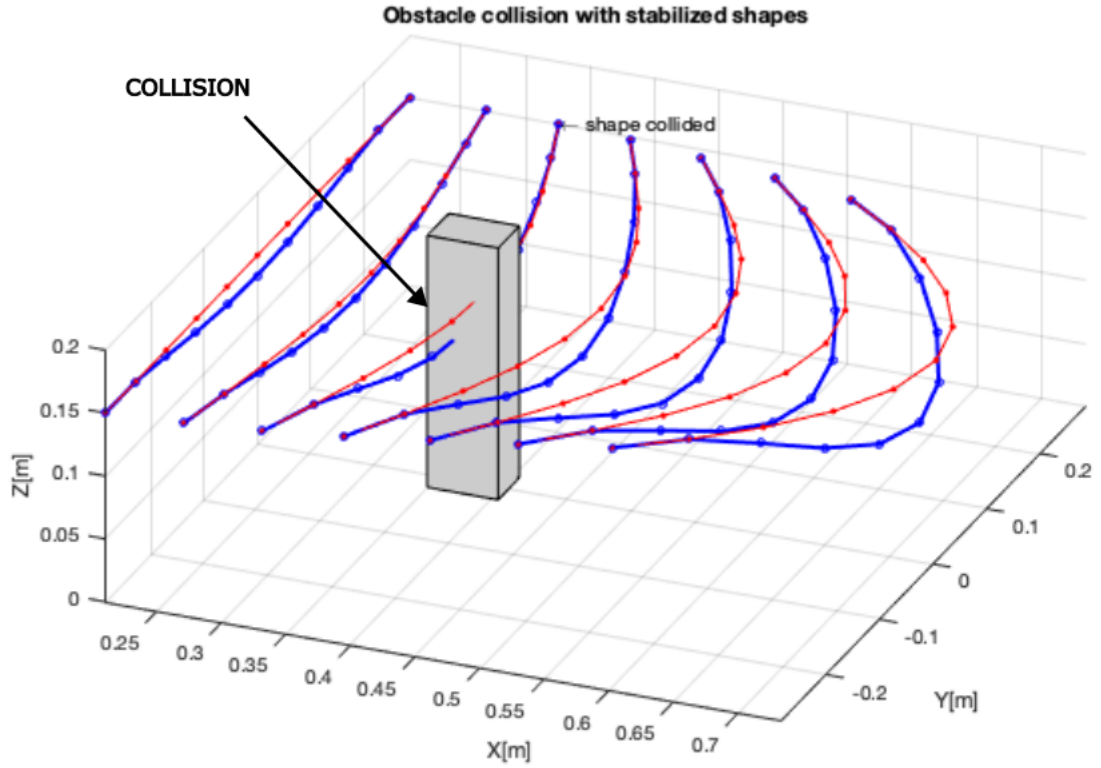


Figure 5.13: Example of collision detection with obstacle and an intermediate shape.

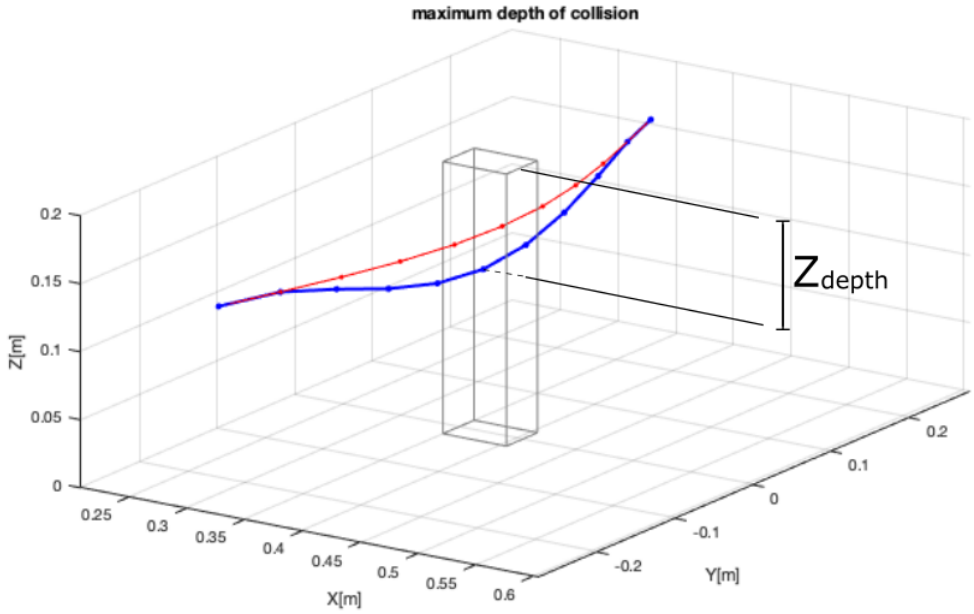


Figure 5.14: Maximum depth of collision computation considering the obstacle and the stabilized shape, focusing on the example of Figure 5.13.

5.4.3. Collision avoidance and re-planning through optimization

The strategy used to avoid obstacles is effective and intuitive: the collided shapes are subjected to a translation along the z-axis. The eight of each mass-point of the collided shape is increased by a value that is equal to the maximum collision depth z_{depth} plus a safety offset z_{safety}, as shown in Figure 5.15. The safety offset is an user defined value that can be tuned depending on the case of interest, in order to face uncertainties in the model of the deformable linear object and in the model of the obstacle.

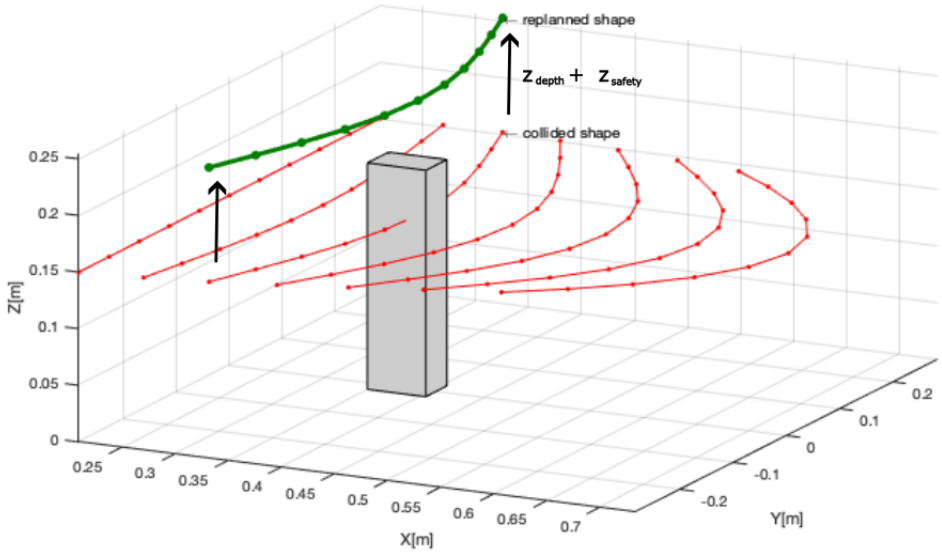


Figure 5.15: Re-planning of the collided shape focusing on the example of Figure 5.13. The safety offset is tuned to be $z_{safety} = 2cm$.

Once the collided shapes have been re-planned, a re-plan of the entire path is necessary, to integrate the obstacle avoidance operation into an optimal and smooth motion. To this aim the optimization procedure described in Chapter 4 is exploited, with the only difference that the paths to be planned are more than one, and they are connecting the re-planned shape with the initial configuration and with the final one. Two set up for two optimization procedures are designed, as described in Figure 5.16:

- The first set up aims to plan the path from the initial shape to the first re-planned shape;
- The second set up aims to plan the path from the last re-planned shape to the target one.

The number of intermediate shapes of each re-optimization procedure is automatically set in order to achieve a final number of shapes that match with the original one.

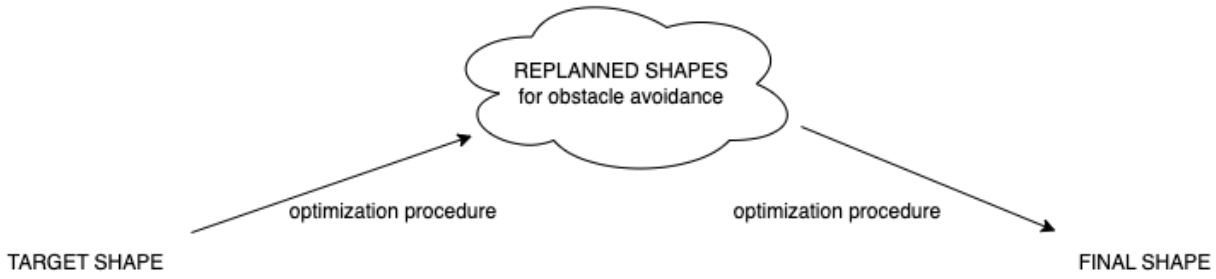


Figure 5.16: Re-optimization scheme after collision avoidance.

The optimization procedure, composed by the basic geometrical optimization, advanced geometrical optimization and physical minimization, results in a number of intermediate shapes, that must be linked in the total path, to obtain the collision-free planned path (Figure 5.17).

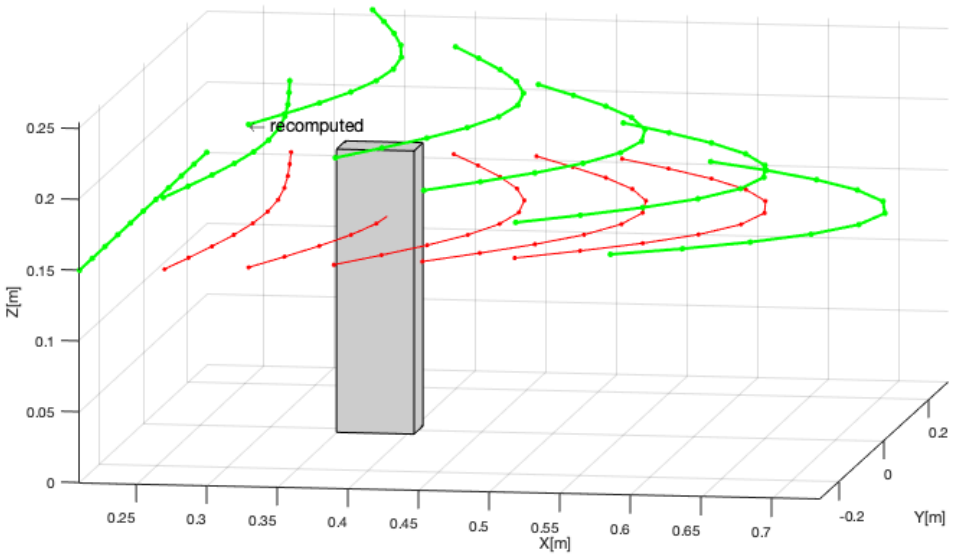


Figure 5.17: Total path with obstacle avoidance.

As described in section 4.6, after the planning phase for the cable, it is possible to compute the gripper poses of the robot from the planned shapes of the cable, and to create the dual arm manipulation trajectories.

6 | Identification of the stiffness of deformable linear objects

In an industrial framework several kinds of cables need to be manipulated, with a wide variety of properties such as length, composing material and so on. Those properties affect the behaviour of the cable under manipulation, for example leading to different poses or different internal forces for an equal manipulation, depending on the cable at hand.

The variation of forces and geometries involved in the manipulation is a critical aspect for the optimization and the simulation phases, for this reason it's necessary to somehow formalize the concept of "cable stiffness", in order to calculate it and then behave accordingly.

The answer to this problem is provided by the mass-spring model: as said in Chapter 3 a deformable linear object can be modeled as a series of mass-points connected with different kinds of springs: axial springs, bending springs and torsional springs. Intuitively it's possible to rely on the definition of "stiffness of a spring" and extend the concept to the whole cable.
The stiffness of a generic spring is denoted by the spring constant k, and exploiting the Hook's law, the forces and the geometric deformations are related such that:

$$F = -k\Delta x \tag{6.1}$$

where F is the generated force and Δx is the deformation.
In the mass-spring model, three spring coefficients are involved: k_s, k_b and k_t, it hods [10] that:

$$\begin{aligned} k_s &= \frac{EA}{l_i} \\ k_b &= \frac{3EI}{l_i} \\ k_t &= \frac{GI_p}{l_i} \end{aligned} \tag{6.2}$$

where E is the Young's modulus associated to the cable material, I is the moment of

inertia of a link: $I = \pi d^4/64$, l_i is the length of the single link: $l_i = l/n$, G is the shear modulus of the material: $G = E/2(1 + v)$ (where v is the Poisson ratio), and I_p is the polar moment of inertia: $I_p = \pi d^4/32$.

Notice that the three coefficients (and hence the "cable stiffness") depend mainly on three aspects:

- the length of the cable l_0;
- the diameter of the cable d;
- the Young's modulus of the cable E.

Although the first and the second terms can be easily measured on the cable, for the Young's modulus measurement there could be some issues.

The first problem is related to the experiment to be performed for the measurement of E: the cable should undergo a tensile test, in order to explore the Hook's law and obtain the right parameter, but very sensitive and accurate machines are needed, ending in a very expensive procedure.
The second problem is related to the composites cables: often in industrial frameworks, cables composed by more than one material are manipulated, such as electrical wires (Figure 6.1), in which a copper core is coated with a layer of plastic material such as PVC, or ethylene. In those cases a tensile test could not be effective, since some layers should behave in different ways with respect to others (for example undergoing a plastic deformation very soon) leading to an untrue measurement. Moreover, since such cables are composed by different materials, their tabulated value of E can't be used, being the cable a composite one and involving more than one material.

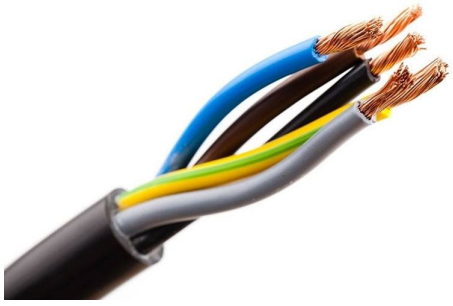


Figure 6.1: Examples of electrical composed wires. A single cable may contain different material layers and multiple wires.

For this reason our work aims to inspect the Young's modulus E with an identification algorithm designed to obtain the right value for our mass-spring model, overcoming issues

related to the tensile test, and managing also to estimate the equivalent Young's modulus for composite cables.

6.1. Young's modulus identification

The general pipeline for Young's modulus identification procedure is reported in Figure 6.2 and it consists in performing a manipulation of a cable in the real world and in the simulation environment. The final poses are compared iteratively repeating the simulation by varying E, and a least square algorithm is able to estimate the true Young's modulus.

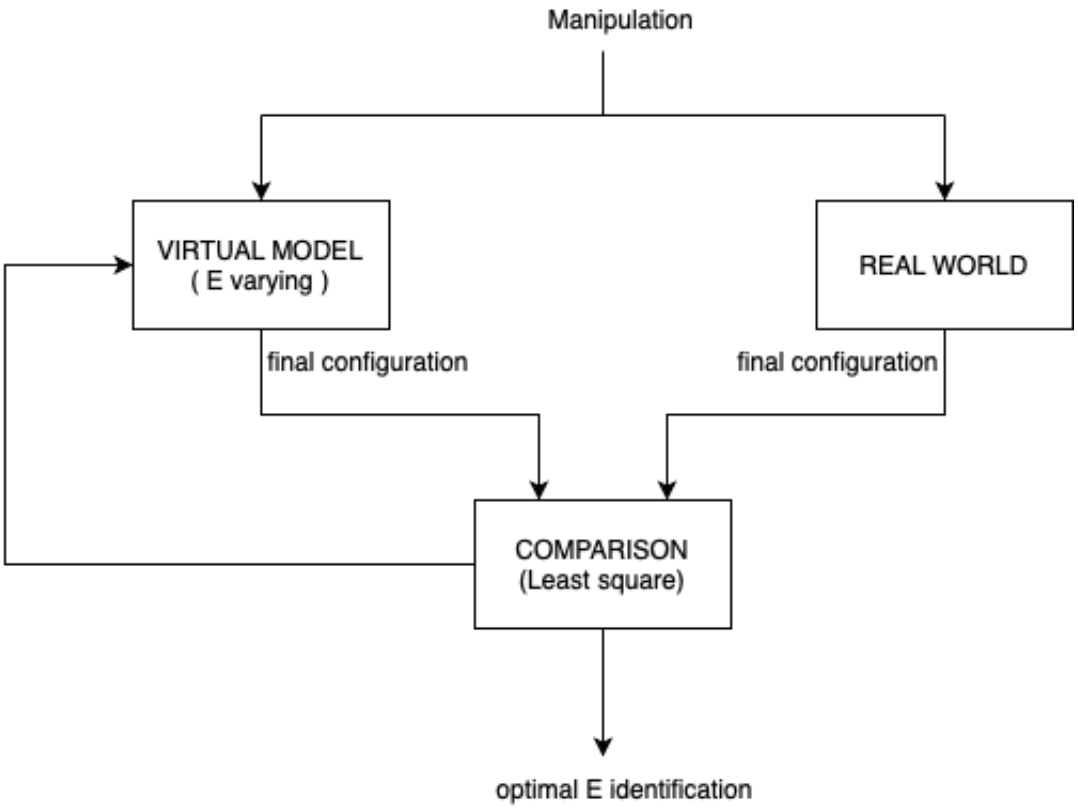


Figure 6.2: Young's modulus identification pipeline.

Experiment design and data collection

The manipulation of the cable should be as informative as possible: the stiffness coefficients are related to axial, torsion and bending deformations, since axial deformations and torsional behaviours are difficult to be inspected in a real framework, the useful information relies on bending geometry of the cable. That's why the experiment manipulation should provide a bent final pose, as in Figure 6.3.

Figure 6.3: Final pose of the designed experiment.

The data collection consists into storing the 3D cable shape, and discretize it in a number $n + 1$ of points, as shown in Figure 6.4 , in order to be easily compared with the simulation outputs. The data collection can be performed with automatic recognition and discretization of a cable, through the Ariadne algorithm [4], and a vision system described in section 7.1. In particular during the design of the algorithm the cable configurations have been inspected and stored through some graph paper, in order to have an as accurate as possible measure. The error introduced by the Ariadne algorithm, comparing the previous measures, is small, and hence in a second time the cable has been inspected through the automatic algorithm for cable segmentation.

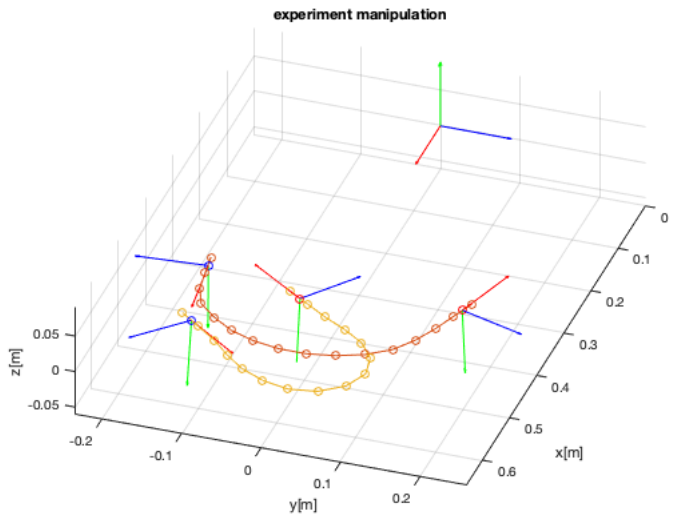


Figure 6.4: Data collection of the initial and final pose in the identification manipulation.

Least square algorithm

The identification is a non linear, constrained, simulation-based least square algorithm: the manipulation is simulated iteratively, varying the Young's Modulus into a certain range of values, and comparing the simulation output with the collected datas to get the errors.

The decision variable is:

$$E \in [E_{min}, E_{max}] \tag{6.3}$$

The cost function to be minimized is:

$$\min_{E} ||\xi_{data,f} - \xi_{sim,f}||^2 \tag{6.4}$$

where $\xi_{data,f}$ is the final shape configuration stored through the manipulation experiment, and $\xi_{sim,f}$ is the final shape provided by a simulation. E_{min} and E_{max} are some values that should be used in order to limit the research, avoiding unfeasible values of E. An example can be $E_{min} = 1 \cdot 10^5[Pa]$ and $E_{max} = 1 \cdot 10^{10}[Pa]$.

Time discretization for simulation

Since the algorithm has to perform a high number of simulations, a refined strategy has been developed, to dicretize the time step of the simulation:

As described in section 3.8 the time step value $\Delta\tau$ is computed by taking in account about the axial stiffness coefficient k_s:

$$\Delta\tau = \sqrt{\frac{m_i l_i}{k_s}} \tag{6.5}$$

The new strategy consists in creating a fictitious Young's Modulus $E_{discretization}$ considered only for the time discretization computation, since k_s is proportional on it. High values of E will provide a thicker dicretization, instead smaller ones will provide larger time steps.

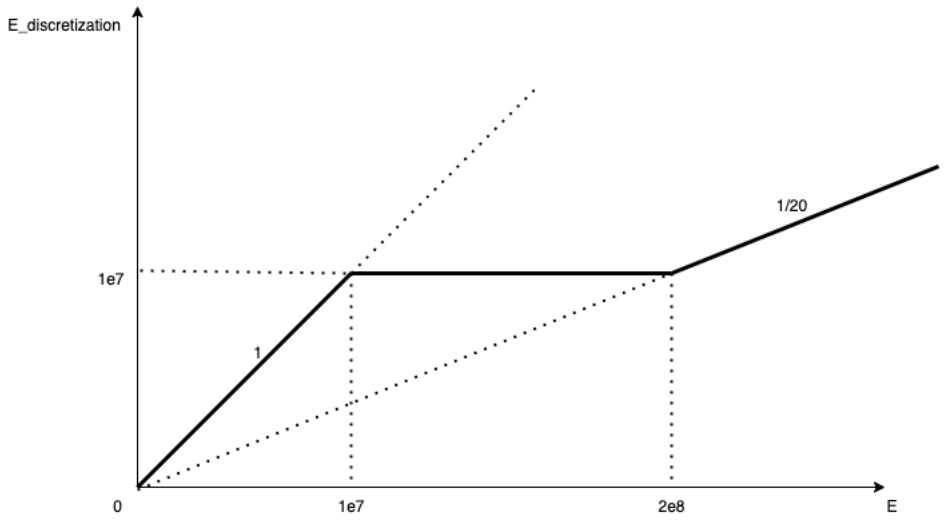


Figure 6.5: Relationship between the stiffness considered for the time discretization and the real one.

As shown in Figure 6.5 for small values of E the time discretization is not changed, instead for higher values the time steps are imposed to be larger.

Sensitive analysis on mass-points discretization

Another parameter to be tuned in the simulation for the algorithm is the number of links n and mass-points $n+1$ of the cable. In Figure 6.6 are shown the final pose of three different simulations for a given value of E, with different cable discretization, compared with the collected data. By increasing the number of mass-points, the shape of the cable is smoother, falling more gradually and leading to a more realistic behaviour. On the other side by inspecting the errors used in the algorithm (the distances between correspondent mass-points provided in Figure 6.7) not an huge variation is observed. Moreover a thicker discretization of the cable may lead to a heavy computational load and a time-consuming procedure. For this reason we prefer to set a restricted number of links, for example $n = 10$.

In Figure 6.6 and 6.7 some examples are shown, providing the errors computed by the algorithm for different discretizations of the cable $n = 10,\ 15,\ 20$, when the data are collected with a Hose tube with Young's modulus $E = 2 \cdot 10^9\,[Pa]$ and a simulation is carried out with $E = 1 \cdot 10^8\,[Pa]$.

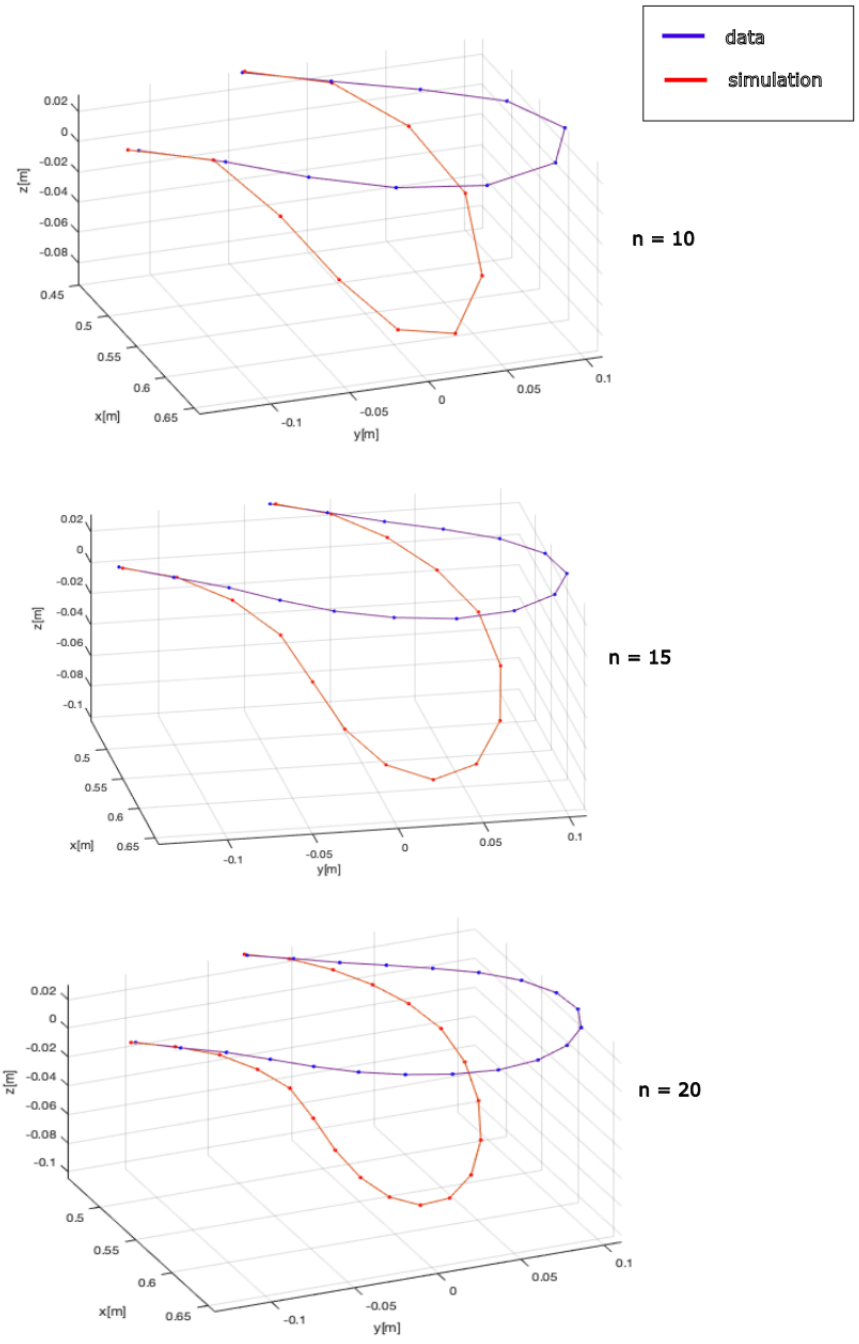


Figure 6.6: Manipulation simulations with different mass-points discretization.

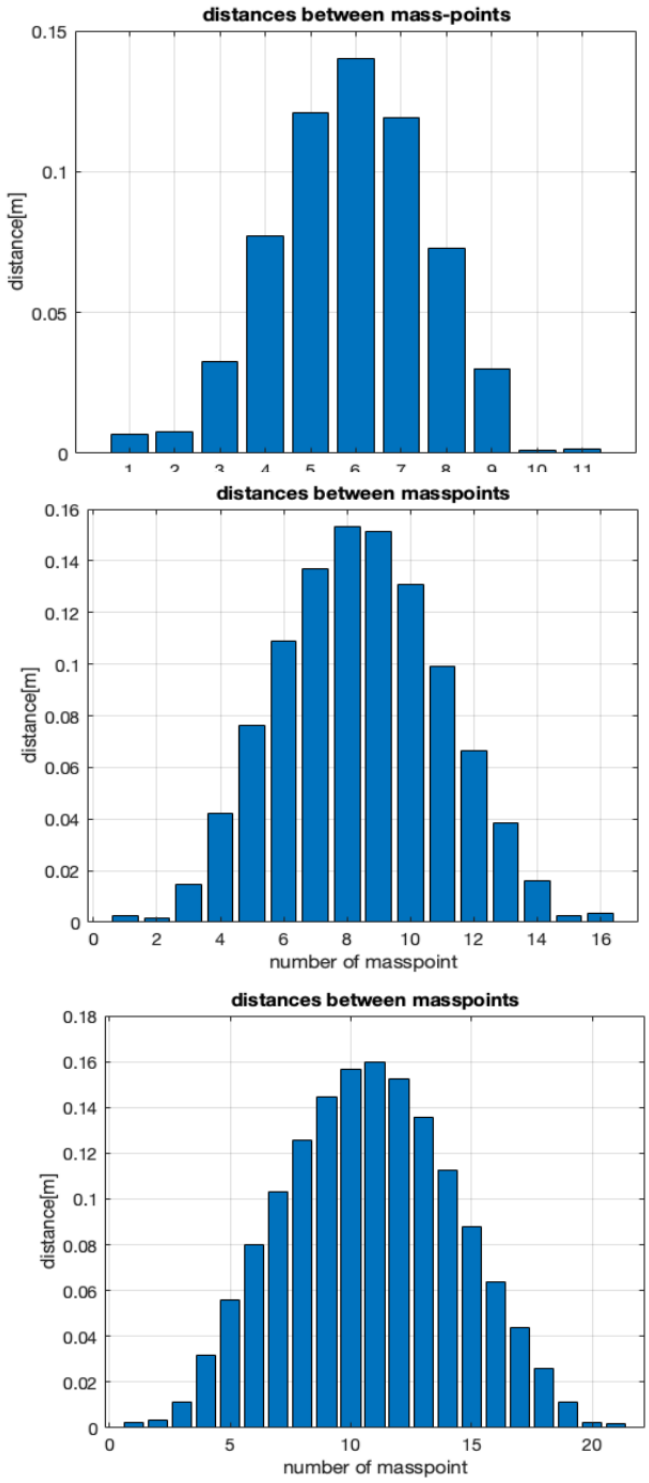
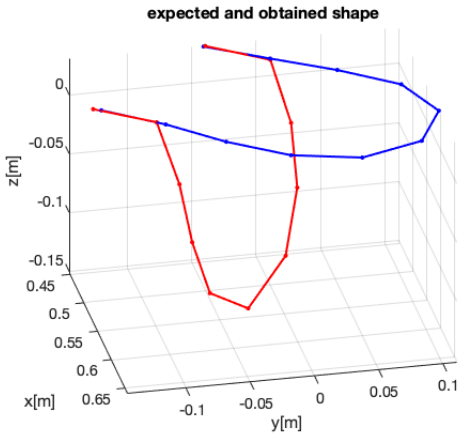


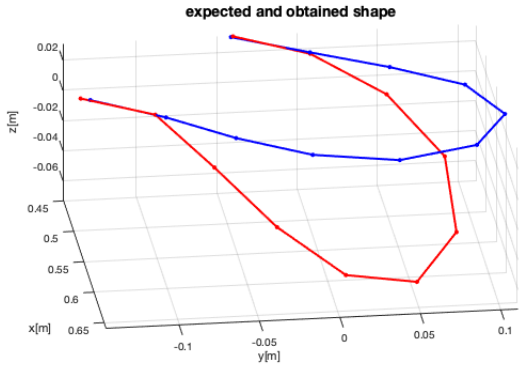
Figure 6.7: Errors computation with different mass-points discretization.

The algorithm will start from an initial proposed value of E, and then will compute different simulations varying the Young's modulus, as shown in Figure 6.8. The algorithm explores different values, driving the research toward the right value (in this example is

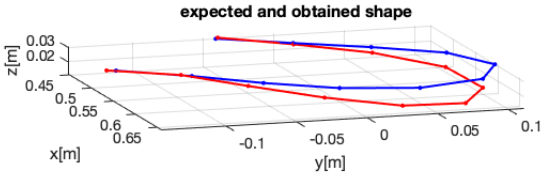
$E_{real} = 2 \cdot 10^9[Pa]$).



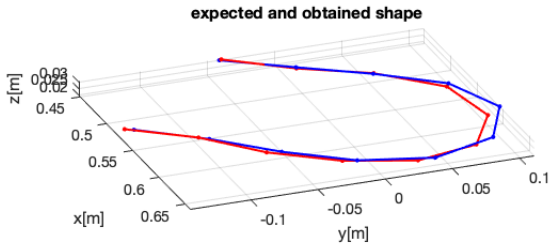
(a) $E_{sim} = 1e6\,Pa.$



(b) $E_{sim} = 1e7\,Pa.$



(c) $E_{sim} = 1e9\,Pa.$



(d) $E_{sim} = 4e9\,Pa.$

Figure 6.8: Example of simulations with different values of E.

6.2. Data driven optimization tuning

When the information about the stiffness of the cable is obtained, it's possible to use it in order to adapt the optimization planning algorithm and the static simulation check algorithm.

- For the static simulation check algorithm, the Young's modulus is necessary for the set up. Depending on the provided stiffness, the cable will vary the gravity deformations, changing the stabilized configuration, and possibly colliding in different ways with obstacles in the environment. For this reason an adaptation on parameter E is very important, leading to big variations of the planned path.
- For the optimization planning step the concept is more subtle, but fundamental: a change on the stiffness of the cable will introduce a change on the relationships between forces and geometrical deformations. This variation may bring an unbalance on the cost functions in the problem, leading to huge variations of the convergence points of the algorithm. For this reason an adaptation of the optimization weights based on parameter E is necessary.

While in the static simulation step to set in the virtual environment the right value of E is enough, for the optimization step the solution is not such trivial: in order to compensate an unbalance of the cost function it's necessary to act on the weights of the different terms involved. For this reason we can talk about a data-driven approach for the tuning of the optimization terms.

Ranges of stiffness

A number of families in which to allocate cables depending on stiffness behaviours are provided. In particular three categories are created: soft-cables, medium-cables, and stiff-cables.

The first analysis was carried out by inspecting the k_s parameter associated to the cable. Equation 6.2 shows as k_s parameter depends on the Young's modulus E, the length l_0 and the diameter d of the cable. Such analysis leads to a huge number of cases, depending on the varied parameter, introducing an excessive complexity to the problem. For this reason the analysis has been limited, considering the environment concerned with our work: the working area and the robot geometries. For this purpose the length of the cable is supposed to be around $0.5\,m$ with a diameter about $6\,mm$.

In this way the stiffness of the cable can be determined uniquely by the Young's modulus.

The three families are then obtained dividing the range of values as shown in Figure 6.9.

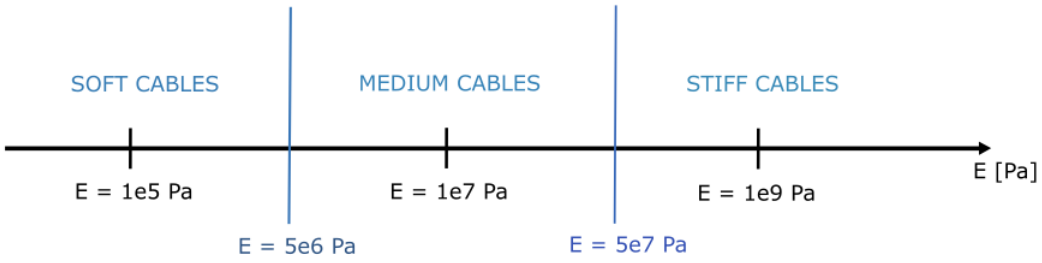


Figure 6.9: Ranges division based on the Young's modulus of the cable.

Optimization tuning

The tuning of the cost function is provided by inspecting the optimization behaviours in the different identified families. In equations 6.6, 6.7 and 6.8 are reported the cost functions involved in the optimization, in order to highlight the weights to be tuned.

Basic geometrical cost function:

$$\min_{\xi_1,\xi_2,\cdots,\xi_S} \sum_{i=1}^{S} w_{i,b}||\gamma_i(\xi_i,\xi_{i-1})||^2 \quad + w_{f,b}||\gamma_f(\xi_o,\xi_1,\cdots,\xi_f)||^2 \tag{6.6}$$

Advanced geometrical cost function:

$$\begin{aligned}\min_{\xi_1,\xi_2,\cdots,\xi_S} \sum_{i=1}^{S} w_{i,a}||\gamma_i(\xi_i,\xi_{i-1})||^2 \quad + w_{f,a}||\gamma_f(\xi_o,\xi_1,\cdots,\xi_f)||^2 \\ + \sum_{i=1}^{S} w_\delta||\delta_i(\xi_i)||^2 + w_{clip}\frac{1}{\Delta_{clip,i}} + w_\beta||\beta_i(\xi_i)-\beta_{o,i}||^2\end{aligned} \tag{6.7}$$

Physical minimization cost function:

$$min_{\xi_{min}} = w_\sigma||\sigma||^2 + w_b||F_b||^2 + w_s||F_s||^2 \tag{6.8}$$

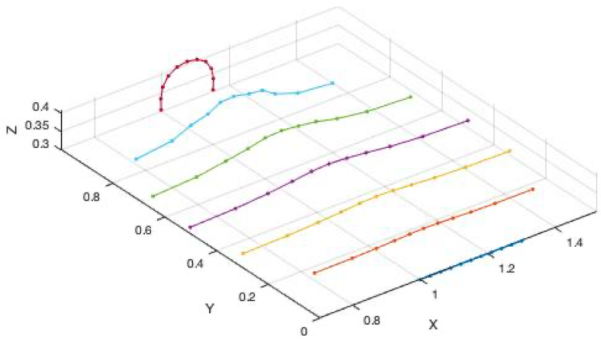
The tuning of the *Basic geometrical* optimization and the *Advanced geometrical* optimization is unchanged in the three families: it relies only on the geometrical side of the problem, and since the length and the diameter of the cable and the working area are fixed it is not necessary a variation. The tuning parameters are shown in Table 6.1.

Geometrical optimization weights tuning

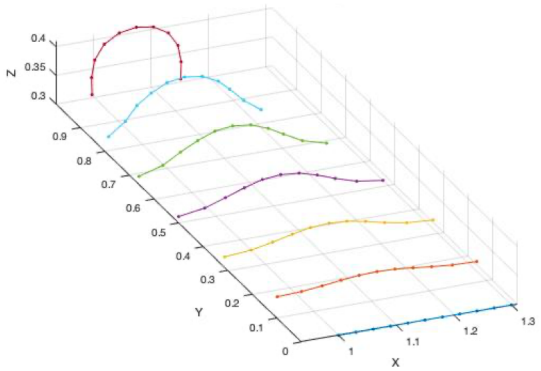
	$w_{i,b}$	$w_{f,b}$	$w_{i,a}$	$w_{f,a}$	w_δ	w_{clip}	w_β
all-stiffness	1	1	20	20	10	0.008	40

Table 6.1: Tuning of the weights in the *basic geometrical* cost function and in the *advanced geometrical cost function*.

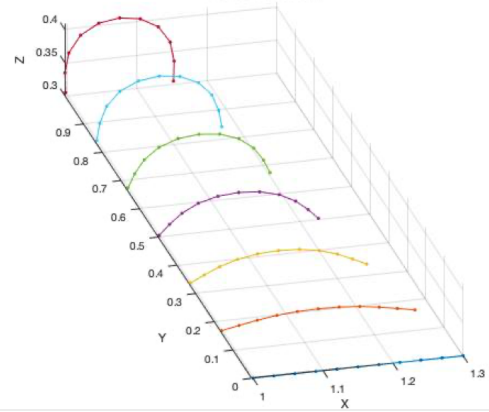
As described in section 4.4, the *advanced geometrical* optimization aims to stretch the cable along the shortest geometrical path without introducing excessive axial deformations. The stretching component is managed in the term weighted by w_{clip}, its counterpart is instead weighted by w_δ. Those terms are used in order to balance the geometrical axial deformation of the cable. An example of wrong tuning can be an excessive small w_δ or an excessive big w_{clip} that may lead to a over-stretched cable. In Figure 6.10 there is an example of different results with different tuning of the parameters. In Figure 6.10a the cable is stretched a lot, hence in Figure 6.10b the w_{clip} term has been decreased: the stretching behaviour is balanced, however the curvature is not very good. For this reason in Figure 6.10c the weight involving the curvature of the path w_β has been increased, leading to the final right tuning.



(a) $w_{clip} = 0.1$, $w_\delta = 10$, $w_\beta = 4$.



(b) $w_{clip} = 0.008$, $w_\delta = 10$, $w_\beta = 4$.



(c) $w_{clip} = 0.008$, $w_\delta = 10$, $w_\beta = 40$.

Figure 6.10: Different geometrical planning for different tuning of the advanced geometrical cost.

In the *physical minimization*, a variation of the weight costs is necessary, because the relationship between geometrical deformations and internal forces is exploited in the optimization. The tuning is reported in table 6.2.

Physical optimization weights tuning

	w_σ	w_b	w_s
medium-cables	10	100	1
soft-cables	10	1000	10
stiff-cables	0.1	100	1

Table 6.2: Tuning of the weights in the *physical minimization* cost function, for cables of different stiffness.

Inspecting the tuning of the medium-cable case it can be noticed that a grater importance has been posed on the bending forces. The reason is that for a DLO in general it's true that $k_s >> k_b$: a small axial deformation of the cable will produce a much higher force compared to a bending one. For this reason the magnitude of the bending forces has been increased with respect to the axial ones.

Once the tuning of the medium-cables is provided, a variation of the weights brings to the tuning for soft-cables and stiff-cables.

- For soft-cables the magnitude of w_b and w_s has been increased about 10 times. This because soft cables will have a low Young's modulus E, that leads to small axial stiffness k_s and bending stiffness k_b, causing the cable to produce low forces also in stressed conditions with unnatural geometries. For this reason it's necessary a cost function more focused on the forces in the cable.
- For stiff-cables the magnitude of w_σ has been decreased about 100 times. This because stiffer cables have a very high Young's modulus E, hence for quite small deformations, some very high forces are generated. Moreover it can be noticed that, by moving a very hard cable, the internal forces may lead to limitations on the range of deformation of the cable. For this reason the geometrical term w_σ has been decreased: the cable should be free to move, in order to avoid excessively stressed configurations.

Once the three families have a proper tuning, the values of the cost can vary, depending on the case of interest.

This adaptation strategy enables the planning for the manipulation of different types of cables, in the area of interest. The Young's modulus identification provides a value of the cable stiffness, that is used to set up the simulation phase and to accordingly tune the optimization algorithms, as shown in Figure 6.11.

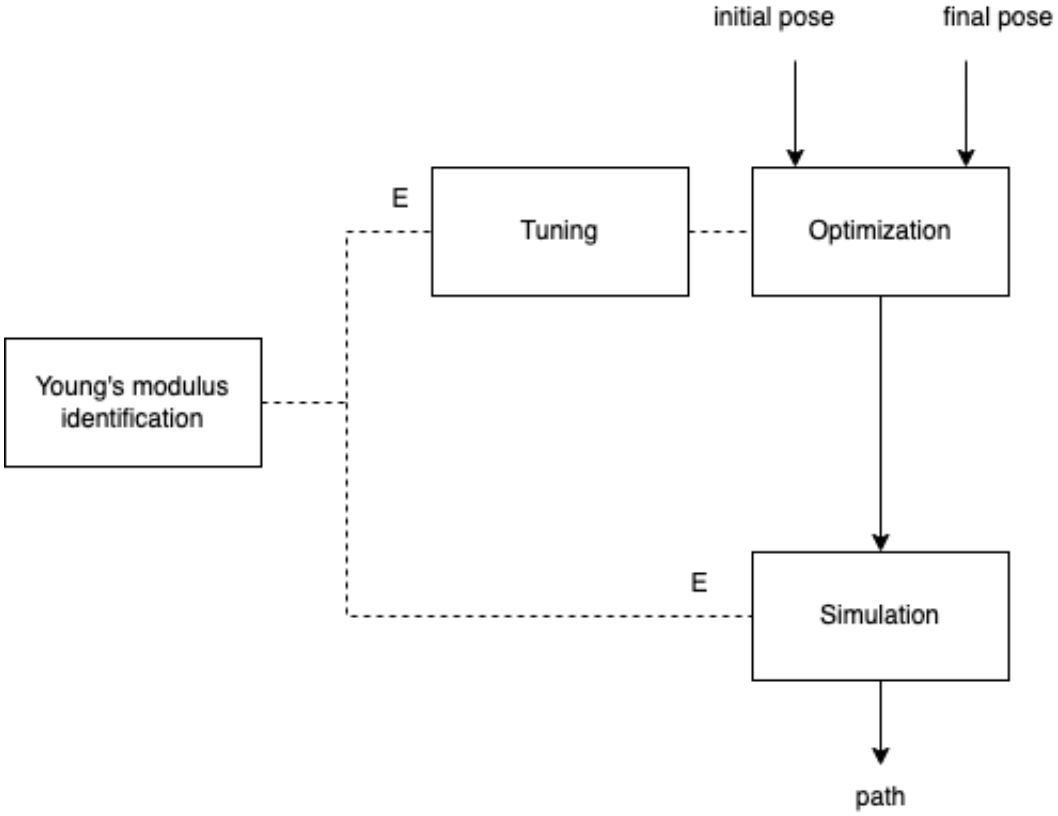


Figure 6.11: Adaptation strategy implemented in the work.

The focus provided on the cable length has been introduced in order to match the size of the problem with the size of the available area on the working table. Such limitation can be easily relaxed: with an analysis and a right tuning of the cost functions, whatever situation can be properly handled.

7 | Experimental analysis and use case

A series of experimental tests involving DLOs of different materials and mechanical properties were carried out to validate the proposed method, providing quantitative and qualitative analysis. Moreover, a use case regarding a wire harness assembly operation involving three different cables has also been considered, exploiting the proposed method.
This chapter analyses in detail the experiments performed and the set up used, including the obtained results, in terms of errors and experienced shapes for different manipulations.

7.1. Experimental setup description

The experimental setup consists into a dual arm robot that manipulates some deformable linear objects according to the proposed methodology. A vision system has been set up, to inspect the shapes provided by the manipulation.
The robot used for the experimental phase is an ABB YuMi robot, with some custom designed grippers for the grasping of the cable, allowing a dual-arm deformable linear object robotic manipulation. The camera used for the shapes inspection is a RealSense camera, mounted with an Eye-to-hand configuration: the camera is placed above the working area, as shown in Figure 7.1. The camera is used to compute the intermediate cable shapes experienced during the experimental manipulation, referred to the robot base frame. In this way it is possible to compare the planned shapes with the obtained ones.

7.1.1. Camera calibration

The camera has to be calibrated before the usage:

- The internal calibration consists into the determination of the intrinsic parameters of the camera, such as the focal length, and some additional distorsion parameters involving lens imperfections and misalignements in the optical system. Those pa-

rameters are automatically provided by the camera, in the Python environment, once the communication channel with an external PC has been initialized.

- The external calibration consists into the determination of the position and orientation of the camera with respect to an external reference frame (in the set-up it coincides with the robot base frame). This calibration procedure is carried out before starting the manipulation, and then the camera must not be moved.

The external calibration provides the homogeneous matrix to be applied to an object, in order to transform its position and orientation from the camera point of view to the robot perspective. To this aim three frames are involved:

- The camera frame, placed in the camera focus;
- The robot frame (called also base frame, because it is the main frame placed at the base of the robot);
- An additional marker frame, provided by a marker (an ArUco), placed with a known position and orientation with respect to the robot base frame.

Those three frames are linked through some homogeneous matrices, as described in Figure 7.1.

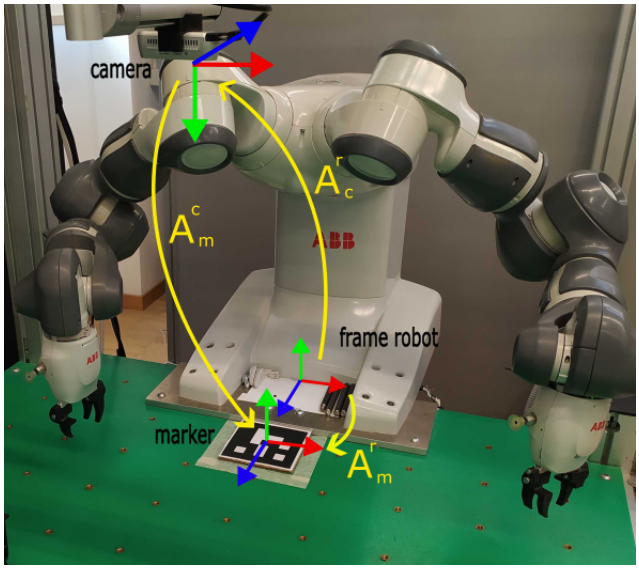


Figure 7.1: Camera frame, Robot base frame and Marker frame, linked with homogeneous matrices.

In particular $A_m^c \in SO(3)$ is the roto-translation matrix that describes the pose of the marker frame from the perspective of the camera one; $A_m^r \in SO(3)$ is the matrix that

express the pose of the marker frame from the point of view of the robot; and $A_c^r \in SO(3)$ is the matrix that provides the pose of the camera from the robot perspective.
The matrix A_c^r is the matrix computed with the extrinsic calibration, and it is used to bring, given the position of an object in the working area, the pose sensed in camera frame to the robot frame. In particular:

- The camera provides the position and the orientation of the object to be detected A_o^c, and of the AruCo marker A_m^c, referred to the camera frame.
- The position and the orientation of the marker referred to the robot base frame A_m^r is known, because it is user defined.
- The relationships between known matrices can be exploited in order to find the position of the object referred to the robot frame A_o^r.

As shown in Figure 7.2, in order to find A_o^r (the pose of the object with respect to the robot base frame) the following relationship can be exploited:

$$\begin{aligned} A_o^r &= A_m^r \cdot (A_m^c)^{-1} \cdot A_o^c \\ &= A_m^r \cdot A_c^m \cdot A_o^c \\ &= A_c^r \cdot A_o^c \end{aligned} \tag{7.1}$$

Leading to the extrinsic matrix A_c^r, used to convert A_o^c into A_o^r.

$$A_c^r = A_m^r \cdot (A_m^c)^{-1} \tag{7.2}$$

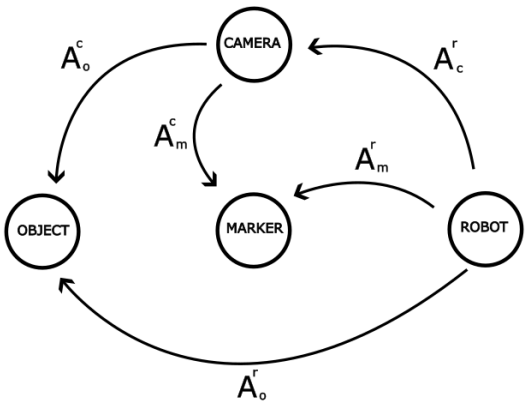


Figure 7.2: Homogeneous matrices used to provide the position of the object with respect to the robot base frame.

Once the extrinsic calibration of the camera is completed, it is possible to find the coordinates in the space with respect to the robot frame of the pixels in the image provided by the camera.

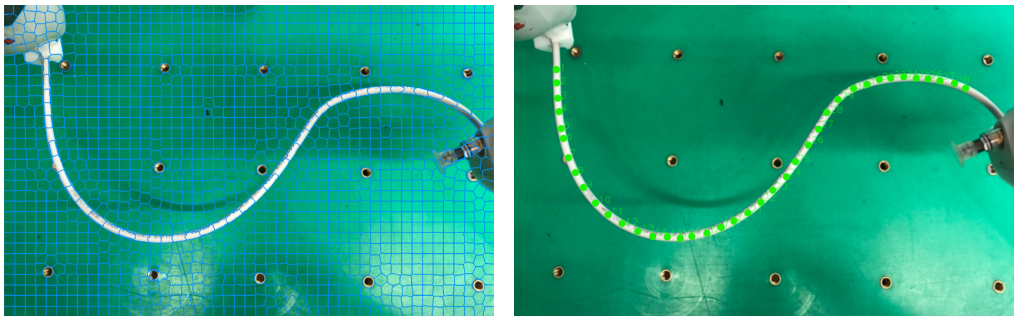
7.1.2. Deformable linear objects registration

To provide the position of a deformable linear object in the space, it is necessary to provide the coordinates of a certain number of points, distributed along the cable shape in the space. Those points are initially computed as pixel coordinates on the image taken by the camera. They are then transformed in $[m]$ coordinates by processing them with the intrinsic parameters of the camera and finally they are referred to the robot base frame, by applying the extrinsic calibration matrix:

$$\tilde{p}^r = A_c^r \cdot \tilde{p}^c \tag{7.3}$$

where $\tilde{p}^r$ is the homogeneous representation of the position referred to the robot frame, and $\tilde{p}^c$ is the homogeneous representation of the same position referred to the camera frame.

The pixel points describing the cable shape can be computed by inspecting the images provided by the camera, or by processing them with an automatic DLO segmentation algorithm: Ariadne [4]. Such algorithm exploits an over-segmentation of the source image into super-pixels (regions of the image), as shown in Figure 7.3a. Then it inspects meaningful regions (based on vision similarities, such as curvature, size, or color of the super-pixels), in order to find the cable shape, by creating a path between super-pixels, as shown in Figure 7.3b. The super-pixels corresponding to the path give the shape of the cable.



(a) Super-pixel segmentation. (b) Cable recognition.

Figure 7.3: Super-pixel segmentation and cable recognition through Ariadne [4] algorithm.

7.2. Experimental Tests

A series of experimental tests have been carried out, to compare the shapes planned on simulation by our method, and the experienced ones during the dual arm robotic manipulation.
Each planned manipulation consists into a number $S+2$ of shapes, and each shape consists into a distribution in the space of $n+1$ mass-points. On the other side a shape registered by the vision system can be discretized into $n+1$ points, in order to have a match between registered points and planned mass-points.
In this way a comparison, between a planned shape and a one registered by the camera, is possible: a measure of error can be provided by computing the distances between planned mass-points and registered ones.
The average of the errors of mass-points composing a shape will be considered as the error associated to such shape.

Different types of deformable linear objects have been used for the tests, in order to explore the robustness of the planner and to validate the chosen weighting in the optimization procedure, depending on the three different DLOs families described in Chapter 6.
In particular cables with different Young's modulus are involved, exploiting the different tuning of the algorithm approaching different stiffness:

- A hose in polyurethane (PU) with diameter $d = 6mm$, length $l_0 = 0.50m$, Young's Modulus $E = 1 \cdot 10^8 Pa$ and mass $m = 11g$;
- A hose in polyamide (PA12) with diameter $d = 6mm$, length $l_0 = 0.50m$, Young Modulus $E = 1 \cdot 10^9 Pa$ and mass $m = 9g$;
- An USB cable with diameter $d = 3mm$, length $l_0 = 0.5m$, mass $m = 10g$ and equivalent identified Young's modulus $E = 2.5 \cdot 10^6 Pa$ (The stiffness identification procedure is provided in the next Subsection).
- An Ethernet cable with diameter $d = 5mm$, length $l_0 = 0.5m$, mass $m = 10g$ and equivalent identified Young's modulus $E = 7.5 \cdot 10^6 Pa$ (The stiffness identification procedure is described in the next Subsection).

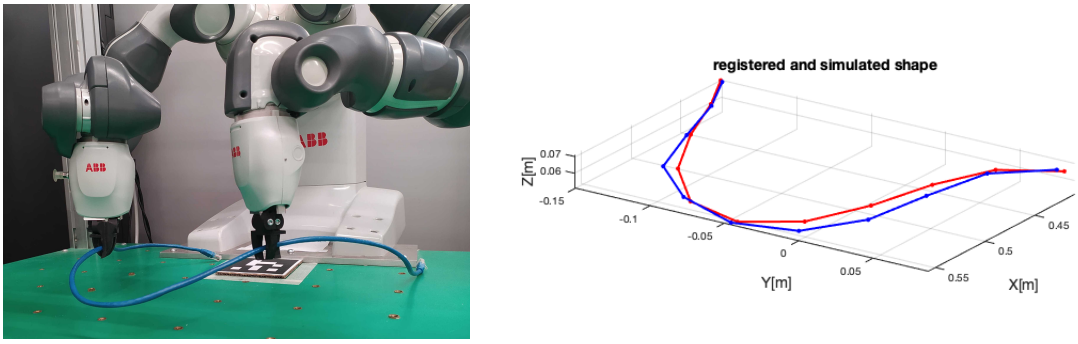
It is important to underlie that the first two mentioned DLOs are composed of an unique known material, hence their Young modulus can be found in tables. On the other hand the USB and the Ethernet cable are composite materials and hence it is necessary to identify their Young's modulus according to the strategy in Chapter 6.

The tests can be divided in two parts: the first part (described in Subsection 7.2.2) involves

some manipulations that keep the cable to be near a two-dimensional configuration, for this kind of tests it was possible to inspect the errors in a quantitative way, exploiting the vision system described in Section 7.1.
The second part of the tests (described in section 7.2.3) involves some three-dimensional manipulations, in this case a qualitative inspection of the shapes is carried out, since a tracking algorithm for deformable linear objects in the 3D space was not available, to have good enough data to perform the comparison and the quantitative analysis.
Finally, a use case about wire harness assembly involving the USB cable, the Ethernet cable and the PU hose is described in section 7.3.

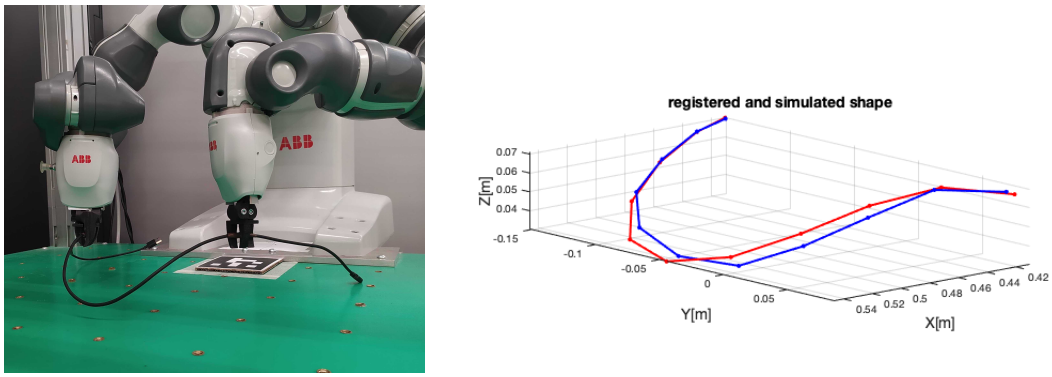
7.2.1. Young's modulus identification for USB and Ethernet cables

The USB and the Ethernet cables are composite cables, for this reason their Young's modulus have been identified with the method proposed in this work (Section 6.1): In particular, both the cables are subjected to a manipulation, and their final shapes have been registered to be used in the identification algorithm as shown in Figure 7.4 for the Ethernet cable, and in Figure 7.5 for the USB cable.



(a) Final manipulation pose. (b) Registered shape and simulated one with $E_{identified}$.

Figure 7.4: Identification manipulation for the Ethernet cable.



(a) Final manipulation pose. (b) Registered shape and simulated one with $E_{identified}$.

Figure 7.5: Identification manipulation for the USB cable.

For both identifications the ranges of values for E are constrained to be between $E_{min} = 1 \cdot 10^6 Pa$ and $E_{max} = 1 \cdot 10^8 Pa$.
The identified values are:

- For Ethernet cables: $E_{identified} = 7.5 \cdot 10^6 Pa$;
- For USB cables: $E_{identified} = 2.5 \cdot 10^6 Pa$.

Further details about the identification are reported in the Table 7.1. The "Simulations" parameter describes the number of simulations executed during the research, the "Iterations" parameters describes the number of computations of the gradient of the function (generally two simulations for each iteration are required), "$f(x)$" parameter describe the optimal value of the cost function, the "First Order Optimality" parameter describe the optimal value of the gradient of the function. Finally "$E_{identified}$" describes the optimal value of the identified Young's modulus.

	Iterations	**Simulations**	**f(x)**	**first order optimality**	$E_{identified}$
Ethernet	22	46	0.0004	$4.75\dot{1}0^{-6}$	$7.5 \cdot 10^6$Pa
USB	20	42	0.00039	$1.68\dot{1}0^{-5}$	$2.5 \cdot 10^6$Pa

Table 7.1: Identification details for the Young's modulus of the USB cable and the Ethernet cable.

7.2.2. Quantitative analysis

For this part of analysis 3 tests are carried out, involving 2 different manipulations (M1 and M2) and 3 different cables (PU hose, USB, PA12 hose). The three tests are:

- A "sinusoidal deformation" manipulation (M1) for the PU Hose (7.2.2);
- A "uniform curvature deformation" manipulation (M2) for the USB cable (7.2.2);
- A "uniform curvature deformation" (M2) manipulation for the PA12 Hose (7.2.2).

Each manipulation is executed and registered 2 times, in order to provide a statistical analysis on the errors of each shape.

In Table 7.2 some measures are shown, describing the average error experienced in each test. Such errors are provided computing the mean errors on the mass-points of all the shapes. Each manipulation is composed by 7 shapes, and each shape is composed by 11 mass-points.

Planned manipulation	DLO	E $[Pa]$	$error_x$ $[m]$	$error_y$ $[m]$	$error_z$ $[m]$	$error_{tot}$ $[m]$
M1	PU Hose	$1 \cdot 10^8$	0.0183	0.0193	0.0142	0.0173
M2(USB)	USB cable	$2.5 \cdot 10^6$	0.0041	0.0062	0.0046	0.005
M2(PA12)	PA12 Hose	$1 \cdot 10^9$	0.0068	0.0111	0.0020	0.0066

Table 7.2: Results provided by the quantitative analysis.

The z-errors are provided by inspecting the height of the registered points of the cable manually, providing points in the 3D space. By analysing the results, it can be noticed that for M2(USB) and M2(PA12) there is not much difference on the errors. Such tests involve the same manipulation with a stiff and a soft cable. This means that the stiffness of the cable has a tiny impact on the accuracy of the manipulation.
A bigger difference is noticed in M1, in which a larger error is experienced. In such test the desired shape is more complex. Hence it can be noticed that the accuracy of the manipulation is much sensitive on the complexity of the shapes of the cable.

Sinusoidal deformation for PU Hose tube

For such manipulation, the PU Hose tube is used. The stiffness of the cable is enough to slightly compensate gravity deformations, but at the same time it allows to assume

quite stressed configurations. The planned manipulation M1 aims to deform the cable in a target sinusoidal shape as shown in Figure 7.6. A top view (from the RealSense camera point of view) is shown in Figure 7.7, for each experienced shape during the dual arm robotic manipulation.

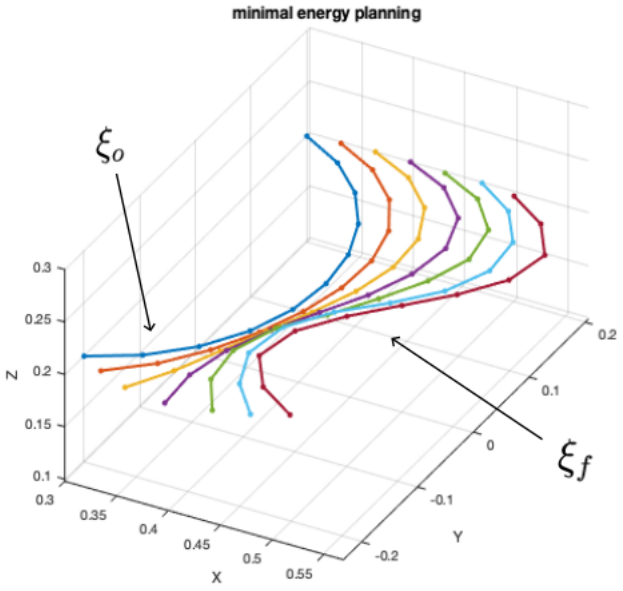
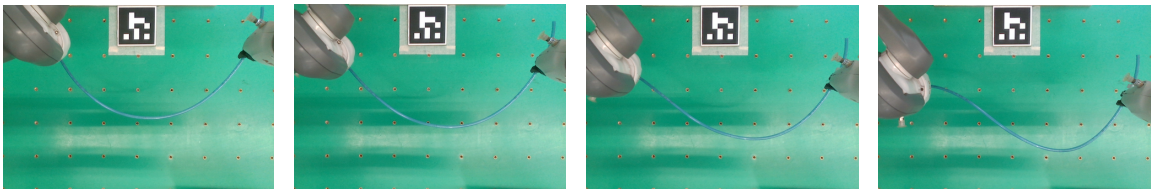
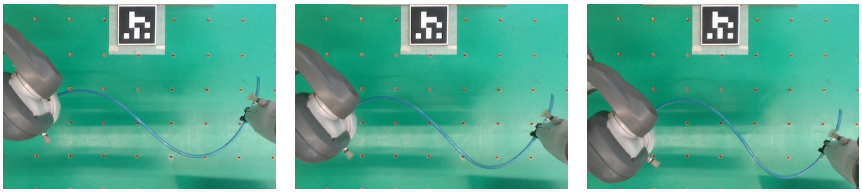


Figure 7.6: Minimal energy shapes planned for the PU Hose tube, according to M1.



(a) Shape ξ_o. (b) Shape ξ_1. (c) Shape ξ_2. (d) Shape ξ_3.



(e) Shape ξ_4. (f) Shape ξ_5. (g) Shape ξ_f.

Figure 7.7: Shapes provided by the dual arm robotic manipulation for the quantitative test 1 (M1).

Such shapes are registered by the vision system, and compared with the stabilized shapes provided by the static simulation environment, as shown in Figure 7.8.

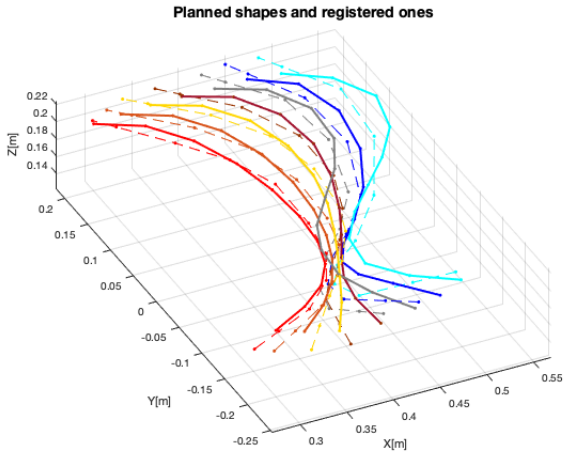
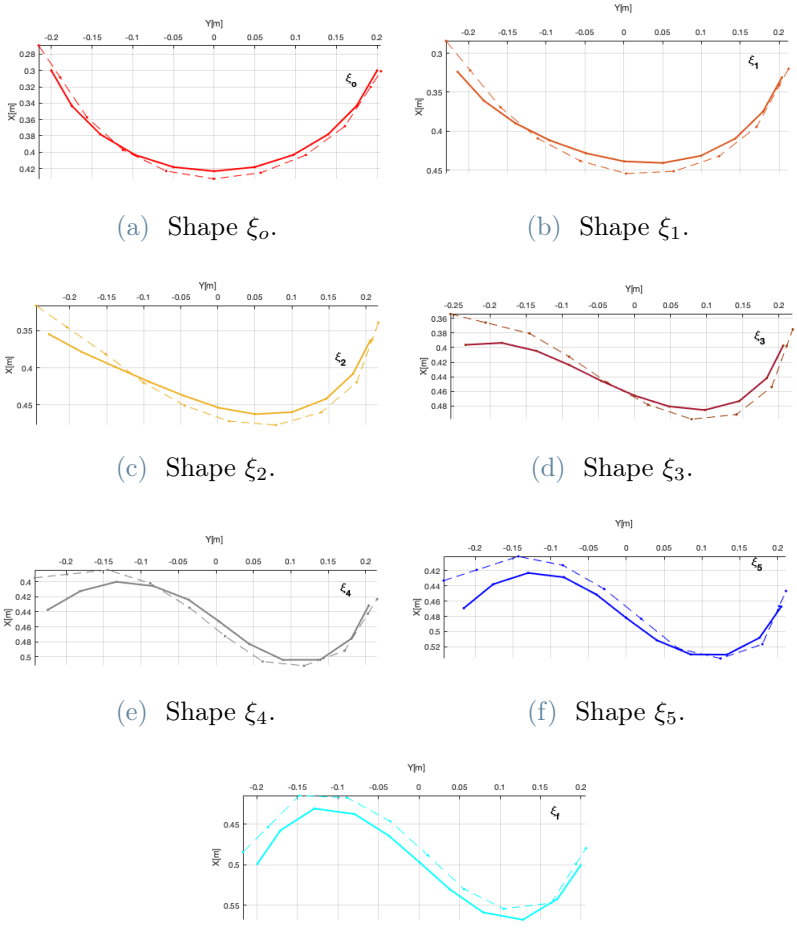


Figure 7.8: Planned shapes (Bold lines) and registered shapes (Dashed lines), referred to the manipulation in Figure 7.6.



(a) Shape ξ_o.

(b) Shape ξ_1.

(c) Shape ξ_2.

(d) Shape ξ_3.

(e) Shape ξ_4.

(f) Shape ξ_5.

(g) Shape ξ_f.

Figure 7.9: Planned shapes and registered ones, for M1.

In figure 7.9 the planned shapes and the registered ones are provided, with a focus on each shape involved in the manipulation.
The mean error of each shape is reported in Figure 7.10. Such errors can be computed considering the average error on each mass-point composing the registered shape, compared to the respective planned one, considering all the test carried out.

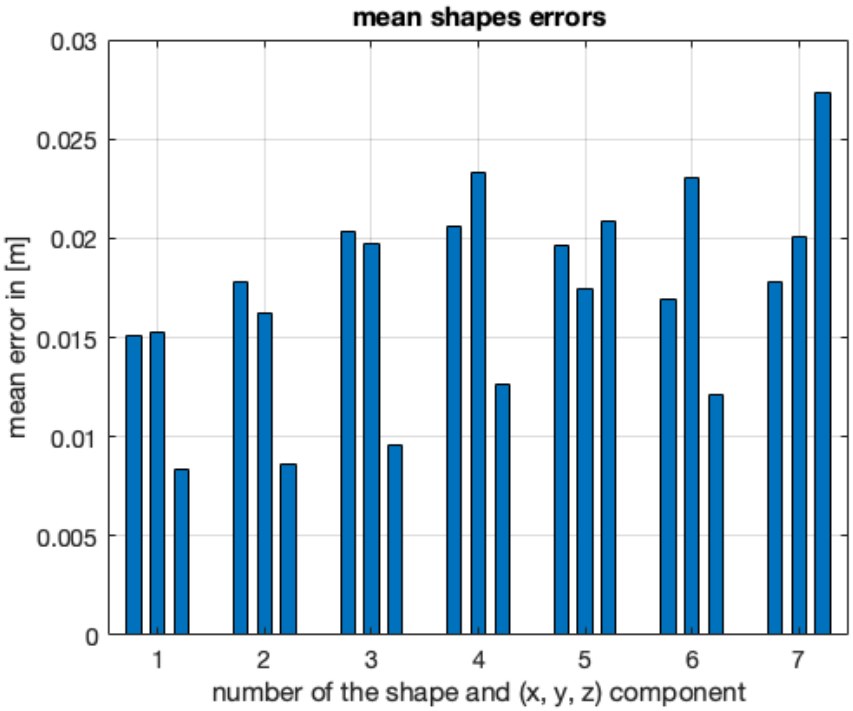


Figure 7.10: Mean errors on each shape for the manipulation provided in Figure 7.8.

Uniform curvature deformation for USB cable

The planned manipulation M2 aims to deform the cable as shown in Figure 7.11. Such manipulation can be considered simpler than M1, because a two dimensional shape with uniform curvature is expected. On the other hand, such manipulation applied on soft cables is very sensitive to gravity effects. For this reason the manipulation is planned with two different cables: M2(USB) is defined for an USB cable (analysed in this subsection), with a low Young's modulus, while M2(PA12) is planned for a PA12 Hose tube (subsection 7.2.2), characterized by an high Young's modulus able to compensate gravity deformations.

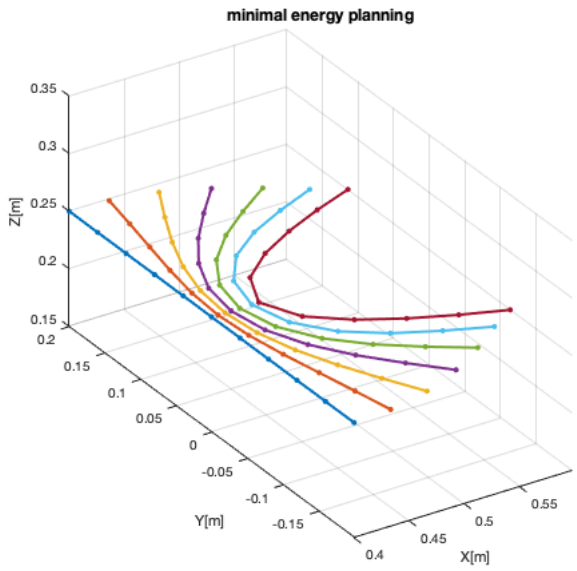


Figure 7.11: Minimal energy shapes planned for the USB cable, according to M2(USB).

Figure 7.12 shows the stabilized shapes: the planned shapes are deformed by gravity effects (especially the last one), due to the low Young's modulus of the cable. Those shapes will be used for the error computation.

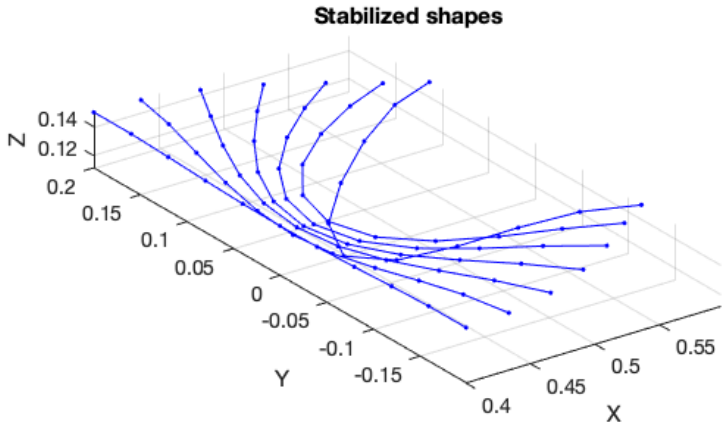
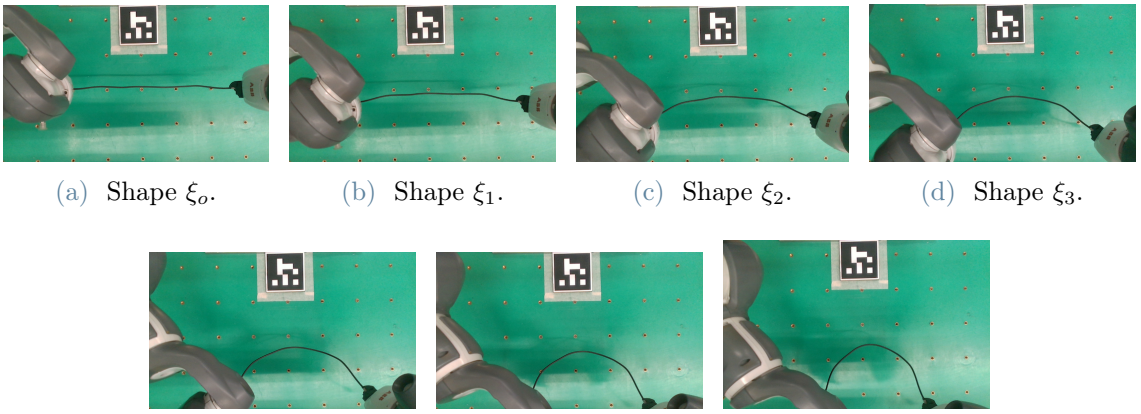


Figure 7.12: Gravity stabilization for the shapes in the manipulation of Figure 7.11, according to M2(USB).

The shapes experienced during the dual arm manipulation are provided in Figure 7.13.



(a) Shape ξ_o. (b) Shape ξ_1. (c) Shape ξ_2. (d) Shape ξ_3.

(e) Shape ξ_4. (f) Shape ξ_5. (g) Shape ξ_f.

Figure 7.13: Shapes provided by the dual arm robotic manipulation for the quantitative test 2 M2(USB).

The comparison between stabilized shapes and registered ones is provided in Figure 7.14: the shapes are very similar and a slight error is found due to a small mismatch on the gravity deformation.
The mean errors on the shapes are provided in Figure 7.15. It can be noticed that a larger error on x, y and z is achieved on the last shape, with respect to the others. Despite the simple features of the manipulation, the soft behaviour of the cable leads to a less accurate prediction of the shape because the cable is less stretched with respect to the other shapes.

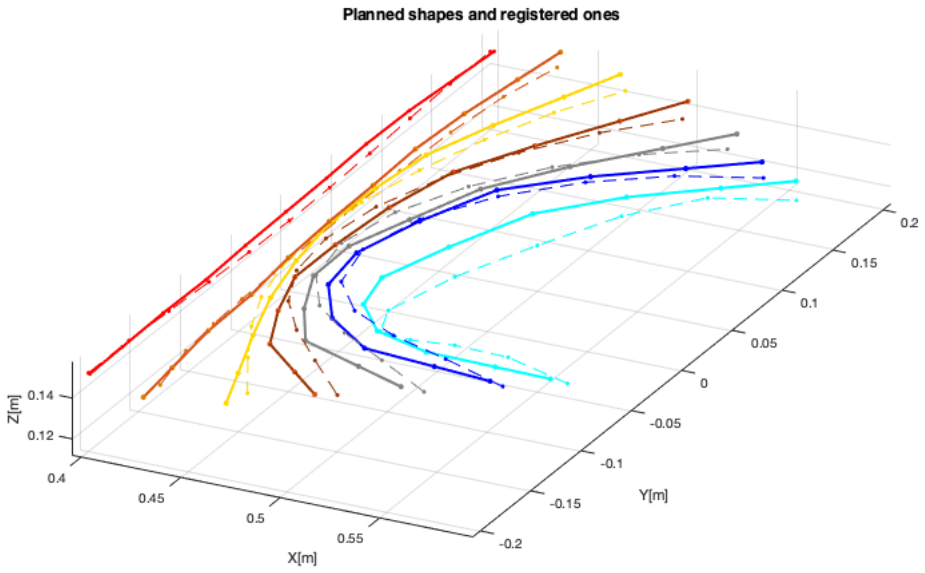


Figure 7.14: Planned shapes (Bold lines) and registered shapes (Dashed lines), referred to the manipulation in Figure 7.11.

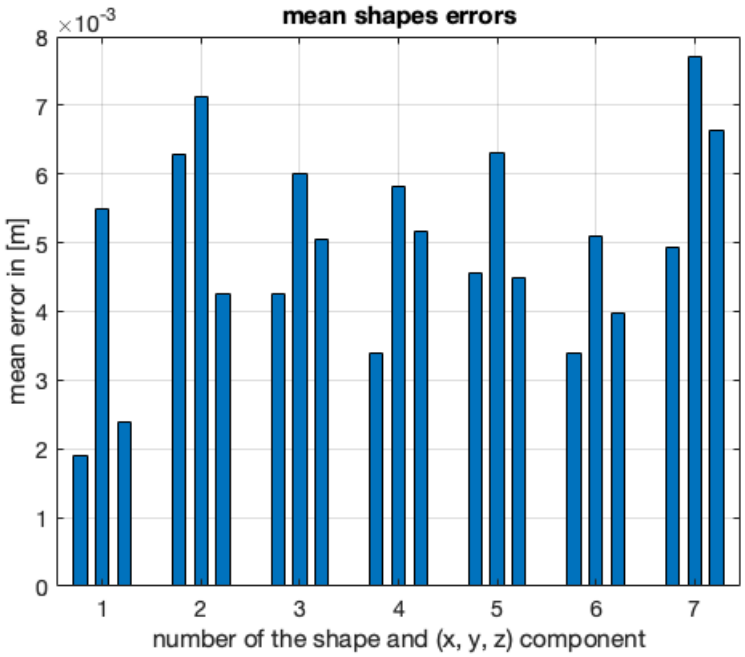


Figure 7.15: Mean errors on each shape for the manipulation in Figure 7.14.

Uniform curvature deformation for PA12 Hose cable

The same target pose ξ_f is imposed for the manipulation of the PA12 Hose, obtaining the manipulation M2(PA12). In Figure 7.16 it is shown as the stabilized shapes are almost identical to the planned ones, due to the high stiffness of the tube. In Figure 7.17 the real shapes experienced during the manipulation are provided.
The comparison between planned and registered shapes (Figure 7.18) are reported, with the averaged errors (Figure 7.19).

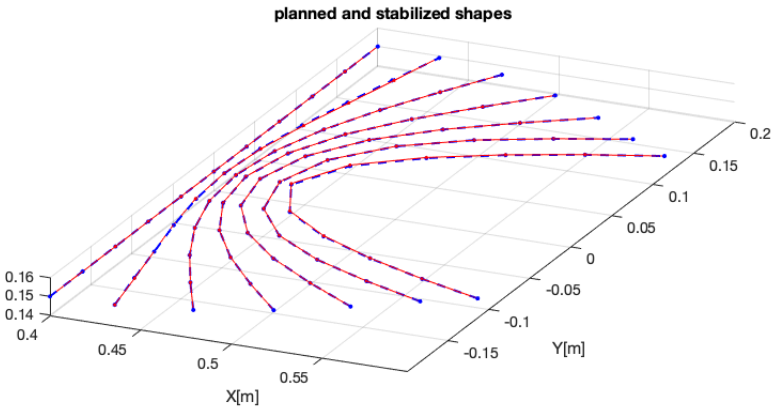
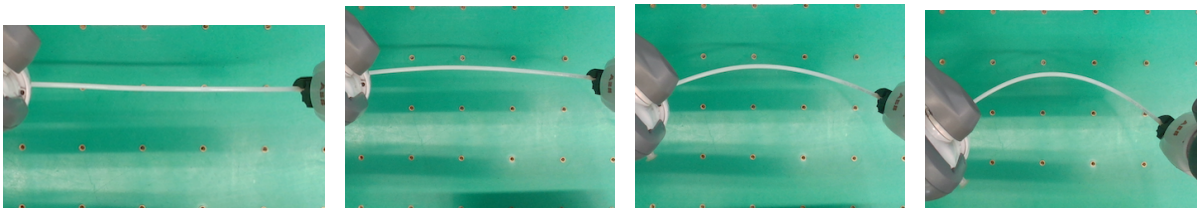
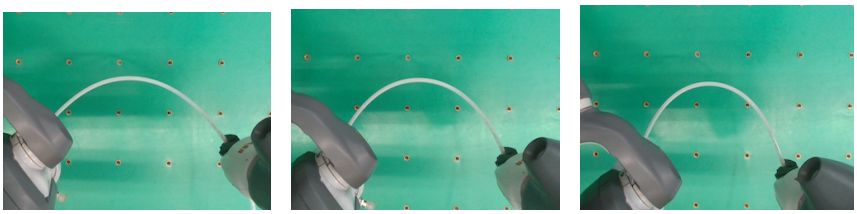


Figure 7.16: Planned (red) and stabilized (dashed blue) shapes for the M2(PA12) manipulation.



(a) Shape ξ_o. (b) Shape ξ_1. (c) Shape ξ_2. (d) Shape ξ_3.



(e) Shape ξ_4. (f) Shape ξ_5. (g) Shape ξ_f.

Figure 7.17: Shapes provided by the dual arm robotic manipulation for the quantitative test 3 M2(PA12).

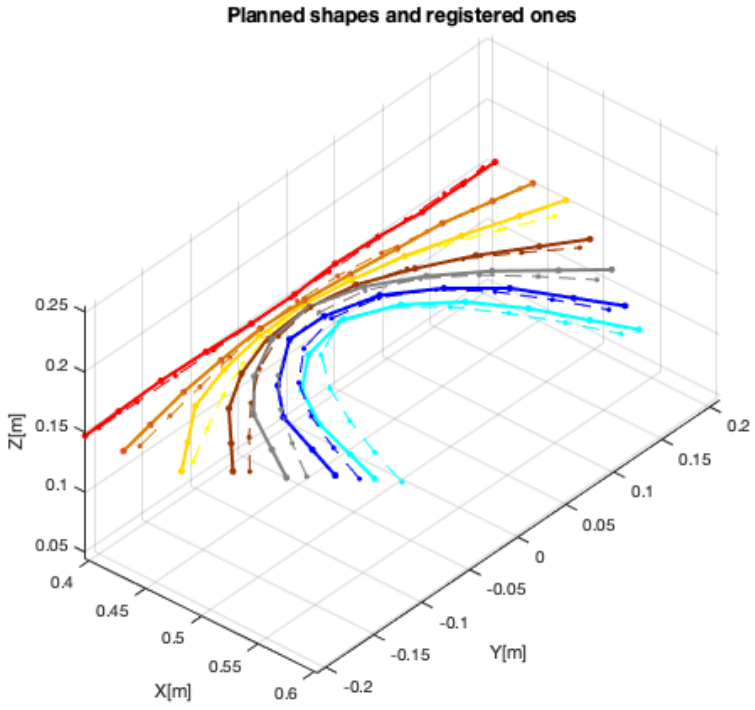


Figure 7.18: Planned shapes (Bold lines) and registered ones (Dashed lines) for a manipulation of a PA12 Hose.

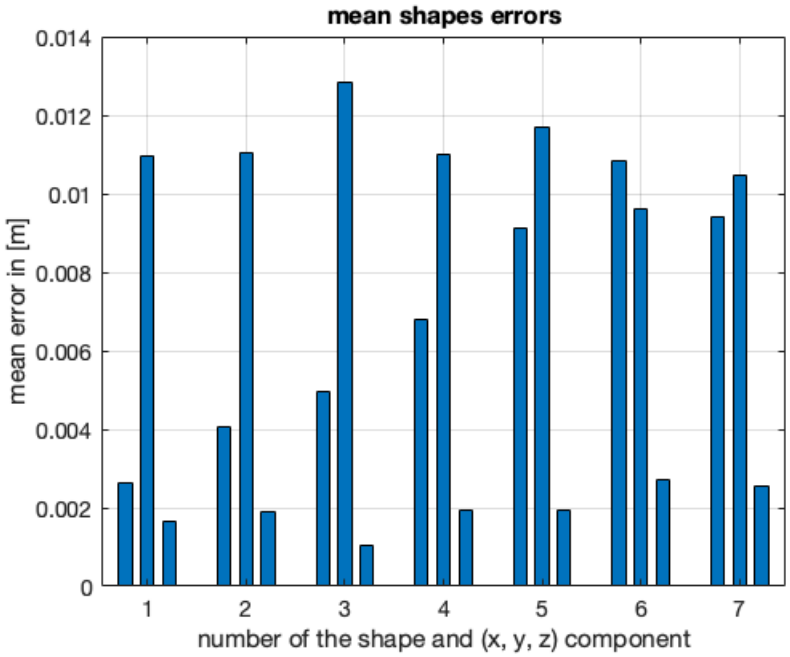


Figure 7.19: Average errors on each shape for the manipulation in Figure 7.18.

In this example a smaller error along the z-axis can be noticed with respect to the USB manipulation case, the reason is that a stiffer deformable linear object undergoes less deformations due to gravity effects, providing stable shapes that are closer to the planned minimal energy ones, involving less motions in the stabilization phase.
On the other hand, an higher error is experienced along the y-axis with respect to the USB cable. That is due to the higher stiffness of the cable, that generates higher bending forces. Those forces tends to highlight non-idealities of the cable, such as plastic deformations, that are not considered in the model.
Moreover it can be noticed that the "sinusoidal deformation" experiment linked to M1 is characterized by higher errors. That's due to the higher complexity of the manipulation, that, together with the gravity deformation phase and a quite high stiffness of the PU Hose can lead to uncertainty in the shape predicted by the mass-spring model used in this work.

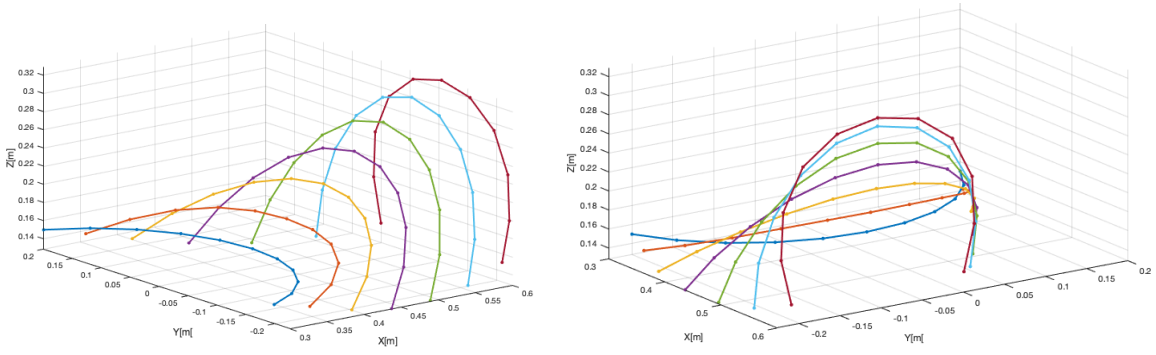
7.2.3. Qualitative analysis

For this experimental phase two tests are involved:

- The first manipulation (denoted as qualitative test 1) concerns a curved PA12 Hose tube, the cable is rotated and translated in the space, involving small deformations.
- The second manipulation (denoted as qualitative test 2) concerns the Ethernet cable, and involves a more various deformation of the cable.

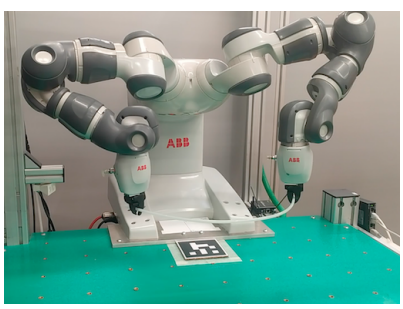
Qualitative test 1

The planned manipulation for the test 1 is provided in Figure 7.20. Such manipulation can be projected in an industrial framework, where a tube laying on the table must be raised in order to allow an assembly operation on the two ends.
The shapes experienced during the dual arm manipulation are provided in Figure 7.21, it can be noticed that the expected configurations are obtained, thanks to the stiffness properties of the cable.
The Young's modulus of the cable is $E = 1 \cdot 10^9 Pa$ and it's labeled as stiff-cable: the optimization tuning providing the shapes is changed accordingly and the static simulation environment provides an almost null deformation due to gravity effects, as found in the real manipulation.
No obstacles are involved in this framework, hence the re-planning phase is not necessary.

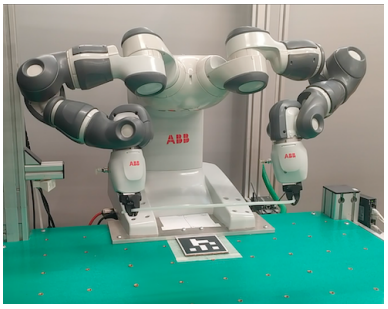


(a) Planned shapes. (b) Planned shapes.

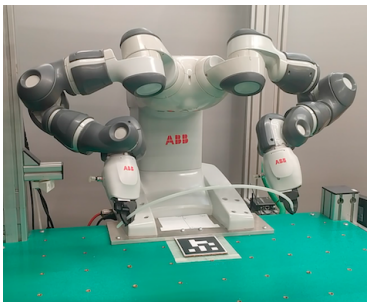
Figure 7.20: Different perspective view of the planned shapes for the qualitative test 1, in order to be compared with the experienced shapes.



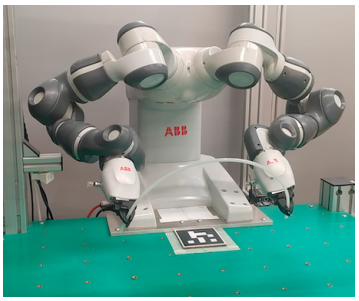
(a) Shape ξ_o.



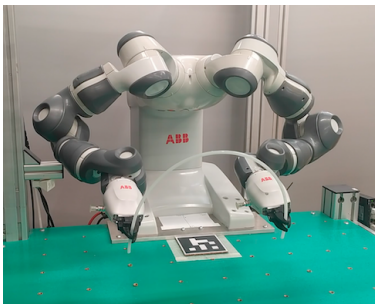
(b) Shape ξ_1.



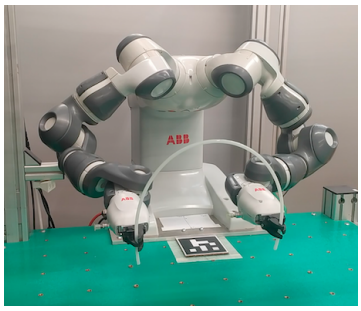
(c) Shape ξ_2.



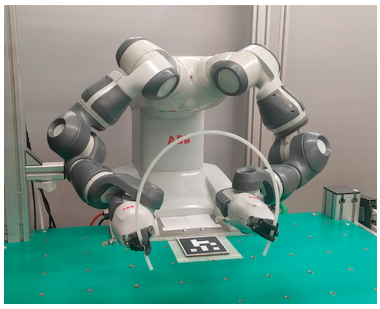
(d) Shape ξ_3.



(e) Shape ξ_4.



(f) Shape ξ_5.



(g) Shape ξ_f.

Figure 7.21: Shapes provided by the dual arm robotic manipulation for the qualitative test 1.

Qualitative test 2

This planned manipulation concerns the Ethernet cable: the planned shapes are provided in Figure 7.22.

Such manipulation is often required in cabin network assembly operations, in which Ethernet cables are used to connect different components in the switchgears. The cable has a Young's modulus about $E = 7.5 \cdot 10^6 Pa$ and it's labeled as a medium-stiff-cable, the optimization and the simulation have been tuned accordingly. As we can see in Figure 7.23, the planned shapes undergo some gravity deformation effects, and the same can be noticed in the experimental phase.

No obstacles are involved in the manipulation. The experienced shapes during the robotic manipulation can be found in Figure 7.24.

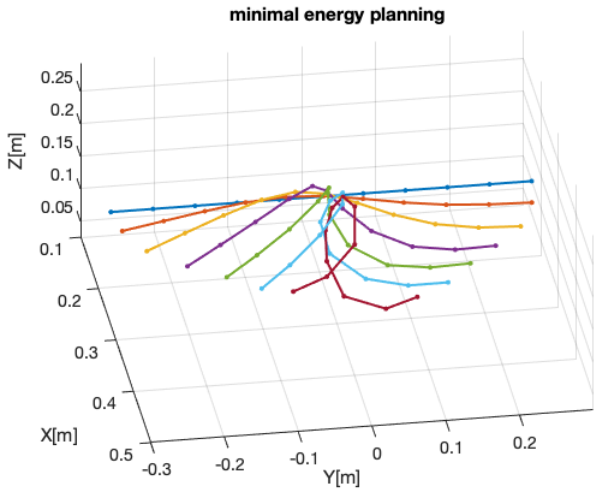


Figure 7.22: Planned shapes for the qualitative test 2.

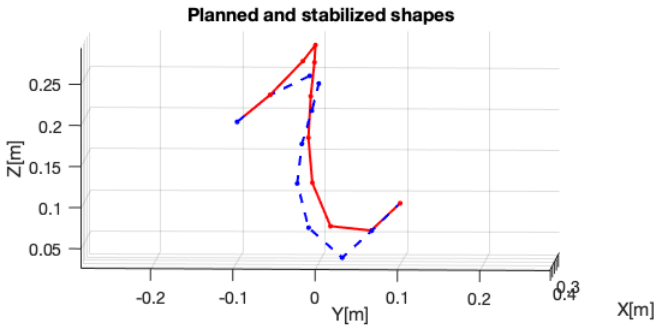
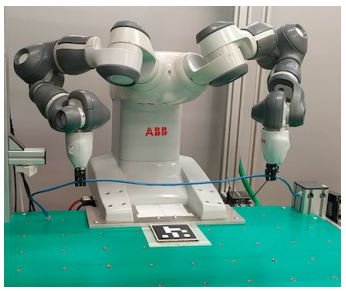
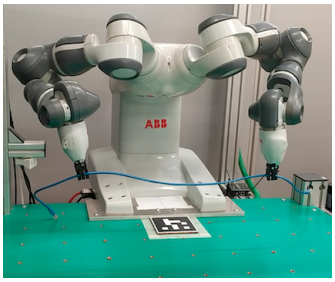


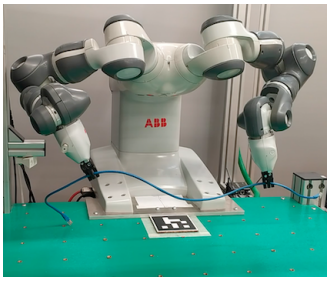
Figure 7.23: Final shape and stabilized one for the qualitative test 2.



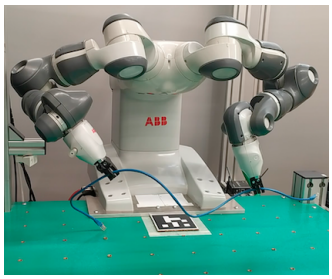
(a) Shape ξ_o.



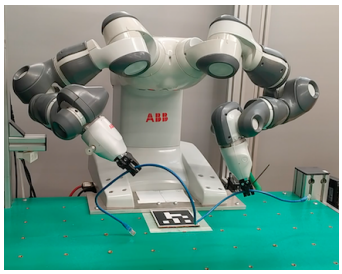
(b) Shape ξ_1.



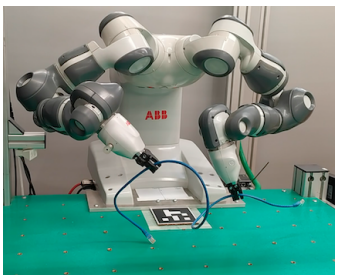
(c) Shape ξ_2.



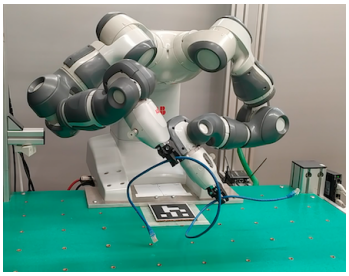
(d) Shape ξ_3.



(e) Shape ξ_4.



(f) Shape ξ_5.



(g) Shape ξ_f.

Figure 7.24: Shapes provided by the dual arm robotic manipulation for the qualitative test 2.

7.3. Use case: Wire harness assembly

In industry, assembly operations often have to deal with deformable linear objects such as wires, cables or tubes: a wide amount of components need to be connected through those kinds of objects, some examples can be electric components such as electric motors or switchgears, or mechanical parts in automotive sectors, in which oil tubes must connect different components. For this reason the proposed strategy can be easily applied in an industrial operation, facing the need of manipulate deformable linear objects.
The use case provided in this work takes place in an assembly operation. The operation is a wire harness assembly task: different kinds of cables must be placed together on an holder, clipping them in some fixtures and shaping them with some pegs, an example of such operation is provided in Figure 7.25. Usually this operation is followed by a clamping phase, in which cables are tied together.

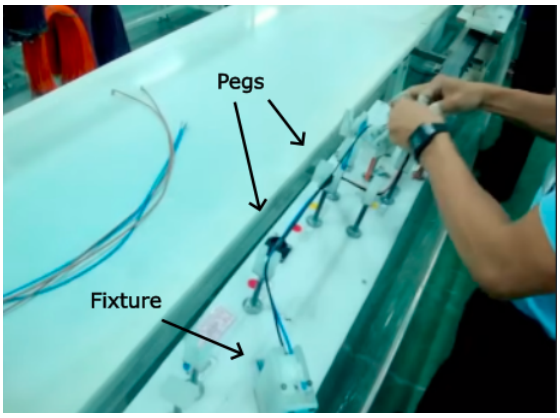
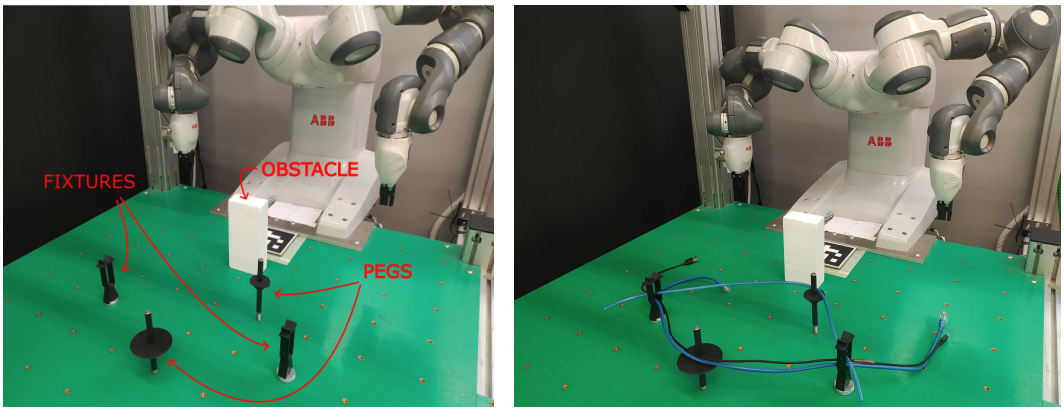


Figure 7.25: Example of harness assembly task.

In the wire assembly task, the operator has to handle cables. Due to complex behaviours associated to DLOs, the operator has to keep attention during the task, inspecting the cable behaviours, and deforming the shape of the cable with hands contacts. Those aspects may over-tire the operator, leading to a slowdown of the cycle time and introducing possible defects of the mounted components.
The proposed methodology can be applied to automatize the routing of cables in wire harness assembly. Indeed, the problem consists into manipulating the cable from an initial position to a final one, providing the necessary curvature to the cable, in order to be inserted in some fixtures and to fit through some pegs. Moreover, to create a more complex and industrial setup, an obstacle is introduced on the working table. Once the deformable linear object has been brought to ξ_f, it can be inserted in the fixtures by applying a downward relative motion, performing a clipping operation.

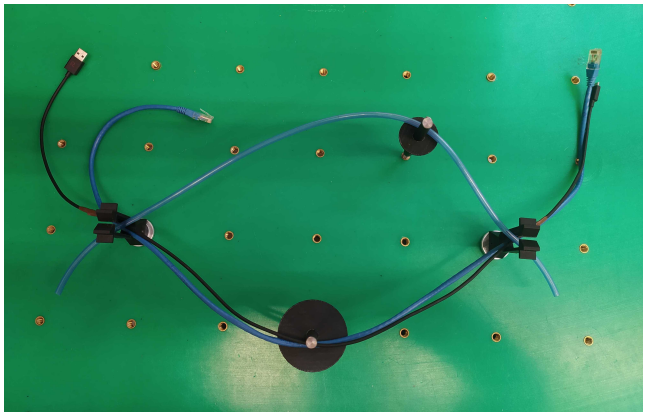
After the clipping operation, the arms tension the DLO moving of a predefined offset in the opposite direction with respect to the fixture. The offset is determined exploiting a simulation carried out with the dynamical mass-spring model, used in order to inspect the behaviour of the cable in contact with the fixtures and the pegs for the insertion phase. The tensioning of the cable allows to properly contact the peg.

The considered set up is shown in Figure 7.26, two fixtures are used to clip the cable at the two ends, and two pegs are used to shape the cables in the desired configuration. Some plates (mounted on the pegs) are used to provide a support on which the cable can lay. A generic known obstacle is introduced in the working table.



(a) Use case framework before the manipulations.

(b) Use case framework after the manipulations.



(c) Cables shaped at the end of the operations.

Figure 7.26: Framework set up for the wire harness assembly operation. The cables must be placed in some fixtures, laying on the plates attached on the pegs

Three cables with different mechanical properties are: the USB cable, the Ethernet cable and the PU Hose tube, described in section 7.2. The operation results are summarised

in Table 7.3, since the aim of this work is in an intermediate point between explicit and implicit planning for DLO manipulation, the analysis of the operations is not provided in terms of errors related to the shapes, but an analysis on the success rate is provided.
In the following the planning procedure for the three cables (the USB cable, the Ethernet cable and the PU Hose) is described, together with the obtained shapes during the real dual arm robotic manipulation.

DLO	E	TCP speed	time	experiment	success	success rate
USB	$2.5 \cdot 10^6[Pa]$	$100[mm/s]$	$9s$	15	13	86%
Ethernet	$7.5 \cdot 10^6[Pa]$	$100[mm/s]$	$9s$	15	14	93%
PU hose	$1 \cdot 10^8[Pa]$	$100[mm/s]$	$9s$	15	12	80%

Table 7.3: Results related to the single operations involved in the use case.

7.3.1. Wire harness assembly for USB cable

The planning procedure is divided in two phases, as shown in Figure 7.27 : the first phase aims to plan the intermediate shapes of the cable and the intermediate poses of the robot clips to bring the cable from the initial configuration to the operation area, in particular in this phase the obstacle avoidance is performed and the curvature of the cable is provided in order to match with the pegs. The second phase exploits the planning strategy in order to provide a deformation of the cable for the alignment of the extremities with the fixtures on the table, preparing it for the insertion. In particular, to properly perform the clipping procedure, the cable must be locally aligned with the fixtures, meaning that also the gripper holding that portion must be aligned with the fixture. An additional step is introduced, that aims to insert the cable into the fixtures exploiting the contact with the pegs, the fixtures and the plates.

Moreover, the internal forces of the cable during the manipulation phase are inspected, to verify that the cable is not stressed during the operation: the mean and the variance of the Bending and Axial forces on each mass points considering all the involved shapes are computed. A low mean is provided because the shapes are minimal energy ones, and also a low variance is provided, because the cable deformation is expected to be uniform during the manipulation.

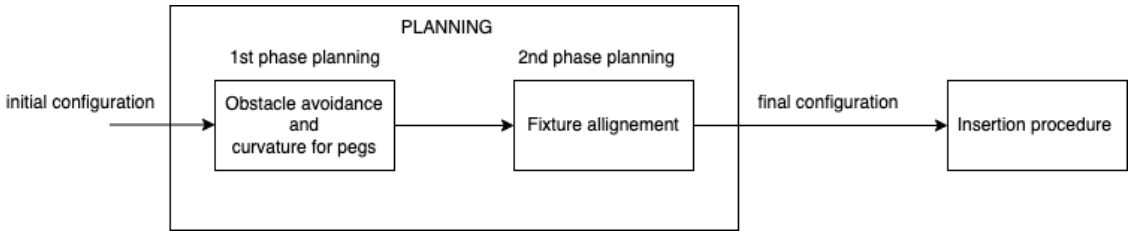
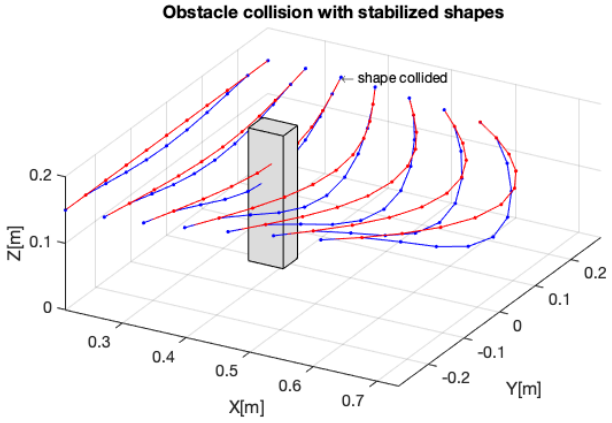
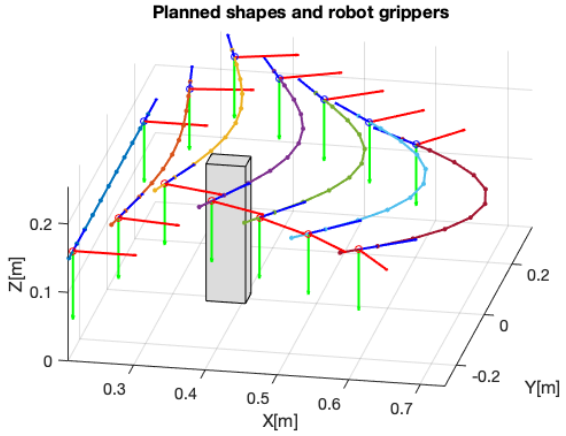


Figure 7.27: Planning procedure for the use case.

Some relevant aspects about the 1^{st} phase planning are provided: the planned shapes and the obstacle avoidance computations are reported in Figure 7.28, the relative analysis on forces is provided in Figure 7.29.
The planned shapes and the forces analysis for the 2^{nd} step planning are provided in Figure 7.30 and 7.31.

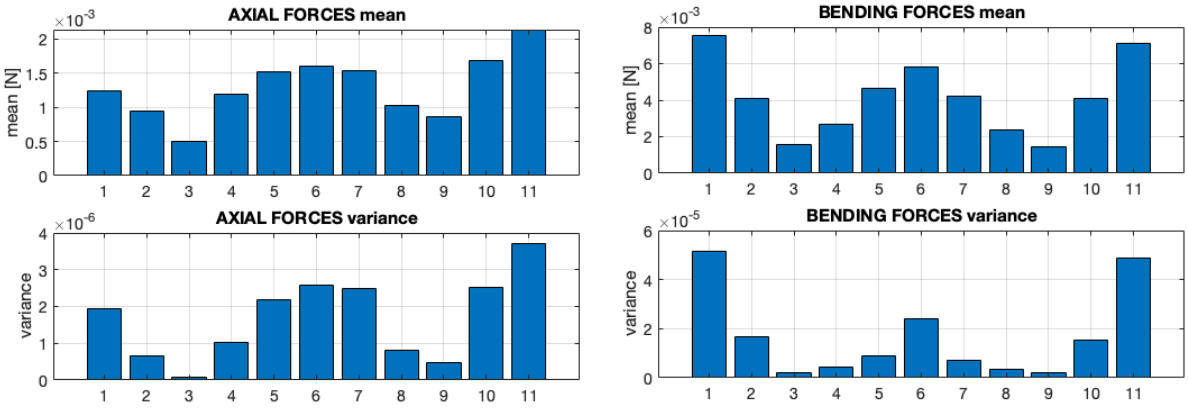


(a) Collision detection with stabilized shapes.



(b) Re-planned shapes and robotic gripper poses.

Figure 7.28: USB cable planning for phase 1 of the operation.



(a) Axial forces analysis.

(b) Bending forces analysis.

Figure 7.29: Mean and variance of axial and bending forces on each mass-point of the shape, considering all the planned shapes for the phase-1 of the operation.

During the phase-1, the forces and their variance are very low, leading to low internal forces and smooth manipulation. It's possible to notice that in the phase-2, the analysis (Figure 7.31) leads to higher values for the forces, this is due to the high curvature to be imposed on the cable (shown in Figure 7.30).

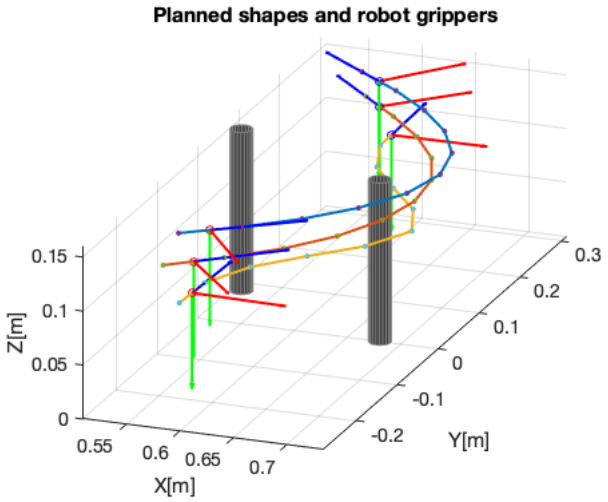
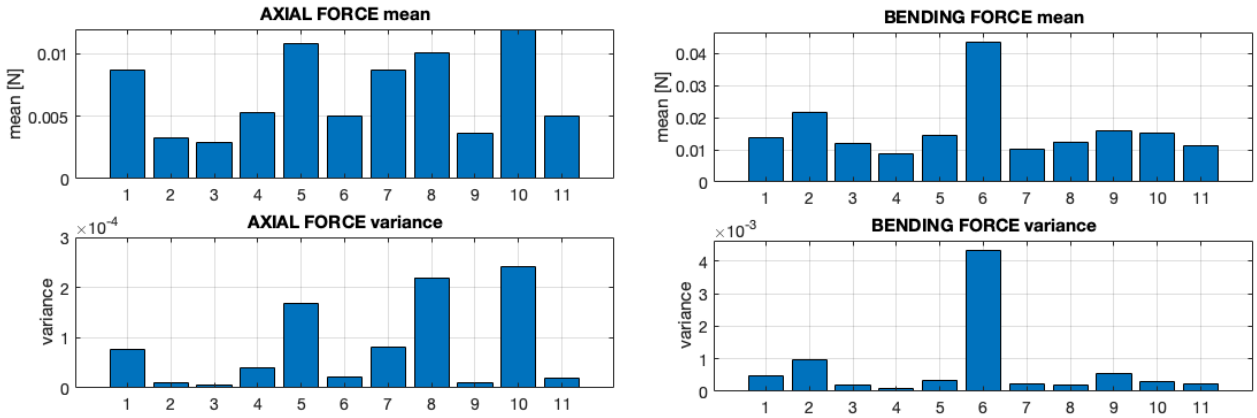


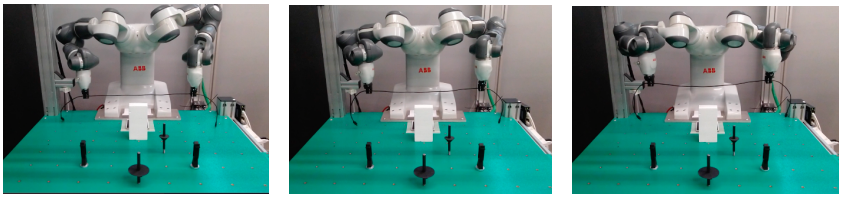
Figure 7.30: USB cable planning for phase-2 of the operation.



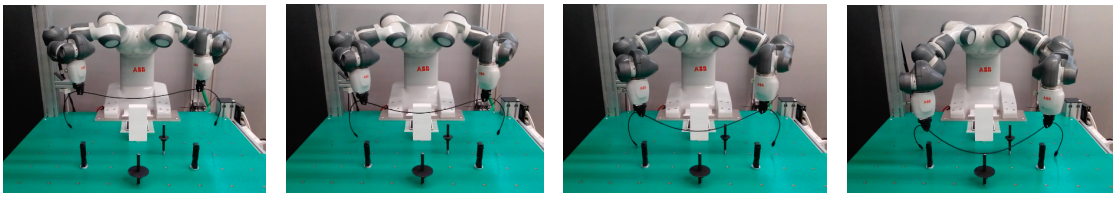
(a) Axial forces analysis. (b) Bending forces analysis.

Figure 7.31: Mean and variance of axial and bending forces on each mass-point of the shape, considering all the planned shapes for the phase-2 of the operation.

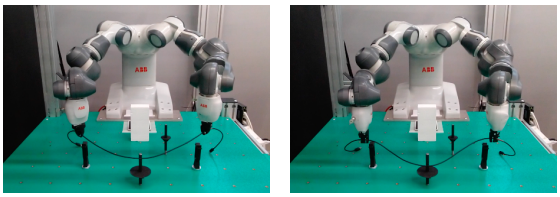
The total dual robotic manipulation with the intermediate shapes experienced is reported in Figure 7.32.



(a) Shape ξ_o. (b) Shape ξ_1. (c) Shape ξ_2.



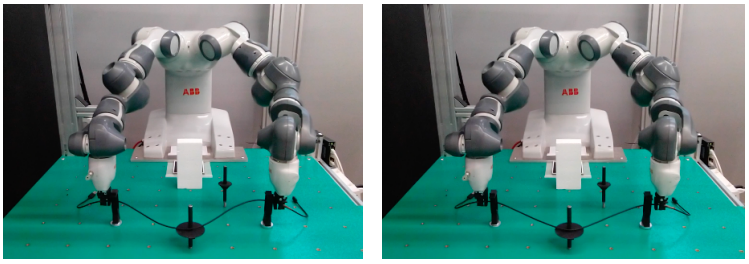
(d) Shape ξ_3. (e) Shape ξ_4. (f) Shape ξ_5. (g) Shape ξ_f.



(h) Shape $\xi_{1,alignment}$. (i) Shape $\xi_{f,alignment}$.

Figure 7.32: Shapes provided by the dual arm robotic manipulation for the USB cable.

The Fixture insertion and the tensioning of the cable are shown in Figure 7.33.



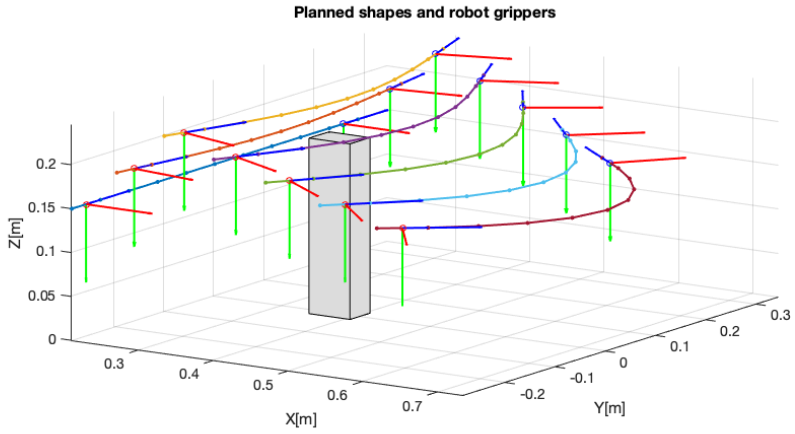
(a) Insertion in the fixtures. (b) Stretching against the peg.

Figure 7.33: Insertion step for the USB cable operation.

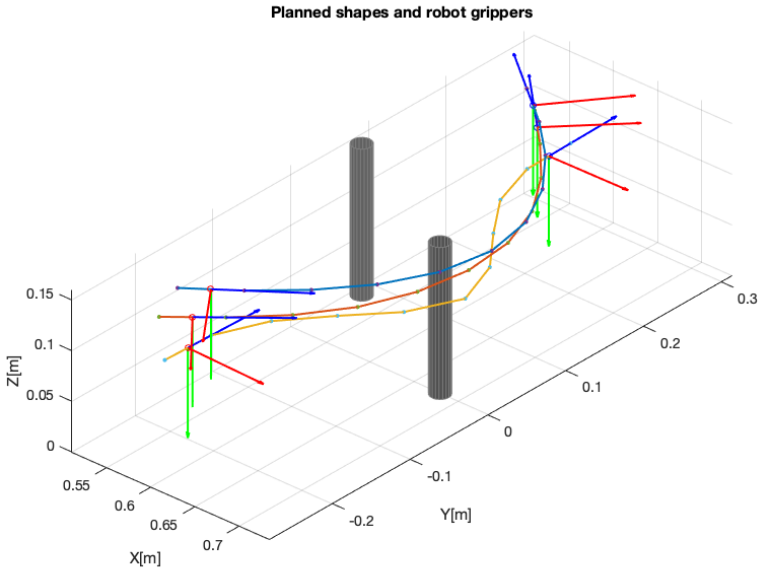
Good behaviours are observed during the manipulation, and good results are obtained in terms of success rate: the task is repeated 15 times in order to inspect the percentage of successful operation. The success rate is registered to be 86%, failures are detected when the cable is not properly gripped, due to the small diameter involved. This, together with the low stiffness of the cable, brings to the introduction of undesired deformations, leading to a failure of the operation. For this reason a very good grasping of the cable is needed when soft cables are involved in the manipulation.
The task is performed with a TCP speed of $100mm/s$, and it's completed in $9s$.

7.3.2. Wire harness assembly for Ethernet cable

The planning procedure is the same used for the USB cable: the phase-1 aims to bring the cable in the operation area, and the phase-2 aims to align the cable with the fixtures. The manipulation is very similar to the previous case, since an identical operation is required: the Ethernet cable must be placed in the fixtures in the same configuration of the USB cable. In the following the planned shapes and the planned grippers for the Ethernet cable manipulation (reported in Figure 7.34), with the related analysis on the internal forces of the cable during the manipulation (Figure 7.35 for the 1^{st} phase of the operation, and in Figure 7.36 for the 2^{nd} phase).
In this case, higher axial and bending forces can be noticed, with respect to the previous case, the reason is that the Ethernet cable is stiffer than the USB cable, and hence for a given deformation an higher force is provided. The intermediate shapes experienced in the experimental phase during the dual arm robotic manipulation are shown in Figure 7.37.

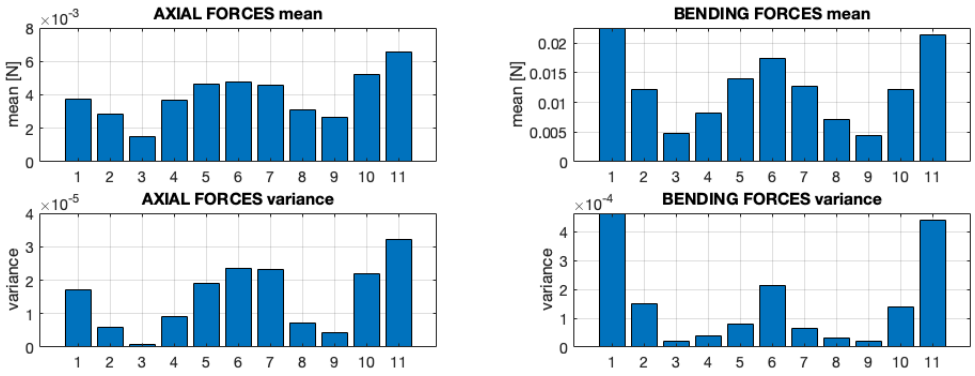


(a) Phase 1 planning, the analysis on forces is provided in Figure 7.35 .



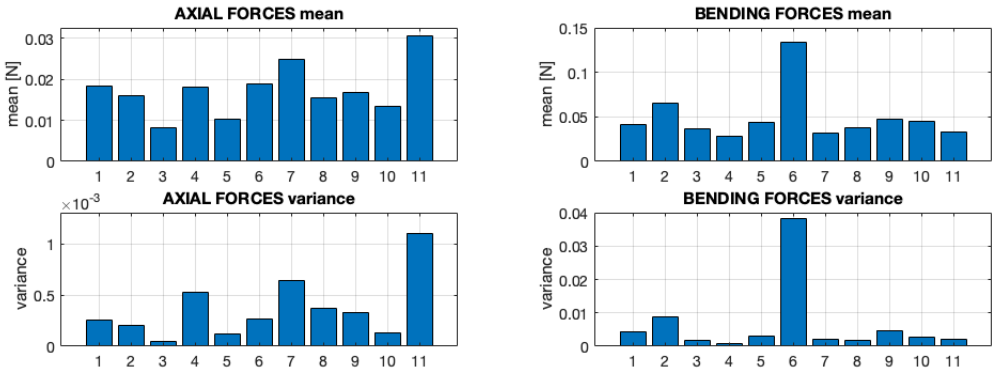
(b) Phase 2 planning, the analysis of the forces is provided in Figure.

Figure 7.34: Planned shapes and intermediate pose for the robotic grippers in the phase 1 and 2, for the Ethernet cable operation



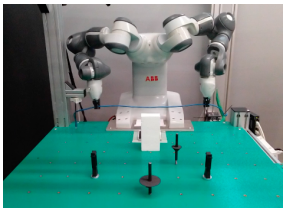
(a) Axial forces analysis. (b) Bending forces analysis.

Figure 7.35: Mean and variance of axial and bending forces on each mass-point of the shape, considering all the planned shapes for the phase-1 of the operation, for the Ethernet cable.

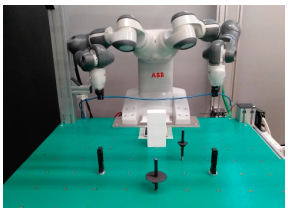


(a) Axial forces analysis. (b) Bending forces analysis.

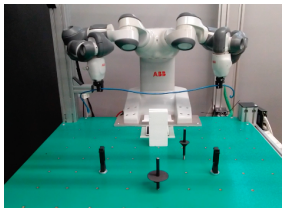
Figure 7.36: Mean and variance of axial and bending forces on each mass-point of the shape, considering all the planned shapes for the phase-2 of the operation, for the Ethernet cable.



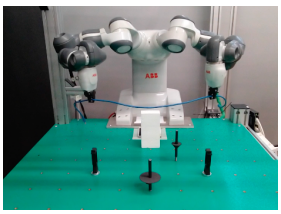
(a) Shape ξ_o.



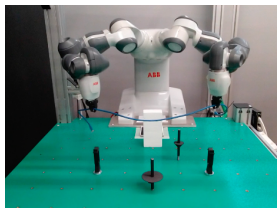
(b) Shape ξ_1.



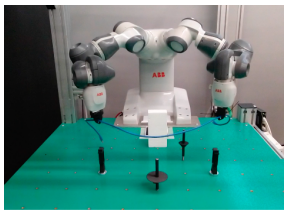
(c) Shape ξ_2.



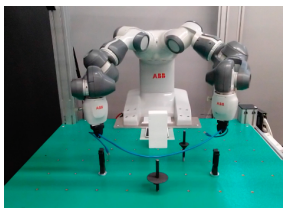
(d) Shape ξ_3.



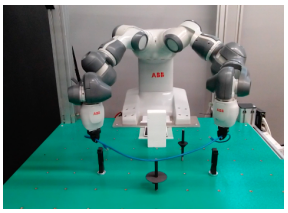
(e) Shape ξ_4.



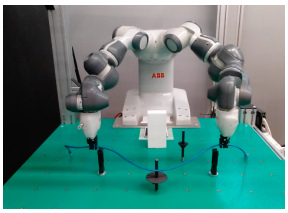
(f) Shape ξ_5.



(g) Shape ξ_f.



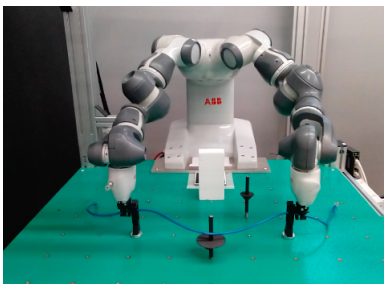
(h) Shape $\xi_{1,alignment}$.



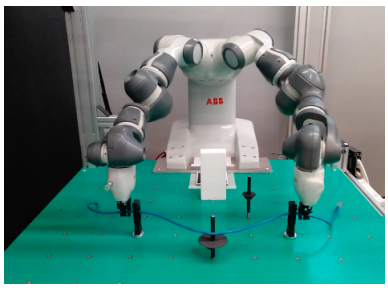
(i) Shape $\xi_{f,alignment}$.

Figure 7.37: Shapes provided by the dual arm robotic manipulation of the Ethernet cable.

The Fixture insertion and the tensioning of the cable are shown in Figure 7.38.



(a) Insertion in the fixtures.



(b) Stretching against the peg.

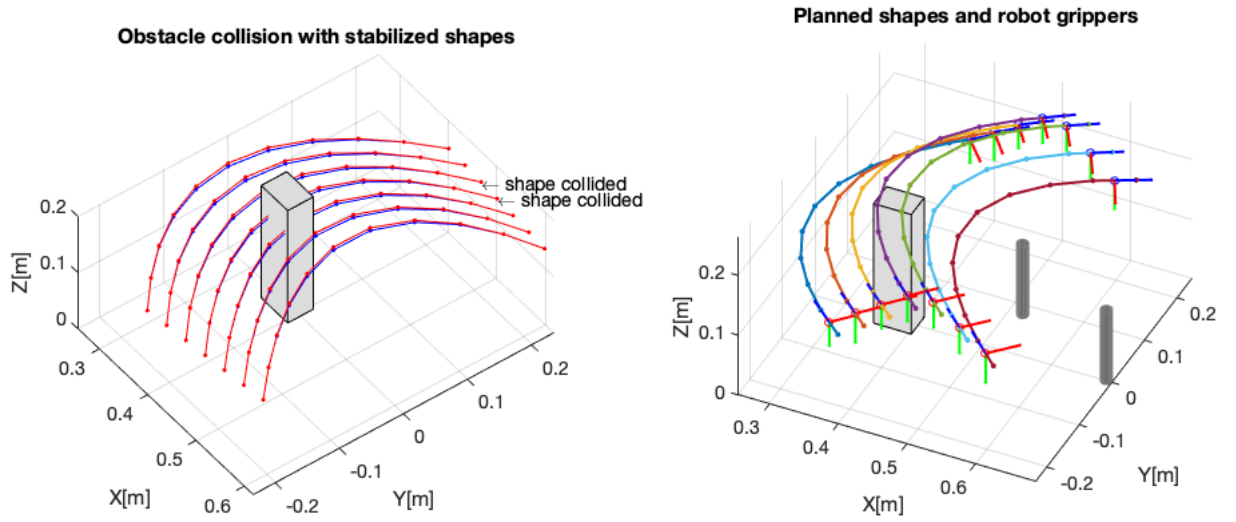
Figure 7.38: Insertion step for the Ethernet cable operation.

Good behaviours are observed during the manipulation, the task is performed with a TCP speed of $100mm/s$, and it's completed in $9s$. The success rate is registered to be 93%. The Ethernet cable is sligthly stiffer than USB, involving less deformations, moreover an higher diameter allows a better gripping of the cable. On the other side the stiffer

behaviour of the cable entails the insertion in the fixtures to be more difficult, due to the resistance of the cable to deform, and due to the creation of stiffer contacts with fixtures and pegs.

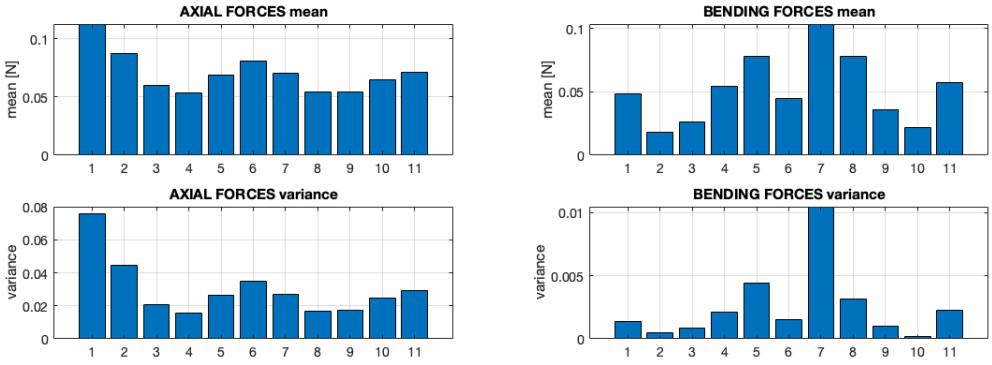
7.3.3. Wire harness assembly for PU Hose

As shown in Figure 7.26c, the PU Hose tube has to be inserted in the fixtures by following the curvature imposed by the other peg on the table. The curvature is not specular, since the peg is not centered between the two fixtures, leading to the introduction of higher internal forces in the cable. Moreover this kind of deformable linear object has a much grater Young's modulus that contributes to the generation of internal forces for a given deformation, making the planning of the manipulation of paramount importance. The planned shapes and the intermediate poses of the grippers for the 1^{st} phase and the 2^{nd} phase of the operation are shown in Figure 7.39 and Figure 7.41. The internal forces of the cable during the operations are provided in Figure 7.40 for the phase 1, and in Figure 7.42 for the phase 2.



(a) Collision detection with stabilized shapes. (b) Re-planned shapes and robotic gripper poses.

Figure 7.39: Hose planning for phase-1 of the operation, two intermediate shapes collided with the obstacle.



(a) Axial forces analysis. (b) Bending forces analysis.

Figure 7.40: Mean and variance of axial and bending forces on each mass-point of the shape, considering all the planned shapes for the phase-1 of the operation, for the Hose.

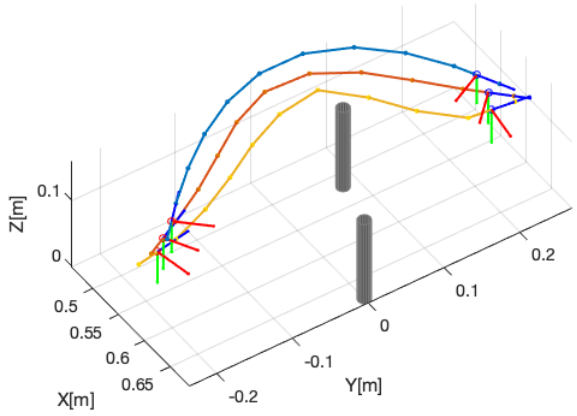
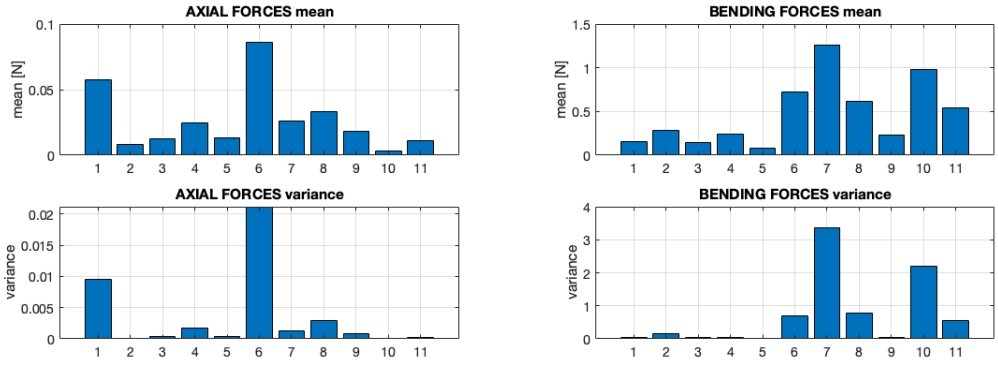


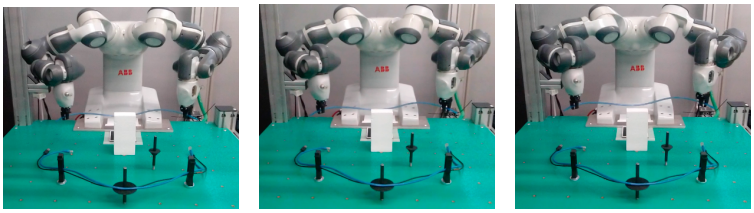
Figure 7.41: PU Hose planning for the phase-2 of the operation.



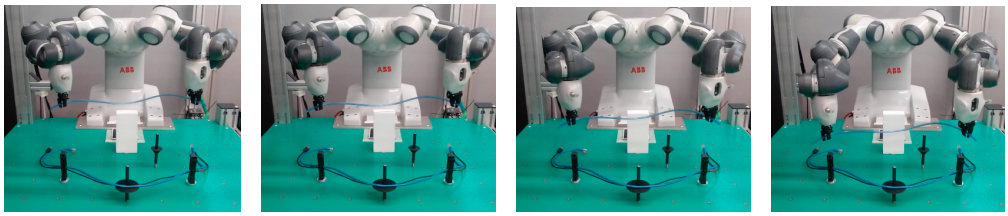
(a) Axial forces analysis. (b) Bending forces analysis.

Figure 7.42: Mean and variance of axial and bending forces on each mass-point of the shape, considering all the planned shapes for the phase-2 of the operation, for the Hose.

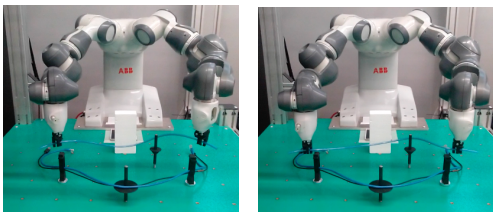
The experienced shapes during the dual arm robotic manipulation are shown in Figure 7.43.



(a) Shape ξ_o. (b) Shape ξ_1. (c) Shape ξ_2.



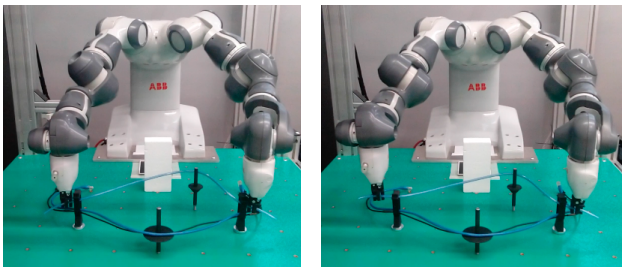
(d) Shape ξ_3. (e) Shape ξ_4. (f) Shape ξ_5. (g) Shape ξ_f.



(h) Shape $\xi_{1,alignment}$. (i) Shape $\xi_{f,alignment}$.

Figure 7.43: Shapes provided by the dual arm robotic manipulation of the Hose.

The Fixture insertion and the tensioning of the cable are shown in Figure 7.44.



(a) Insertion in the fixtures. (b) Stretching against the peg.

Figure 7.44: Insertion step for the Hose operation.

The task is performed with a TCP speed of $100mm/s$, and it's completed in $9s$. The success rate of the operation is registered to be 80%. The failure in the task are mainly

introduced by the high stiffness of the tube and the higher forces introduced in the operation. Those properties leads to more critical behaviours when the deformable linear object is in contact with the pegs and the fixtures, leading to unpredictable behaviours during the insertion phase.

8 | Conclusions

The planning for deformable linear objects manipulation through a dual-arm robot is particularly challenging, due to the large number of degrees of freedom and variables involved in the representation of this kind of objects. In particular the limitations encountered in the modeling phase are projected in the applications of the model, introducing limitations also in the planning phase. Moreover, it is not sufficient to build a more accurate model, because a trade-off between accuracy and efficiency must be reached, where a very accurate model is unnecessarily computationally heavy and impacting on the optimization phase.

Despite the many challenges involved in dealing with deformable linear objects, the method proposes a solution to plan the motion of a dual arm robot to manipulate DLOs, that offers good prospects and possibilities for future industrial applications. In particular, an optimization-based strategy has been developed, that exploits an informative and computationally-cheap model such as the mass-spring model, in order to plan the trajectories for the robot grippers, exploiting the shortest path for the DLO and by keeping it near minimal energy configurations.

Low errors and good behaviours are achieved during the experimental validation. Accurate deformations are experienced on simpler manipulations, while small errors are introduced with the complexity of the shapes to be imposed to the cable. Moreover, during the manipulations, the deformable linear objects appear to be in not-stressed configurations, and the obstacle avoidance is correctly achieved despite the uncertaintly introduced by the model, achieving the goal of performing an implicit planning involving deformable linear objects, even in complex and industrial frameworks.

These behaviours are not affected by the material composing the cable. Thanks to the adaptation strategy based on the stiffness of the DLO, a Young's modulus identification procedure is carried out before the planning phase, allowing to determine the equivalent Young's modulus for composite cables. This makes the method robust and generalizable to different kinds of cables and tubes, thanks to an adaption strategy based on the Young's modulus for the optimization phase and the simulation one.

The only limit is represented by the speed of manipulation: in order to face the complex-

ity of the model and filter out uncertainties introduced by dynamical behaviours such as inertia contributes during the manipulation, the planning aims to generate and analyse quasi-static intermediate configurations for the deformable linear object. For this reason the velocities related to the gripper trajectories can't be too high. However, in the use-case validation, good time cycle are achieved, performing a wire assembly operation in less than 10 seconds.

8.1. Future developments

Despite the good results obtained, and the relevance of the method in an industrial context, some future works can be outlined.
A first improvement can be introduced by applying a more accurate model to the optimization technique, such as the discrete elastic rod model: the optimization phase inspects quasi-static configurations of the model by exploiting the forces and the positions of the mass-points. It is possible to replace the model that provides those values with a more accurate one, without changing the optimization structure. On the other hand it's important to say that a more accurate and detailed model may bring issues related to the optimization, leading to a much complex problem to be solved.
Other improvements can be introduced by working on a control structure for the manipulation. This work aims to provide a path to be followed by the deformable linear object, and the actuation variables are imposed to be the poses of the grippers holding the cable. While in this work the main objective is to perform an offline computation of the grippers trajectories with a model-based analysis of the cable behaviours, an effective improvement can be achieved by building an online control scheme for the shaping of the cable, considering the shapes planned in this work as reference shapes. A first example can be a visual servoing strategy: a tracking algorithm of the cable in the space is necessary, then a local model and a deformation control strategy must be carried out. An other possibility is the usage of a vision sensor in order to inspect if the obtained shapes are matching with the planned ones, and if that is not true, a recovery strategy can be developed.
Finally an improvement can be achieved by the application of reliable force sensors and actuators on the robotic grippers. This allows the possibility of closing an additional control loop based on the planned forces exploited by the mass-spring model.

Bibliography

[1] Viscous damping ratio for different systems and material. URL `https://help.solidworks.com/2016/english/solidworks/cworks/r_viscous_damping_ratios.htm`.

[2] M. Bergou, M. Wardetzky, S. Robinson, B. Audoly, and E. Grinspun. Discrete elastic rods. *ACM Trans. Graph.*, 27(3):1–12, aug 2008. ISSN 0730-0301.

[3] T. Bretl and Z. McCarthy. Quasi-static manipulation of a kirchhoff elastic rod based on a geometric analysis of equilibrium configurations. *International Journal of Robotics Research*, 33(1):48–68, Jan. 2014. ISSN 0278-3649. doi: 10.1177/0278364912473169.

[4] D. De Gregorio, G. Palli, and L. Di Stefano. Let's take a walk on superpixels graphs: Deformable linear objects segmentation and model estimation. In C. V. Jawahar, H. Li, G. Mori, and K. Schindler, editors, *Computer Vision – ACCV 2018*, pages 662–677, Cham, 2019. Springer International Publishing. ISBN 978-3-030-20890-5.

[5] D. R. Haumann and R. E. Parent. The behavioral test-bed: Obtaining complex behavior from simple rules. *The Visual Computer*, 4:332–347, 2005.

[6] R. Herguedas, G. López-Nicolás, R. Aragüés, and C. Sagüés. Survey on multi-robot manipulation of deformable objects. In *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 977–984, 2019. doi: 10.1109/ETFA.2019.8868987.

[7] T. Hermansson, R. Bohlin, J. S. Carlson, and R. Söderberg. Automatic assembly path planning for wiring harness installations. *Journal of Manufacturing Systems*, 32(3):417–422, 2013. ISSN 0278-6125.

[8] P. Kaufmann, S. Martin, M. Botsch, and M. Gross. Flexible simulation of deformable models using discontinuous galerkin fem. *Graphical Models*, 71:153–167, 01 2009.

[9] A. Loock and E. Schömer. A virtual environment for interactive assembly simulation: From rigid bodies to deformable cables. In *Proceedings of the 5th World Multiconfer-*

ence on Systemics, Cybernetics and Informatics (SCI-01): Emergent computing and virtual engineering, volume 3, pages 325–332, USA, 2001. International Institute of Informatics and Systemics.

[10] N. Lv, J. Liu, X. Ding, J. Liu, H. Lin, and J. Ma. Physically based real-time interactive assembly simulation of cable harness. *Journal of Manufacturing Systems*, 43:385–399, 2017. ISSN 0278-6125.

[11] M. Moll and L. Kavraki. Path planning for deformable linear objects. *IEEE Transactions on Robotics*, 22(4):625–636, 2006. doi: 10.1109/TRO.2006.878933.

[12] O. Roussel, A. Borum, M. Taïx, and T. Bretl. Manipulation planning with contacts for an extensible elastic rod by sampling on the submanifold of static equilibrium configurations. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3116–3121, 2015. doi: 10.1109/ICRA.2015.7139627.

[13] A. Sintov, S. Macenski, A. Borum, and T. Bretl. Motion planning for dual-arm manipulation of elastic rods. *IEEE Robotics and Automation Letters*, 5(4):6065–6072, 2020. doi: 10.1109/LRA.2020.3011352.

[14] M. Yu, H. Zhong, and X. Li. Shape control of deformable linear objects with offline and online learning of local linear deformation models. *CoRR*, abs/2109.11091, 2021.

[15] Y. Yu, X. Wang, and Z. Chen. A simplified finite element model for structural cable bending mechanism. *International Journal of Mechanical Sciences*, 113:196–210, 2016. ISSN 0020-7403.

[16] J. Zhu, B. Navarro, P. Fraisse, A. Crosnier, and A. Cherubini. Dual-arm robotic manipulation of flexible cables. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 479–484, 2018. doi: 10.1109/IROS.2018.8593780.

[17] J. Zhu, B. Navarro, R. Passama, P. Fraisse, A. Crosnier, and A. Cherubini. Robotic manipulation planning for shaping deformable linear objects withenvironmental contacts. *IEEE Robotics and Automation Letters*, 5(1):16–23, 2020. doi: 10.1109/LRA.2019.2944304.

List of Figures

List of Tables