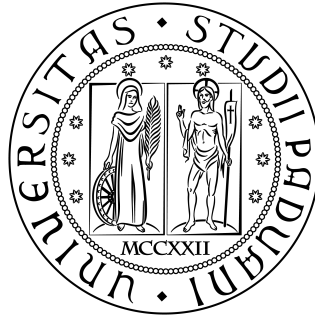




UNIVERSITY OF PADUA

DEPARTMENT OF INFORMATION ENGINEERING

MASTER'S DEGREE IN
CONTROL SYSTEMS ENGINEERING

**Adaptive Control of Deformable Linear Objects with
Physical Parameter Estimation**

Supervisor

Prof. Valcher Maria Elena

University of Padova

Co-supervisor

Prof. Della Santina Cosimo

Trumic Maja

Feliu-Talegon Daniel

Technical University of Delft

Candidate

Capraro Giovanni

ID: 2158516

ACADEMIC YEAR 2025/2026



Contents

1	Sommario	9
1	Abstract	11
2	Introduction	13
3	Dynamic model of the DLO	21
3.1	GVS Modeling	21
3.1.1	Strain parametrisation	22
3.1.2	Recursive kinematics	23
3.1.3	Lagrangian dynamics	24
3.1.4	Revolute joint + DLO	25
3.1.5	Collocated Coordinates	27
3.1.6	Single Arm robot + DLO	28
3.1.7	Bimanual robot + DLO	33
4	Adaptive Control	37
4.1	Structural Properties and Control Law	37
4.1.1	Properties proof	37
4.1.2	Problem statement	42
4.1.3	Parameter Identification	45
5	Simulations	49
5.1	Results	49
5.1.1	Fixed-Arm	50

5.1.2	Single-Arm Manipulation	57
5.1.3	Dual-Arm	63
6	Experimental Validation	73
6.0.1	Experimental Setup	73
6.0.2	Control Law	75
6.0.3	Results	75
7	Conclusion	83
8	Appendices	85
8.1	GVS Model derivation	85
8.1.1	Key structural elements	85
8.1.2	Recursive kinematics: from strain to velocity and acceleration	86
8.1.3	Recovering absolute velocity and acceleration	88
8.1.4	Discretized strain and the Jacobian recursion	88
8.1.5	Lagrangian dynamics	88
8.2	Properties demonstration with generalized coordinates	90

0

List of Figures

2.1	Cosserat Rod representation of a DLO and its forces-momentum distribution along its length. Adapted from Armanini [1].	14
2.2	Representation of all possible strain states of the DLO, considering one active strain at a time. Adapted from Della Santina [7].	15
3.1	Structure of the i -th soft link used in the reconstruction of the system's kinematics. Adapted from Matthews' [13]	23
3.2	Revolute joint + DLO.	25
3.3	Representation of the two-nested quadrature scheme, used in the integration of a soft link.	26
3.4	DLO grasped at one end by a manipulator	29
3.5	DLO grasped at both ends by a bimanual robot	34
5.1	Desired actuated-coordinate regulation using the adaptive controller from the initial configuration $\theta_a = \pi/3$. Panel (a) shows the tracking performance of the actuated coordinate, while panel (b) reports the corresponding velocity response and sliding-variable errors.	51
5.2	Control effort and adaptive parameter estimation for the desired actuated-coordinate regulation task shown in Fig. 5.1a, using the adaptive controller. Panel (a) reports the control torque required to achieve the observed tracking performance, while panel (b) shows the evolution of the estimated dynamic parameters starting from the initial estimate $\pi(0) = [10^3, 4 \cdot 10^5, 6 \cdot 10^5]$	51
5.3	Regulation task comparison among the Adaptive (blue), PID (green), and PD+ (red) controllers against the reference (orange), in the nominal case (DLO2). Panel (a) shows the actuated-coordinate tracking together with the tracking error, while panel (b) shows the corresponding control torque for the three strategies.	52
5.4	Regulation task comparison among the Adaptive (blue), PID (green), and PD+ (red) controllers against the reference (orange), considering DLO3. Panel (a) shows the actuated-coordinate tracking together with the tracking error, while panel (b) shows the corresponding control torque for the three strategies.	53

5.5	Regulation task comparison among the Adaptive (blue), PID (green), and PD+ (red) controllers against the reference (orange), considering DLO4. Panel (a) shows the actuated-coordinate tracking together with the tracking error, while panel (b) shows the corresponding control torque for the three strategies.	53
5.6	Adaptive controller performance across the four DLOs (DLO1–DLO4) considered in this work, completing the set-point tracking task from the same initial condition $\theta_a = \pi/3$. Panel (a) shows the actuated-coordinate tracking together with the tracking error, while panel (b) shows the corresponding control torque for each object.	54
5.7	Parameter estimation comparison considering the 4 kind of DLO, starting from the initial condition $\pi(0) = [10^3, 4 \cdot 10^5, 6 \cdot 10^5]$	54
5.8	Tip-regulation task performance using the adaptive controller with the inverse kinematics algorithm based on the fully known model. Panel (a) shows the evolution of the Cartesian tip coordinates toward the desired position, while panel (b) shows the corresponding actuated-coordinate evolution and tracking error.	55
5.9	Velocity-level behaviour and parameter estimation for the tip-regulation task shown in Fig. 5.8a. Panel (a) reports the actuated and unactuated velocity components together with the corresponding sliding-variable errors, while panel (b) shows the evolution of the estimated dynamic parameters starting from the initial estimate $\pi(0) = [10^3, 4 \cdot 10^5, 6 \cdot 10^5]$	55
5.10	DLO configuration evolution in time while achieving the tip’s regulation task	56
5.11	DLO configuration evolution and parameter estimation for the tip-regulation task solved with the inverse kinematics algorithm based on the online parameter-estimate update (Eq. 5.4). Panel (a) shows the resulting configuration evolution of the DLO, while panel (b) shows the corresponding parameter estimation starting from the initial estimate $\pi(0) = [10^3, 4 \cdot 10^5, 6 \cdot 10^5]$	57
5.12	Base-pose tracking and parameter estimation for the moving-arm scenario, using the adaptive controller to converge to a sinusoidal trajectory varying along the x-axis. Panel (a) shows the evolution of the base orientation and position, while panel (b) shows the evolution of the estimated dynamic parameters starting from the initial estimate $\pi(0) = [10^3, 4 \cdot 10^5, 6 \cdot 10^5]$	58
5.13	Actuated-velocity and sliding-variable behaviour for the moving-arm sinusoidal-tracking task shown in Fig. 5.12a. Panel (a) shows the evolution of the actuated velocity components, while panel (b) shows the corresponding sliding-error evolution for both the actuated and unactuated components.	58
5.14	Base-pose and control-torque comparison among the Adaptive (blue), PD+ (orange), and PID (purple) controllers completing the configuration-convergence task for the moving-arm scenario, in the nominal case (DLO2). Panel (a) shows the base-pose evolution, while panel (b) shows the corresponding control torque for the three strategies.	59
5.15	Base-pose and control-torque comparison among the Adaptive (blue), PD+ (orange), and PID (purple) controllers completing the configuration-convergence task for the moving-arm scenario, considering DLO3. Panel (a) shows the base-pose evolution, while panel (b) shows the corresponding control torque for the three strategies.	59
5.16	Base-pose and control-torque comparison among the Adaptive (blue), PD+ (orange), and PID (purple) controllers completing the configuration-convergence task for the moving-arm scenario, considering DLO4. Panel (a) shows the base-pose evolution, while panel (b) shows the corresponding control torque for the three strategies.	60
5.17	Adaptive controller performance across the four DLOs (DLO1–DLO4) considered for the moving-arm scenario. Panel (a) shows the base-pose evolution, while panel (b) shows the corresponding control torque for each object.	61
5.18	Parameter estimation and sliding-variable behaviour across the four DLOs (DLO1–DLO4) considered for the moving-arm scenario. Panel (a) shows the evolution of the estimated dynamic parameters, while panel (b) shows the corresponding sliding-error evolution.	61

5.19	Zoom of the Last five second evolution of the estimation of the material's damping in the scenario compating different DLOs	61
5.20	Tip-settling task performance for the moving-arm scenario using the adaptive controller with the inverse kinematics algorithm based on the fully known model (Eq. 5.3). Panel (a) shows the evolution of the Cartesian tip coordinates, while panel (b) shows the corresponding actuated-coordinate evolution.	62
5.21	Sliding-variable behaviour and parameter estimation for the tip-settling task shown in Fig. 5.20. Panel (a) shows the evolution of the sliding error, while panel (b) shows the corresponding parameter estimation.	62
5.22	Time Evolution of the single arm DLO configuration while achieving tip's regulation	62
5.23	Actuated-coordinate evolution and parameter estimation for the tip-regulation task solved with the inverse kinematics algorithm based on the online parameter-estimate update (Eq. 5.4). Panel (a) shows the actuated-coordinate evolution required to reach the desired tip position, while panel (b) shows the corresponding parameter estimation.	63
5.24	Time Evolution of the single arm DLO configuration while achieving tip's regulation, using the estimation of the parameters for computing the actuated coordinate	63
5.25	Position and orientation evolution of both end-points of the nominal DLO (DLO2). Considering the sinusoidal trajectory tracking with initial configuration $\theta = [0, 0, 0.1, 0.035, 0.09, 0, 0.65, 0.87, 0, 0.1]^T$	64
5.26	Control effort and parameter convergence under the fully-actuated adaptive controller in the nominal case DLO (DLO2), with parameter initial condition $\pi(0) = [10^3, 4 \cdot 10^5, 6 \cdot 10^5]$	65
5.27	Velocity-level tracking and sliding-variable convergence under the fully-actuated adaptive controller in the nominal case DLO (DLO2).	65
5.28	Pose tracking comparison for DLO2 considering the three different control actions– Adaptive (Blue), PD+cancellation (Orange), PsatID(Green), Reference(Purple).	66
5.29	Control-effort and velocity comparison for DLO2 considering the three different control actions– Adaptive (Blue), PD+cancellation (Orange), PsatID(Green).	66
5.30	Pose tracking comparison for DLO3 considering the three different control actions– Adaptive (Blue), PD+cancellation (Orange), PsatID(Green), Reference(Purple).	67
5.31	Control-effort and velocity comparison for DLO3 considering the three different control actions– Adaptive (Blue), PD+cancellation (Orange), PsatID(Green).	67
5.32	Pose tracking comparison for DLO4 considering the three different control actions– Adaptive (Blue), PD+cancellation (Orange), PsatID(Green), Reference(Purple).	67
5.33	Control-effort and velocity comparison for DLO4 considering the three different control actions– Adaptive (Blue), PD+cancellation (Orange), PsatID(Green).	67
5.34	Adaptive controller performance across the four DLOs (DLO1–DLO4) considered for the dual-arm scenario. Panel (a) shows the position evolution of base and tip, while panel (b) shows the orientation evolution for each object.	68
5.35	Parameters convergence and control effort across the four DLOs (DLO1–DLO4) considered for the dual-arm scenario. Panel (a) shows the parameters evolution, while panel (b) shows the corresponding control torque for each object.	69
5.36	Sliding-errors convergence and velocities across the four DLOs (DLO1–DLO4) considered for the dual-arm scenario. Panel (a) shows the sliding errors evolution, while panel (b) shows the corresponding actuated velocities for each object	69
5.37	Pose evolution comparison considering the under-actuated case(Orange) and the fully-actuated case(blue) while completing a regulation task.	70
5.38	Comparison of the Torque evolution and the Parameter estimation considering the under-actuated case(Orange) and the fully-actuated case(blue) while completing a regulation task.	70
5.39	Pose comparison considering the fully-actuated case (blue), the under-actuated case 3^{rd} order (Orange) and the under-actuated case 5^{th} order(Green) while completing a regulation task.	71

5.40	Comparison of the Torque evolution and the Parameter estimation considering the fully-actuated case (blue), the under-actuated case 3^{rd} order (Orange) and the under-actuated case 5^{th} order (Green) while completing a regulation task.	71
5.41	Actuated Coordinate associated to the deforming points and Configuration evolution of the nominal case DLO (DLO2) in the dual arm model	72
5.42	Parameter estimation and sliding error evolution in time related to the configuration control in the dual arm model. Panel (b) we see the known result of convergence to zero of the sliding error. . .	72
6.1	The experimental setup used in the validation cases proposed.	74
6.2	Actuated coordinate performance considering different initial conditions of the parameter values- Blue for the parameter values with nominal ratio 0.1, Orange for the parameter values with nominal ratio 2.0	76
6.3	Parameter evolution and steady state error behaviour analysis- Blue for the parameter values with nominal ratio 0.1, Orange for the parameter values with nominal ratio 2.0	76
6.4	Actuated coordinate performance considering different control strategies-Blue for the adaptive control action, Orange for the PD+ Cancellation method, and Green for the PD control action . . .	77
6.5	Average Steady-state error comparison, with steady state error associate to each step of the three different control strategies	77
6.6	Average steady state error comparison between an wrongly model DLO, namely, an offset in the characteristics of the initial condition of the physical parameters	78
6.7	Actuated coordinate performance considering different control strategies-Blue for the adaptive control action, Orange for the PD+ Cancellation method, and Green for the PD control action . . .	78
6.8	Average Steady-state error comparison, with steady state error associate to each step of the three different control strategies	79
6.9	Actuated coordinate performance considering different initial conditions of the physical parameter values, while completing the trajectory tracking task.	79
6.10	Parameter estimation evolution and total error generated considering different initial conditions of the physical parameter values, while completing the trajectory tracking task.	80
6.11	Actuated coordinate performance considering different control strategies while completing the trajectory tracking task.	80
6.12	Total error generated by each control strategy while completing the trajectory tracking task. . . .	80
6.13	Unactuated component control evolution in time, based in the inverse kinematics algorithm	81

1

Sommario

La manipolazione robotica di Oggetti Lineari Deformabili (DLO), quali cavi, tubi flessibili e suture chirurgiche, rappresenta un'esigenza crescente nell'automazione industriale e medica. I controllori basati sul modello dell'oggetto offrono garanzie formali di stabilità ed elevate prestazioni di inseguimento, ma la loro efficacia dipende da una conoscenza accurata delle proprietà fisiche del DLO: densità volumetrica, modulo di Young e smorzamento elastico del materiale. Nella pratica, questi parametri sono raramente noti a priori per un dato esemplare e sono difficili da identificare offline, limitando la praticabilità dei progetti nominali puramente basati sul modello. Questa tesi propone uno schema di controllo adattativo che, simultaneamente, insegue una traiettoria desiderata del DLO e stima online i parametri fisici incogniti, eliminando la necessità di un'identificazione preliminare. Il DLO è modellato come un'asta di Cosserat discretizzata mediante il metodo GVS (Geometric Variable Strain), che fornisce una rappresentazione Lagrangiana a dimensione finita. L'obiettivo di questo lavoro è formulare un controllore per un sistema sottoattuato in forma collocata, la cui legge adattativa estenda lo schema di Slotine-Li al caso sottoattuato. Vengono studiate tre configurazioni di complessità crescente: un DLO collegato a un giunto rotante fisso, un DLO manipolato da un singolo braccio robotico a un'estremità, e un DLO afferrato a entrambe le estremità da un sistema bimanuale. Per ciascuna configurazione viene derivato un cambiamento di coordinate in forma collocata e il controllore adattativo viene applicato in modo adeguato al modello, tenendo conto delle differenze intrinseche tra i sistemi analizzati. La convergenza asintotica dell'errore è garantita tramite analisi di Lyapunov. I risultati di simulazione confermano che tutti e tre i parametri del materiale sono stimati accuratamente durante la manipolazione closed-loop, dimostrando la fattibilità del framework proposto per il controllo di DLO precedentemente non caratterizzati. In fine, è stato trattato in diversi esperimenti il modello di un DLO manipolato da un singolo braccio, confermando i risultati ottenuti in simulazione.

1

Abstract

Robotic manipulation of Deformable Linear Objects (DLOs) — cables, hoses, and surgical sutures — is a growing requirement in industrial and medical automation. Model-based controllers offer formal stability guarantees and high tracking performance, but their effectiveness depends on accurate knowledge of the DLO’s physical properties: volumetric density, Young’s modulus, and elastic material damping. In practice these parameters are rarely known a priori for a given specimen and are difficult to identify offline, limiting the deployability of purely nominal model-based designs. This thesis proposes an adaptive control framework that simultaneously tracks a desired DLO trajectory and estimates the unknown physical parameters online, eliminating the need for prior identification. The DLO is modelled as a Cosserat rod discretized by the Geometric Variable Strain (GVS) method, which yields a finite-dimensional Lagrangian representation in which three material parameters appear linearly in the system matrices. The manipulation task is formulated as the control of an underactuated Lagrangian system in collocated form, and the adaptive law extends the Slotine–Li scheme to the underactuated setting. Three configurations of increasing complexity are studied: a DLO attached to a fixed revolute joint, a DLO manipulated by a single robotic arm at one end, and a DLO grasped at both ends by a bimanual system. For each configuration, a change of coordinates to collocated form is derived and the adaptive controller is applied uniformly. Asymptotic stability of the tracking error is guaranteed via Lyapunov analysis. Simulation results confirm that all three material parameters are accurately estimated during closed-loop manipulation, demonstrating the viability of the proposed framework for the control of previously uncharacterized DLOs. Lastly, a broad series of experiments were conducted to validate the results considering the single-arm manipulated DLO, achieving and confirming the results obtained in simulation.

2 Introduction

Motivation: Robots in a Deformable World

Robotic manipulation research has historically concentrated on rigid objects, whose motion is fully described by a finite set of rigid-body transformations and whose dynamics are well-understood [3, 6, 19]. Yet the physical world is predominantly non-rigid: cables, hoses, garments, food products, sutures, and a host of other objects encountered in industry, medicine, and everyday life deform continuously under applied forces. Automating the handling of such objects, a capability indispensable for the full deployment of robots in human environments, requires confronting a qualitatively different class of problems in modeling, perception, planning, and control [7, 23].

The contrast with rigid manipulation is fundamental. When a force is applied to a rigid body, the result is a wrench that drives a well-defined sequence of motions in $SE(3)$. When the same force is applied to a deformable body, it simultaneously displaces and reshapes the object; the precise outcome depends on the material's internal constitution, its stiffness, mass distribution, and energy dissipation. Material properties thus become an integral part of the system dynamics; yet these properties are rarely known with sufficient accuracy for a given specimen. High-fidelity mechanical models, such as those based on finite elements (FEM), require detailed knowledge of the material, knowledge that is unavailable to a robot encountering an object for the first time. This dual challenge of accurate modeling under parameter uncertainty motivates the adaptive approach developed in this thesis.

The industrial and societal relevance of this problem is broad. In automotive and aerospace manufacturing, wiring harnesses and hydraulic hoses must be routed and fastened with millimeter-level precision. In surgical robotics, sutures and flexible endoscopes must be manipulated within confined anatomical spaces where forces must be tightly controlled. In domestic service

robotics, flexible objects such as cables and garments pervade everyday tasks. Despite considerable research effort, reliable, physics-aware robotic manipulation of deformable objects remains an open problem [1].

Deformable Linear Objects: Ubiquity and Unique Challenges Among the broad family of deformable objects, Deformable Linear Objects (DLOs) constitute a class of particular relevance to industrial automation. The term encompasses cables, electrical wires, ropes, surgical sutures, pneumatic hoses, and flexible pipes, items whose defining geometric property is slenderness: their spatial configuration evolves primarily along a single curvilinear dimension [2].

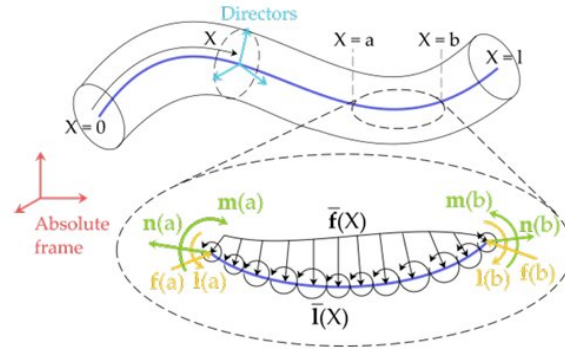


Figure 2.1: Cosserat Rod representation of a DLO and its forces-momentum distribution along its length. Adapted from Armanini [1].

This apparent simplicity belies a profound complexity. Because a DLO is a continuously deformable body, its full configuration space is in principle infinite-dimensional; even approximating it requires careful modeling choices. Moreover, large continuous deformations (bending, twisting, stretching) are commonplace during manipulation, making linear or small-deformation approximations inadmissible for accurate simulation and control.

The literature on DLO manipulation can broadly be organized into three paradigms. A first family of approaches treats the object as quasi-static, focusing on geometric shape control under the assumption that inertial and damping effects are negligible [10, 22]. Although tractable, this assumption fails for objects with significant stiffness or mass, or under rapid manipulation movements, precisely the conditions encountered with industrial DLOs such as thick cables and pressurized hoses. A second paradigm leverages model-free learning strategies, sidestepping the modeling challenge at the cost of data efficiency, generalizability, and formal stability guarantees [5, 11, 12]. The third, and most principled, family is the physics-based, model-based approach: the DLO dynamics are explicitly captured within a tractable reduced-order model, and that model is exploited directly in the design of the controller. This approach provides a principled way to account for the physical properties and dynamic behavior of the manipulated object, while avoiding the restrictive quasi-static assumptions of the first paradigm and the extensive data requirements of model-free learning. It is this third paradigm that this thesis pursues, extending it to the adaptive setting where three physical parameters of the system are unknown.

A recent and important contribution in this area is the model-based closed-loop control

framework proposed for DLOs gripped at multiple locations, which treats manipulation as a shape-regulation problem with formal convergence guarantees [8]. That work demonstrates that, when object dynamics are explicitly modeled and the problem is cast as an underactuated control problem, controllers with provable stability properties can be derived. The present thesis builds on this foundation and addresses its principal limitation: the need for accurate knowledge of the DLO's physical parameters.

Why the Geometric Variable Strain Model? Any physics-based controller for DLOs must be grounded in a dynamic model that balances accuracy, interpretability, and computational tractability. The most general continuum mechanics framework for slender deformable bodies is the Cosserat rod model, which captures all six modes of deformation, bending in two planes, axial extension, axial twist, and shear in two directions, in a geometrically exact formulation that is valid for arbitrarily large deformations [1].

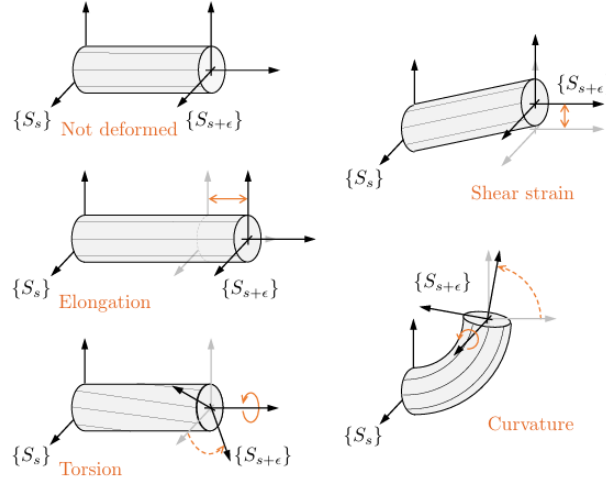


Figure 2.2: Representation of all possible strain states of the DLO, considering one active strain at a time. Adapted from Della Santina [7].

Directly integrating the Cosserat partial differential equations in a real-time control loop is, however, computationally prohibitive. This motivates the development of reduced-order models (ROMs) that represent the infinite-dimensional state space by a finite, minimal set of generalized coordinates.

Strain-based ROMs have emerged as a particularly attractive approach in soft and continuum robotics [1]. Rather than interpolating positions or orientations directly, strain-based models discretize the strain field, the spatial rate of change of the rod's pose, and recover the rod configuration by spatial integration. This strategy preserves the geometric exactness of the Cosserat formulation, avoids the objectivity issues associated with direct orientation interpolation, and yields accurate approximations with very few degrees of freedom.

Among the available strain-based ROMs, the Geometric Variable Strain (GVS) model is chosen as the modeling framework for this thesis [13]. The GVS model expresses the strain field as a linear combination of user-defined basis functions, with the corresponding generalized

coordinates evolving according to Lagrangian dynamics derived via the Magnus expansion. Its key properties make it ideal for the control application considered here. First, the GVS model is geometrically exact and supports all deformation modes of a Cosserat rod. Second, it yields a compact, finite-dimensional representation whose accuracy can be systematically improved by increasing the number of basis functions. Third, and crucially for this work, its Lagrangian structure enables direct application of analytical mechanics tools, in particular, the collocated coordinate transformation and the adaptive control theory on which this thesis builds [14, 18].

In the formulation adopted here, the DLO is modeled as a floating-base Cosserat rod subject to wrenches applied at one or multiple contact points by the robotic manipulator. The GVS discretization yields a set of second-order ordinary differential equations in Lagrangian form, whose mass matrix, Coriolis terms, gravitational potential, and elastic restoring forces depend on the DLO's physical parameters, namely, radius, length, volumetric mass density, Young's modulus, and elastic damping coefficient. This dependence is the entry point for the adaptive estimation scheme developed in subsequent chapters.

The Need for Adaptive Control: Unknown Physical Parameters A model-based controller grounded in the GVS dynamics requires numerical values for the DLO's physical parameters that are not assumed to be known: the volumetric density ρ , which governs the distribution of inertia along the rod; Young's modulus E , which characterizes the elastic restoring forces against bending, extension, and shear; and the elastic material damping coefficient, which determines the rate at which mechanical energy is dissipated during deformation. In practice, all three quantities are rarely known with sufficient accuracy for a given cable or hose, as they depend on the material composition, cross-sectional geometry, and manufacturing history of the specific specimen. They may vary across objects of the same nominal type or change over the operational lifetime of the object [2, 4].

The naïve response to use nominal values obtained from material data sheets introduces systematic modeling errors that degrade both trajectory tracking performance and the validity of any formal stability guaranty. Offline identification, in which the DLO's parameters are estimated through dedicated experiments prior to the control task, mitigates this issue but is costly in time and instrumentation, requires a controlled experimental setup that may not be available at deployment time, and does not transfer across objects. Another possible solution to the problem can be developed using a model free approach, as in [5]. In this work, a two-phase solution is used: first, it builds a general DLO model, and then it improves it by running a gradient method to perform both the control action and adapt the DLO parameters. A more principled solution is provided by adaptive control [9, 14].

Adaptive control integrates parameter estimation directly into the feedback loop. A set of parameter estimates is maintained and updated online, driven by the discrepancy between the observed and predicted system behavior. Under suitable conditions on the system structure and the richness of the reference trajectory, namely, the issue of input excitation; the adaptive law drives the state tracking error to zero and converges the parameter estimates toward the true values [14]. The foundational result in this direction for mechanical systems is the Slotine-Li

adaptive controller [20], which exploits the linear parameterization of the inverse dynamics of fully actuated Lagrangian systems with respect to a set of constant base parameters. The control and update laws follow from a Lyapunov analysis that guarantees asymptotic stability of the tracking error and boundedness of the parameter estimates, without requiring direct measurement of accelerations.

In this thesis, the parameters to be identified online are the volumetric density, the Young's modulus, and the elastic damping coefficient of the DLO. The simultaneous online estimation of these three quantities during closed-loop shape-trajectory tracking constitutes the parameter estimation contribution of this work, eliminating the need for prior offline identification and yielding a controller that is immediately deployable on a previously unseen DLO.

Adaptive Control of Underactuated Systems in Collocated Form Extending adaptive control from fully actuated to underactuated systems is not straightforward, and understanding why requires examining the structure of DLO manipulation more carefully. The wrenches exerted by the robotic manipulator at the contact points act directly on a subset of the DLO's generalized coordinates, those associated with the actuation locations, while the remaining coordinates, which describe the distributed internal deformation of the DLO, are only indirectly influenced through elastic and inertial coupling. The number of independent control inputs is therefore strictly less than the number of degrees of freedom; thus, the system is underactuated.

Under-actuation introduces a structural obstacle to the direct application of the Slotine–Li adaptive scheme. In that scheme, the linearity of the inverse dynamics with respect to the base parameters, the key property that enables the derivation of the update law, holds globally for fully actuated systems. For underactuated systems expressed in the standard form, the collocated inverse dynamics (the portion of the inverse dynamics associated with the actuated coordinates) depend nonlinearly on the non-actuated accelerations, precluding direct application of classical adaptive control schemes [20].

Two recent results together resolve this difficulty and form the theoretical backbone of this thesis. The first, entitled “Input Decoupling of Lagrangian Systems via Coordinate Transformation” [16], establishes that for a class of mechanical systems called collocated systems, which includes DLOs manipulated by forces applied at specific locations, there exists a change of generalized coordinates, called actuation coordinates, under which the control inputs appear in a fully decoupled form in the equations of motion. In these coordinates, each actuator directly drives one and only one equation of motion, and the input matrix takes the canonical form $\begin{bmatrix} I \\ 0 \end{bmatrix}$. This collocated representation is not merely a computational convenience: it exposes the natural partition of the degrees of freedom into directly actuated and indirectly influenced coordinates, and it is precisely the representation in which the adaptive control law can be derived.

The second key result, “Collocated Adaptive Control of Underactuated Mechanical Systems” [14], extends the Slotine–Li adaptive controller to the underactuated case when the system is expressed in collocated form. The central insight, inspired by back-stepping, is to redefine the sliding variable as the difference between the system's velocity and an exogenous reference state,

whose dynamics are treated as a control input. This redefinition restores a form of the inverse dynamics that is linear in the base parameters, enabling the derivation of an adaptive update law. The stability of the closed-loop system and the convergence of the collocated tracking error are established in the sense of Lyapunov, without requiring acceleration measurements and without any approximation of the system's nonlinear structure.

This thesis applies this combined theoretical framework, GVS modeling of the DLO, transformation to collocated coordinates, and collocated adaptive control, to the problem of DLO shape-trajectory tracking with simultaneous physical parameter estimation. The GVS-discretized DLOs dynamics, once expressed in actuation coordinates, satisfy the conditions required by the collocated adaptive controller of [14], making the overall design theoretically sound and the stability analysis rigorous. To develop this work, basic assumptions were set: the radius along the circular DLOs is considered constant, the physical parameters of different DLOs are considered constant along the abscissa of the object, with values reported in Tab.5.1. The three models presented later share the model and the control strategy but allow for different motions. The first model achieves planar motion, while the second and third work in 3D. It is assumed that no slipping is present at the grasped end of the DLO.

Research Objectives and Contributions The central objective of this thesis is to develop, analyze, and validate an adaptive controller for the robotic manipulation of Deformable Linear Objects that simultaneously achieves shape trajectory tracking and online estimation of the DLO's physical parameters (volumetric density, Young's modulus, and elastic material damping) without requiring prior offline identification. The specific contributions of this work are:

1. **Application of collocated adaptive control to DLO manipulation.** The adaptive control framework of [14] is applied to the GVS-discretized dynamics of a DLO under multi-point contact, providing a model-based controller with formal Lyapunov stability guaranties for shape trajectory tracking.
2. **Online estimation of DLO physical parameters.** Linear density, Young's modulus, and elastic material damping are estimated simultaneously online during closed-loop manipulation. The adaptive update law is derived consistently with the theoretical framework of [14] and does not require offline parameter identification or acceleration measurements.
3. **Analysis of the collocated structure of DLO manipulation.** The collocated coordinate transformation of [16] is applied to the GVS dynamic model of the DLO, and the conditions under which the DLO manipulation system satisfies the collocated property are characterized.
4. **Simulation and Experimental validation.** The proposed approach is validated in simulation on representative DLO manipulation scenarios with initially unknown physical parameters, demonstrating both trajectory tracking performance and convergence of the parameter estimates. The experimental results are validated through single-arm manipulation experiments on different tasks.

Thesis Outline The remainder of this thesis is organized as follows.

Chapter 3 presents the dynamic model of the DLO. Starting from the Cosserat rod equations, the GVS discretization is derived, and the model is expressed in Lagrangian form as an underactuated mechanical system. The collocated coordinate transformation of [18] is then applied, and the physical parameters are identified as base parameters of the resulting collocated dynamics.

Chapter 4 derives the adaptive control law. Building on [14], the collocated sliding variable is defined, the adaptive update law is constructed, and the stability of the closed-loop system is analyzed via a Lyapunov argument. The conditions required for parameter convergence are discussed.

Chapter 5 presents the simulation results. The proposed controller is evaluated on representative DLO manipulation tasks with unknown physical parameters, and comparisons are drawn between the adaptive and non-adaptive cases.

Chapter 6 presents the experimental results on the single arm manipulation scenario, considering different initial conditions and tasks.

Chapter 7 concludes with a summary of the contributions, a discussion of the limitations of the proposed approach, and directions for future research.

3

Dynamic model of the DLO

Three complementary systems are studied in this work, all modeling a Deformable Linear Object (DLO) as a single Cosserat rod within the Geometric Variable Strain (GVS) framework [13]. The first system, depicted in Figure 3.2, couples the DLO to a fixed revolute joint and targets the control of the revolute joint angle. The second, shown in Figure 3.4, is a natural evolution of the previous model; it considers a DLO grasped at one end by a manipulator, allowing full motion in $\mathbb{R}^6$. The third, depicted in Figure 3.5, attaches each end of the DLO to a robotic end-effector and targets full shape control via bimanual manipulation.

Despite their different tasks, all systems share the same GVS modeling pipeline: strain parametrization, recursive kinematics via Magnus expansion, and Lagrangian dynamics. What differs between them is the *base joint model* and the *actuation structure*, which in turn determines whether a collocated change of coordinates is needed. These differences are highlighted as each system is introduced.

3.1 GVS Modeling

A Deformable Linear Object is, in the continuum limit, a Cosserat rod with infinitely many degrees of freedom. To make the model computationally tractable while retaining an accurate description of bending, shear and extension, this thesis adopts the Geometric Variable Strain (GVS) method of Mathew and Feliu-Talegon [13], a reduced-order discretization built directly on the strain field of the rod rather than on the cross-sectional poses used by finite-element approaches. A further advantage of GVS, exploited throughout this chapter, is that it treats rigid and soft elements within the same recursive scheme, so that a revolute joint, a free-floating base, and a soft rod can all be assembled into one kinematic chain.

This section introduces the parts of the GVS pipeline that are common to all three configurations studied in this work: the strain parametrisation, the resulting Jacobian relation, and the generic form of the equations of motion. The three configurations of Sections 3.1.4–3.1.7 differ only in *how these generic quantities are instantiated* – the base joint model and the actuation structure – not in the pipeline itself. The complete, step-by-step derivation of every equation in this section (Magnus expansion, Adjoint mapping, and the rigid/soft distinction) is algebraically involved but not needed to follow the rest of this thesis; it is given in full in Appendix 8.1.1, following the derivation of Mathew and Feliu-Talegon [13].

For the sake of simplicity, we report here some of the basic symbolic conventions that will be adopted in the following:

- $(\hat{\cdot})$ means that the variable belongs to $\mathfrak{se}(3)$
- $(\cdot)$ means that the variable takes real values
- $(\dot{\cdot})$ denotes to the time derivative
- $(\cdot)'$ denotes to the spatial derivative
- $ad_{(\cdot)}$ and $ad_{(\cdot)}^*$ correspond to the adjoint and coadjoint operator of $\mathfrak{se}(3)$
- $Ad_{(\cdot)}$ and $Ad_{(\cdot)}^*$ correspond to the adjoint and coadjoint map of $SE(3)$

3.1.1 Strain parametrisation

Each link – rigid or soft – is described along its curvilinear abscissa X , by a pose curve $g(X) \in SE(3)$, whose spatial derivative defines the *strain twist* $\hat{\xi}(X) \in \mathfrak{se}(3)$:

$$g'(X) = g(X) \hat{\xi}(X) \quad (3.1)$$

$$g(X) = \begin{bmatrix} R & r \\ 0 & 1 \end{bmatrix} \quad \hat{\xi}(X) = \begin{bmatrix} \tilde{k} & p \\ 0 & 1 \end{bmatrix} \quad (3.2)$$

where $X = X_i/L$ is the normalised curvilinear abscissa that varies between 0 and 1, which correspond to the extremities of the link. For a rigid link, the curvilinear abscissa X is not related to a position, but it just takes value 0 for a motion not completed and 1 when the motion has occurred. Furthermore, the pose curve $g(X)$ is composed of a rotation matrix R and a position vector r associated to X , while the strain twist $\hat{\xi}(X)$ consists of $\tilde{k}$ and p , the angular and linear strains, respectively.

For a rigid joint, ξ is constant and plays the role of the joint twist; for a soft link, ξ varies continuously along X and is approximated in finite dimensions by a linear combination of basis functions (Legendre polynomials in this work):

$$\xi(X, q) = \Phi_{\xi}(X) \mathbf{q} + \xi^*(X), \quad X \in [0, 1], \quad (3.3)$$

where $\mathbf{q}$ collects the generalized coordinates, $\Phi_\xi(X)$ is the discretization matrix, and ξ^* is the reference strain of the undeformed configuration. Equation (3.3) is the single most important modeling choice in this thesis: because Φ_ξ does not depend on $\mathbf{q}$, both the velocity and acceleration twists are linear in $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$, and – as shown below – the material parameters ρ , E and η enter the resulting dynamics *linearly*, which is precisely the structural property the adaptive law of Chapter 4.1 relies on and that will be demonstrated in Th.4.1.5.

3.1.2 Recursive kinematics

Given the strain field, the pose g , velocity twist η , and acceleration twist η at any point X are obtained recursively along the rod. They are computed following Eq.8.22–Eq.8.23.

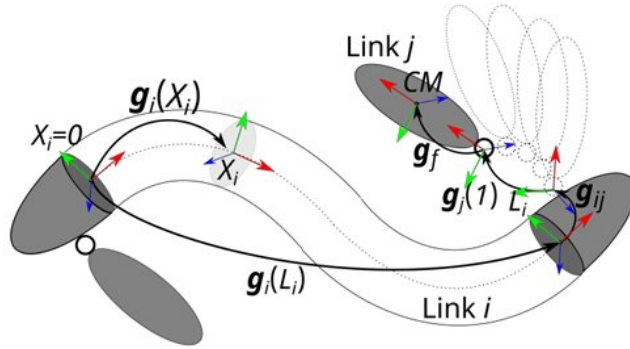


Figure 3.1: Structure of the i -th soft link used in the reconstruction of the system's kinematics. Adapted from Matthews' [13]

For a rigid joint the integration is exact, $g(1) = \exp(\hat{\xi})$; for a soft link, the pose update requires the Magnus expansion of the strain twist $\hat{\Omega}(h)$ over each integration interval $h = (X_{k+1} - X_k)L$,

$$g(X + h) = g(X) \exp(\hat{\Omega}(h)), \quad (3.4)$$

approximated to fourth order by a two-point Zanna quadrature so that only short link segments need to be integrated at a time. Differentiating this recursion with respect to time yields the corresponding update for the velocity and acceleration twists, culminating in the standard robotics-like Jacobian relation

$$\eta(X) = J(X) \dot{\mathbf{q}}, \quad \dot{\eta}(X) = J(X) \ddot{\mathbf{q}} + \dot{J}(X) \dot{\mathbf{q}}, \quad (3.5)$$

with $J(X)$ updated recursively alongside $g(X)$. The derivation of Eq. (3.5) – which requires expressing η in terms of the Adjoint and adjoint operators, distinguishing the rigid and soft cases, and integrating spatially via the Leibniz rule – is carried out in full in Appendix 8.1.2 and is not repeated here, as none of the three configurations in this thesis modify it: only the initial condition of the recursion (the base joint) and the composition of Φ_ξ (the active strain modes) change from one configuration to the next, as detailed in Sections 3.1.4–3.1.7.

3.1.3 Lagrangian dynamics

The generalized dynamics follow from projecting the free dynamics of each link, rigid or soft, respectively defined in Eq.8.24-8.26, onto the generalized coordinates via the Jacobian J , using D'Alembert's principle. For a soft link this projection, denoted by $\mathcal{F}_i$, involves the internal elastic and damping wrench of the material:

$$\mathcal{F}_i(X) = \Sigma(\xi(X) - \xi^*(X)) + \Upsilon \dot{\xi}(X) = \Sigma \Phi_\xi \mathbf{q} + \Upsilon \Phi_\xi \dot{\mathbf{q}}, \quad (3.6)$$

where Σ and Υ are the screw elasticity and screw damping matrices of the cross-section, defined in Eq.8.40-8.34, and it is here that the material parameters ρ (through the inertia density), E (through Σ) and η (through Υ) enter the model, linearly in $\mathbf{q}$ and $\dot{\mathbf{q}}$. Assembling the projected balance over the whole rod by Gaussian quadrature yields the equations of motion in the same form used throughout this thesis:

$$M(\mathbf{q}) \ddot{\mathbf{q}} + [C(\mathbf{q}, \dot{\mathbf{q}}) + D] \dot{\mathbf{q}} + K\mathbf{q} = A(\mathbf{q}) u + Q(\mathbf{q}), \quad (3.7)$$

with

$$M(q) = \int_0^L J^T \bar{M} J dX, \quad C(q, \dot{q}) = \int_0^L J^T (\text{ad}_\eta^* \bar{M} J + \bar{M} \dot{J}) dX, \quad (3.8)$$

$$D = \int_0^L \Phi_\xi^T \Upsilon \Phi_\xi dX, \quad K = \int_0^L \Phi_\xi^T \Sigma \Phi_\xi dX, \quad (3.9)$$

$$Q(q) = \int_0^L J^T \text{Ad}_{g^{-1}}^* [0, 0, 0, 0, 0, -\rho \mathcal{A} g_0]^T dX, \quad (3.10)$$

where $\bar{M}$ is the screw inertia density matrix, defined in Eq.8.32, J and $\dot{J}$ are the jacobian and its derivative, $\mathcal{A}$ is the area associated to the curvilinear abscissa and g_0 is the acceleration of gravity. $A(\mathbf{q})$ is the actuation map, which depends on where and how the control input enters the chain. Equations (3.7)–(3.10) are the generic form instantiated by every configuration in this thesis: what changes between Sections 3.1.4, 3.1.6 and 3.1.7 is only the generalized coordinate vector q , the active strain modes in Φ_ξ , and the actuation map $A(\mathbf{q})$ – never the structure of M , C , D , K or Q itself. The full derivation of Eq. (3.7) from the per-link free dynamics (rigid and soft, including the boundary wrench-balance conditions at the link interfaces) is given in Appendix 8.1.5.

3.1.4 Revolute joint + DLO

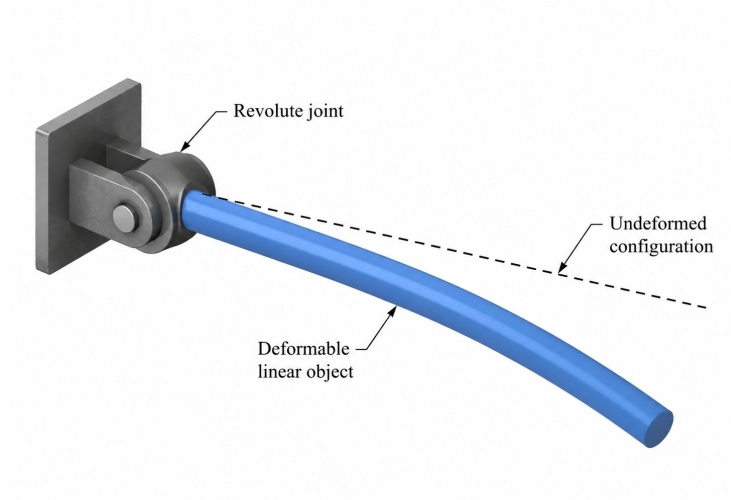


Figure 3.2: Revolute joint + DLO.

Generalised coordinates

The kinematic chain consists of a single revolute joint fixed at the base, rotating about the local y -axis, followed by the soft Cosserat rod. The generalized coordinate vector is:

$$\mathbf{q} = [\theta_a, \mathbf{q}_s^T]^T \in \mathbb{R}^n, \quad n = 1 + n_{\text{soft}}, \quad (3.11)$$

where θ_a is the revolute joint angle and $\mathbf{q}_s \in \mathbb{R}^{n_{\text{soft}}}$ collects the strain coordinates of the soft link.

Strain parametrization

The continuous strain field of the DLO is parametrized as

$$\boldsymbol{\xi}(X, \mathbf{q}_s) = \Phi_{\boldsymbol{\xi}}(X) \mathbf{q}_s + \boldsymbol{\xi}^*, \quad X \in [0, 1], \quad (3.12)$$

where X is the normalised curvilinear abscissa and $\boldsymbol{\xi}^* = [0, 0, 0, 1, 0, 0]^T$ is the reference strain, encoding inextensibility and zero shear in the straight configuration. In this case only bending about the local y -axis is activated, discretised by a Legendre polynomial basis $p(X)$ of order n_{soft} :

$$\Phi_{\boldsymbol{\xi}}(X) = \begin{pmatrix} \mathbf{0} \\ P_0(X) \cdots P_{n_{\text{soft}}}(X) \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \in \mathbb{R}^{6 \times (n_{\text{soft}}+1)}[X], \quad (3.13)$$

giving $n = n_{\text{soft}} + 1$ total DOFs. Since the basis is independent of $\mathbf{q}_s$ and t , $\dot{\xi} = \Phi_\xi \dot{\mathbf{q}}_s$ and $\ddot{\xi} = \Phi_\xi \ddot{\mathbf{q}}_s$.

Forward kinematics

The kinematics are evaluated recursively through a two-level nested quadrature scheme following the strategy presented in [13]. At the inner level, to compute the approximation of the Magnus expansion Ω of the strain, a fourth-order Zanna collocation method is used for each Gauss interval h that divides the DLO. This computation allows to reconstruct the strain ξ relative to the Gauss interval. At the outer level, to merge the Magnus expansions associated with each interval a 5-point Gauss-Legendre quadrature scheme integrates the distributed quantities along the rod to obtain the description of the pose curve $g(X)$ and the Jacobians.

Revolute joint. The joint strain is constant and corresponds to a pure rotation about the local y -axis, whose rotation angle is defined by the actuated coordinate θ_a , so the following strain is obtained:

$$\xi_1 = [0, \theta_a, 0, 0, 0, 0]^T, \quad g_1(1) = \exp(\hat{\xi}_1) \in \text{SE}(3). \quad (3.14)$$

The actuation basis vector is $\mathbf{B}_\Omega = [0, 1, 0, 0, 0, 0]^T$, necessary vector to compute the Magnus expansion of the strain, and the Jacobian column associated with θ_a is:

$$J_1 = \text{Ad}_{g_1^{-1}} T(\xi_1) \mathbf{B}_\Omega \in \mathbb{R}^6. \quad (3.15)$$

where the above equation, which exploits the Adjoint map associated with the inverse pose curve, $\text{Ad}_{g_1^{-1}}$, and the tangent operator $T(\xi_1)$, defined in Eq.8.13, is detailed in Eq.8.18, considering the constant strain scenario.

Soft link. The DLO is subdivided into Gauss intervals $[X_k, X_{k+1}]$ of physical length $h = (X_{k+1} - X_k)L$.

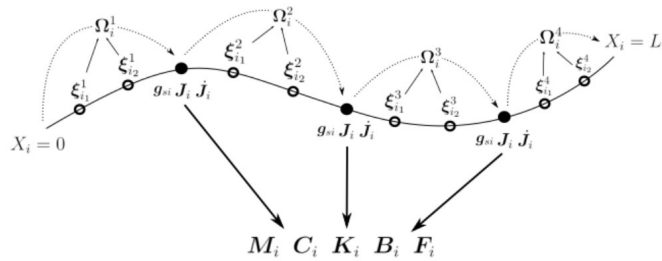


Figure 3.3: Representation of the two-nested quadrature scheme, used in the integration of a soft link.

The two Zanna collocation points within each interval are $z_{1,2} = X_k + (\frac{1}{2} \mp \frac{\sqrt{3}}{6})h$. The fourth-order Magnus expansion of the strain twist over $[X_k, X_k+h]$ is:

$$\Omega(h) = \frac{h}{2}(\xi_{z_1} + \xi_{z_2}) + \frac{\sqrt{3} h^2}{12} \text{ad}_{\xi_{z_1}} \xi_{z_2}, \quad (3.16)$$

and the configuration is updated recursively as $g(X+h) = g(X) \exp(\hat{\mathbf{\Omega}}(h))$, where the difference between $\mathbf{\Omega}$ and $\hat{\mathbf{\Omega}}$ stands on the fact that $\mathbf{\Omega} \in \mathbb{R}^6$ instead $\hat{\mathbf{\Omega}} \in \mathfrak{se}(3)$. The Magnus expansion of the basis matrix $\mathbf{B}_{\mathbf{\Omega}}(h) \in \mathbb{R}^{6 \times n_{\text{soft}}}$ is:

$$\mathbf{B}_{\mathbf{\Omega}}(h) = \frac{h}{2}(\Phi_{z_1} + \Phi_{z_2}) + \frac{\sqrt{3} h^2}{12}(\text{ad}_{\xi_{z_1}} \Phi_{z_2} - \text{ad}_{\xi_{z_2}} \Phi_{z_1}). \quad (3.17)$$

The geometric Jacobian $J(X) \in \mathbb{R}^{6 \times n}$, satisfying $\dot{\eta}(X) = J(X)\dot{\mathbf{q}}$, is then updated as:

$$J(X+h) = \text{Ad}_{\exp(-\hat{\mathbf{\Omega}})} \left[J(X) + \begin{pmatrix} \mathbf{0}_{6 \times 1} & T(\mathbf{\Omega}) \mathbf{B}_{\mathbf{\Omega}} \end{pmatrix} \right], \quad (3.18)$$

where the first column of zeros corresponds to θ_a and $T(\mathbf{\Omega})$ acts on the n_{soft} soft columns.

Equations of motion

Since independent bases are used, the equations of motion take the generic form of Eq. (3.7),

$$M(\mathbf{q}) \ddot{\mathbf{q}} + [C(\mathbf{q}, \dot{\mathbf{q}}) + D] \dot{\mathbf{q}} + K \mathbf{q} = A(\mathbf{q}) u + \mathbf{Q}(\mathbf{q}), \quad (3.19)$$

with M , C , D , K and $\mathbf{Q}$ assembled by Gaussian quadrature exactly as in Eqs. (3.8)–(3.10) of Section 3.1. The matrices $\tilde{\mathbf{M}}$, $\mathbf{\Sigma}$ and $\mathbf{Y}$ appearing in Eqs. (3.8)–(3.10) are, respectively, the screw inertia density, screw elasticity, and screw damping matrices of the uniform circular cross-section scenario presented in Mathew's [13].

Actuation map. The scalar joint torque u acts on the system through the following actuation matrix:

$$A(\mathbf{q}) = J_{\text{base}}^T [0, 1, 0, 0, 0, 0]^T \in \mathbb{R}^n, \quad (3.20)$$

which is the collocated actuation map of the revolute joint. Because u acts directly on θ_a , the system is *naturally in collocated form*: the controlled output and the actuated coordinate coincide, and no change of coordinates is required.

3.1.5 Collocated Coordinates

It makes sense at this point to introduce one of the most relevant tools exploited in this thesis. The *Collocated form* represents a transformed representation of a system. The objective of this transformation is to obtain a description of the model under which the actuator directly performs work on a subset of the configuration variables, namely the actuated variables.

In this paragraph, it will be briefly described the work presented in [16].

Consider a Lagrangian system with generalized coordinates q and equations of motion

$$\frac{d}{dt}(\partial \mathcal{L} / \partial \dot{q})^T - (\partial \mathcal{L} / \partial q)^T = A(q) u,$$

where $A(q)$ is a (possibly configuration-dependent) actuation matrix. Pustina et al. [16] pose

the *Input Decoupling* problem: does a change of coordinates $\theta = h(q)$ exist such that, in the new coordinates, each input u_i performs work on one and only one equation of motion, i.e. the transformed dynamics take the form $[I \ O]^T u$? A system for which such a change of coordinates exists is called *collocated*.

The key result is that such change of coordinates exists if and only if the *passive output* $\dot{y} = A^T(q)\dot{q}$ is integrable, i.e. there exists a scalar function $g(q)$ with $\partial g / \partial q = A^T(q)$, where $A(q)$ corresponds to the actuation map of the system. When $g(q)$ exists, $y = g(q)$ represents the *actuation coordinates*, namely the driving input u only acts on y through the actuation $A(q)$. For a fully-actuated system, the following choice of the actuation coordinates

$$\theta = g(q)$$

already solves the problem ([16, Thm. 1]).

For an *underactuated* system – the case of interest throughout this thesis – the actuated coordinates are again $\theta_a = g(q)$, while the remaining unactuated coordinates θ_u can be completed by *any* function of q whose Jacobian, stacked with that of g , remains invertible; no further constraint is needed for the decoupling to hold ([16, Thm. 2 and Remark]). Namely, what we want to obtain is

$$\theta = \begin{pmatrix} g(q) \\ 0_{n-m} \end{pmatrix} + \begin{pmatrix} \mathbf{0}_{m \times m} & \mathbf{0}_{m \times n-m} \\ \mathbf{0}_{n-m \times m} & I_{n-m} \end{pmatrix} q \quad \dot{\theta} = J\dot{q}$$

This freedom in choosing θ_u is exactly what is exploited in Sections 3.1.6 and 3.1.7: the actuated task-space pose Θ_b (or Θ) plays the role of θ_a , while the untouched strain coordinates $\mathbf{q}_s$ are kept as the complement θ_u , so that the collocated dynamics of Eqs. (3.41) and (3.49) follow directly from [16, Thm. 2] rather than requiring an ad-hoc derivation for each configuration.

3.1.6 Single Arm robot + DLO

The single-robot configuration is the intermediate case between the systems developed in this work: the base of the DLO is grasped and moved by a single robotic end-effector—replacing the fixed revolute joint of Section 3.1.4—while the distal tip remains free, as opposed to the bimanual setup of Section 3.1.7. A prototypical instance of this setup is a robot arm holding one end of a hanging cable, hose, or suture, while the other end is unconstrained.



Figure 3.4: DLO grasped at one end by a manipulator

The central structural consequence of this configuration is that the system is *always underactuated*: the single end-effector provides $m = 6$ control inputs, while the total number of degrees of freedom is $n = 6 + n_{\text{soft}} > 6$ for any nonzero rod discretization. This follows the revolute-joint model, which is naturally in collocated form (Section 3.1.4).

The bimanual model of Section 3.1.7 will be built as a direct extension of the construction derived here, adding a second actuated end-effector at the tip. The single-robot configuration therefore constitutes the canonical case for the collocated adaptive controller of Chapter 4.1: the six base-pose coordinates are directly driven by the end-effector, while the n_{soft} internal strain coordinates are entirely passive and evolve through elastic and inertial coupling alone.

Differences with respect to the revolute-joint model

Three structural differences distinguish this configuration from the revolute-joint model of Section 3.1.4:

1. **Base joint.** The revolute joint is replaced by a **free-floating 6-DOF joint**. Its configuration is encoded by the full Lie-algebra vector $\Gamma_b = \mathbf{q}_b \in \mathbb{R}^6$, with basis matrix $B_{\Omega_b} = I_6$ (all six motion modes are free), giving the generalized coordinate vector:

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_b \\ \mathbf{q}_s \end{bmatrix} \in \mathbb{R}^n, \quad n = 6 + n_{\text{soft}}. \quad (3.21)$$

2. **Active strain modes.** Allowing the end-effector to command the DLO's base anywhere in $\text{SE}(3)$, rather than along a single rotational coordinate, requires the rod itself to be able to bend in both transverse planes. Two strain components are therefore activated: bending about the local y -axis (ξ_2) and bending about the local z -axis (ξ_3), each discretized by a

Legendre polynomial of order n_{ord} :

$$\Phi_\xi(X) = \begin{bmatrix} 0 & \dots & 0 & 0 & \dots & 0 \\ P_0(X) & \dots & P_{n_{\text{ord}}}(X) & 0 & \dots & 0 \\ 0 & \dots & 0 & P_0(X) & \dots & P_{n_{\text{ord}}}(X) \\ 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & \dots & 0 \end{bmatrix} \in \mathbb{R}^{6 \times 2 \cdot (n_{\text{ord}} + 1)}[X], \quad (3.22)$$

In this case we have $n_{\text{soft}} = 2 \cdot (n_{\text{ord}} + 1)$ and $n = 6 + n_{\text{soft}}$ total DOFs. This same strain basis will be reused without modification in the bimanual case of Section 3.1.7.

3. **Actuation map.** Only the base end-effector is present; the tip is free. The end-effector exerts a single 6-DOF spatial wrench $\mathbf{u}_b \in \mathbb{R}^6$ at $X = 0$. As detailed in Section 3.1.6, this yields the actuation matrix $A(\mathbf{q}) = J_b^T \text{Ad}_{g_b}^* \in \mathbb{R}^{n \times 6}$, which has rank $6 < n$ for all $n_{\text{soft}} \geq 1$.

Kinematics

The mathematical derivation of the recursive kinematics of the soft rod follows exactly the procedure of Section 3.1.4 – Equations (3.16)–(3.18) remain unchanged – with two modifications that stem from the free-floating base:

Free-floating base. Instead of a single-axis rotation, the base configuration is:

$$g_b = \exp(\hat{\Gamma}_b) \in \text{SE}(3), \quad \Gamma_b = \mathbf{q}_b, \quad (3.23)$$

and the Jacobian at the base is initialized as:

$$J_b = \text{Ad}_{g_b^{-1}} T(\Gamma_b) I_6 \in \mathbb{R}^{6 \times n}, \quad (3.24)$$

with $T(\Gamma_b) = \int_0^1 \exp(s \text{ad}_{\Gamma_b}) ds$ the tangent operator of the exponential map. The Jacobian update for the soft rod then becomes:

$$J(X + h) = \text{Ad}_{\exp(-\hat{\Omega})} \left[J(X) + \begin{pmatrix} \mathbf{0}_{6 \times 6} & T(\Omega) B_\Omega \end{pmatrix} \right], \quad (3.25)$$

where the leading zero block is now 6×6 (one column per rigid base DOF) instead of 6×1 .

Tip Jacobian. The tip Jacobian $J_t = J(X = 1)$ is the final value of the recursion. In this configuration the tip is free, so J_t is only used to compute the (uncontrolled) tip velocity twist $\eta_t = J_t \dot{\mathbf{q}}$ for monitoring purposes; it does not enter the actuation map. This will no longer be the case in the bimanual configuration of Section 3.1.7, where a second end-effector acts at the tip.

Equations of motion

All system matrices $M(\mathbf{q})$, $C(\mathbf{q}, \dot{\mathbf{q}})$, D , K , and $\mathbf{Q}(\mathbf{q})$ are assembled exactly as in Eqs. (3.8)–(3.10) of Section 3.1, using the kinematics of Section 3.1.6. The only structural difference with respect to the generic form of Eq. (3.7) is in the external generalised force, which now reflects the single-base actuation:

$$M(\mathbf{q}) \ddot{\mathbf{q}} + [C(\mathbf{q}, \dot{\mathbf{q}}) + D] \dot{\mathbf{q}} + K \mathbf{q} = \underbrace{J_b^T \text{Ad}_{g_b}^*}_{\mathbf{A}(\mathbf{q}) \in \mathbb{R}^{n \times 6}} \mathbf{u}_b. \quad (3.26)$$

Because the actuation matrix $\mathbf{A}(\mathbf{q}) \in \mathbb{R}^{n \times 6}$ has exactly $m = 6$ columns and the system has $n = 6 + n_{\text{soft}}$ degrees of freedom, the deficiency $n - m = n_{\text{soft}}$ is strictly positive: the system is *always underactuated*, regardless of the discretization order n_{ord} .

Collocated change of coordinates

As anticipated in Section 3.1.5, the generalised coordinates $\mathbf{q}$ are not collocated with the task-space input $\mathbf{u}_b$: the wrench acts on the rod's base in the world frame, whereas $\mathbf{q}_b$ encodes the base configuration as a Lie-algebra vector. A coordinate transformation is therefore required, following Theorem 2 of [16]: the actuated coordinates θ_a are taken as the (here directly available, rather than integrated) pose of the base, while the unactuated complement θ_u is simply left as $\mathbf{q}_s$, which is always a valid choice since the resulting Jacobian is invertible.

Because only the six floating-base coordinates $\mathbf{q}_b$ are replaced — by the corresponding task-space pose Θ_b — while the n_{soft} soft coordinates $\mathbf{q}_s$ are left unchanged, the resulting Jacobian is square and block-diagonal by construction, which simplifies both the inversion and the control analysis. This is the simplest instance of the collocated transformation used in this thesis; Section 3.1.7 will extend it to the case where *both* endpoints carry actuation, and hence the full coordinate vector $\mathbf{q}$ — not just $\mathbf{q}_b$ — must be remapped.

Task-space coordinates. The collocated coordinates are the ZYX Euler angles and Cartesian position of the base frame:

$$\Theta_b = [\phi_b, \sigma_b, \psi_b, x_b, y_b, z_b]^T \in \mathbb{R}^6, \quad (3.27)$$

obtained from the ZYX decomposition of $g_b = g(0) = \exp(\hat{\Gamma}_b) \in \text{SE}(3)$. Because g_b depends only on $\mathbf{q}_b$ (not on $\mathbf{q}_s$), the n_{soft} strain coordinates are left as non-collocated variables. The complete n -dimensional coordinate set is therefore:

$$\Theta = \begin{bmatrix} \Theta_b \\ \mathbf{q}_s \end{bmatrix} \in \mathbb{R}^n = \mathbb{R}^{6+n_{\text{soft}}}, \quad (3.28)$$

which partitions the state vector into $m = 6$ collocated and n_{soft} non-collocated coordinates.

Task-space Jacobian. Since g_b depends only on $\mathbf{q}_b$, the body-frame base Jacobian $J_b \in \mathbb{R}^{6 \times n}$ is non-zero only in its first six columns:

$$J_b = \begin{bmatrix} \underbrace{J_{b,r}}_{\in \mathbb{R}^{6 \times 6}} & \mathbf{0}_{6 \times n_{\text{soft}}} \end{bmatrix}, \quad (3.29)$$

and its spatial counterpart is:

$$J_b^s = \text{Ad}_{g_b} J_b = \begin{bmatrix} \underbrace{\text{Ad}_{g_b} J_{b,r}}_{S_{6 \times 6}} & \mathbf{0}_{6 \times n_{\text{soft}}} \end{bmatrix} \in \mathbb{R}^{6 \times n}. \quad (3.30)$$

The angular-velocity components are rotated to ZYX Euler-angle rates via the standard mapping

$$P(\sigma, \psi) = \begin{bmatrix} \cos \psi \sec \sigma & \sin \psi \sec \sigma & 0 \\ -\sin \psi & \cos \psi & 0 \\ \cos \psi \tan \sigma & \sin \psi \tan \sigma & 1 \end{bmatrix}, \quad (3.31)$$

which will be reused, by the endpoint, in the bimanual case of Section 3.1.7. Applying it to the base gives:

$$\Psi_b = \text{blkdiag}(P(\sigma_b, \psi_b), I_3) \in \mathbb{R}^{6 \times 6}, \quad (3.32)$$

and the task-space Jacobian associated with Θ_b :

$$J_a^b = \Psi_b J_b^s = \begin{bmatrix} \underbrace{\Psi_b S_{6 \times 6}}_{A_{6 \times 6}} & \mathbf{0}_{6 \times n_{\text{soft}}} \end{bmatrix} \in \mathbb{R}^{6 \times n}, \quad (3.33)$$

so that $\dot{\Theta}_b = J_a^b \dot{\mathbf{q}}$, as described in Section 3.1.5.

The Jacobian of the full transformation (3.28) is:

$$J_a = \begin{bmatrix} A_{6 \times 6} & \mathbf{0}_{6 \times n_{\text{soft}}} \\ \mathbf{0}_{n_{\text{soft}} \times 6} & I_{n_{\text{soft}}} \end{bmatrix} \in \mathbb{R}^{n \times n}, \quad \dot{\Theta} = J_a \dot{\mathbf{q}}, \quad (3.34)$$

which is block-diagonal with a square, invertible structure (away from the kinematic singularities of the base, where Ψ_b degenerates at $\sigma_b = \pm\pi/2$).

Coordinate transformation. The following describes the procedure exploited in Feliu-Talegon's work [8], where the modeling and control of a DLO grasped in different points is developed. Differentiating (3.34) gives $\ddot{\Theta} = \dot{J}_a \dot{\mathbf{q}} + J_a \ddot{\mathbf{q}}$, hence:

$$\ddot{\mathbf{q}} = J_a^{-1}(\ddot{\Theta} - \dot{J}_a \dot{\mathbf{q}}). \quad (3.35)$$

Substituting into (3.26) and left-multiplying by $\mathbf{J}_a^{-T}$ yields the transformed dynamics:

$$M_\Theta \ddot{\Theta} + C_\Theta \dot{\Theta} + D_\Theta \dot{\Theta} + K_\Theta = J_a^{-T} A(\mathbf{q}) \mathbf{u}_b + \mathbf{Q}_\Theta, \quad (3.36)$$

with:

$$M_{\Theta} = J_a^{-T} M J_a^{-1}, \quad C_{\Theta} = J_a^{-T} (C - M J_a^{-1} \dot{J}_a) J_a^{-1}, \quad D_{\Theta} = J_a^{-T} D J_a^{-1}, \quad K_{\Theta} = J_a^{-T} K \mathbf{q}. \quad (3.37)$$

The block-diagonal structure of $\mathbf{J}_a$ in Eq. (3.34) makes its inverse explicit:

$$J_a^{-1} = \begin{bmatrix} A_{6 \times 6}^{-1} & \mathbf{0} \\ \mathbf{0} & I_{n_{\text{soft}}} \end{bmatrix}, \quad J_a^{-T} = \begin{bmatrix} A_{6 \times 6}^{-T} & \mathbf{0} \\ \mathbf{0} & I_{n_{\text{soft}}} \end{bmatrix}. \quad (3.38)$$

Collocated input. The actuation matrix $A(\mathbf{q}) = J_b^T \text{Ad}_{g_b}^* = (J_b^s)^T$ has, by Eq. (3.30), the explicit form:

$$A(\mathbf{q}) = (J_b^s)^T = \begin{bmatrix} S_{6 \times 6}^T \\ \mathbf{0}_{n_{\text{soft}} \times 6} \end{bmatrix} \in \mathbb{R}^{n \times 6}. \quad (3.39)$$

Left-multiplying by J_a^{-T} (Eq. (3.38)) and using $A_{6 \times 6} = \Psi_b S_{6 \times 6}$ leads to:

$$J_a^{-T} A(\mathbf{q}) = \begin{bmatrix} A_{6 \times 6}^{-T} & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix} \begin{bmatrix} S_{6 \times 6}^T \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} A_{6 \times 6}^{-T} S_{6 \times 6}^T \\ \mathbf{0}_{n_{\text{soft}} \times 6} \end{bmatrix} = \begin{bmatrix} \Psi_b^{-T} \\ \mathbf{0}_{n_{\text{soft}} \times 6} \end{bmatrix}, \quad (3.40)$$

where the last equality follows from $A_{6 \times 6}^{-T} S_{6 \times 6}^T = (S_{6 \times 6} A_{6 \times 6}^{-1})^T = (S_{6 \times 6} (\Psi_b S_{6 \times 6})^{-1})^T = \Psi_b^{-T}$.

Defining the transformed input $\tilde{\mathbf{u}}_b = \Psi_b^T \tilde{\mathbf{u}}_b$ and substituting into (3.36) yields the **collocated dynamics**:

$$M_{\Theta} \ddot{\Theta} + (C_{\Theta} + D_{\Theta}) \dot{\Theta} + K_{\Theta} = \underbrace{\begin{bmatrix} I_6 \\ \mathbf{0}_{n_{\text{soft}}} \end{bmatrix}}_{\text{collocated structure}} \tilde{\mathbf{u}}_b + \mathbf{Q}_{\Theta}. \quad (3.41)$$

Remark 3.1.1 (Collocated structure and underactuation). The input matrix $[I_6; \mathbf{0}_{n_{\text{soft}}}]^T$ in Eq. (3.41) has a direct physical interpretation. The six base-pose coordinates Θ_b are *collocated*: each component of $\tilde{\mathbf{u}}_b$ drives exactly one component of Θ_b , with no coupling to $\mathbf{q}_s$ at the input level. The n_{soft} strain coordinates $\mathbf{q}_s$ are *non-collocated*: they receive no direct input and evolve entirely through the inertial and elastic coupling encoded in the off-diagonal blocks of M_{Θ} , C_{Θ} , and D_{Θ} .

3.1.7 Bimanual robot + DLO

In this configuration each end of the DLO is held by a robotic end-effector. The control objective is to regulate the full 6-DOF pose of both the base and the tip of the DLO. The low-level joint control of each arm is handled separately; the model developed here operates at the task-space level, treating the end-effector wrenches as the control inputs. As anticipated in Section 3.1.6, this configuration is obtained by adding a second actuated end-effector at the tip of the single-robot model, and the presentation below focuses on this one addition rather than re-deriving quantities already established.



Figure 3.5: DLO grasped at both ends by a bimanual robot

Differences with respect to the single-robot model

Only the actuation structure changes with respect to Section 3.1.6; the base joint and the active strain modes are unchanged:

1. **Base joint.** As in Section 3.1.6, the DLO's base is a free-floating 6-DOF joint, giving the same generalized coordinate vector structure as in Eq. (3.21), $\mathbf{q} = [\mathbf{q}_b^T, \mathbf{q}_s^T]^T \in \mathbb{R}^n$ with $n = 6 + n_{\text{soft}}$, and n_{soft} is computed as in the previous case.
2. **Active strain modes.** Three-dimensional shape control was already required in the single-robot case to allow the base to reach an arbitrary pose in $\text{SE}(3)$; the same two active strain components (bending about the local y - and z -axes) and the same discretization matrix $\Phi_\xi(X)$ from Eq. (3.22) are reused here without modification.
3. **Actuation structure.** Unlike the single-robot case, where only the base carries a 6-DOF wrench, here *two* end-effectors apply 6-DOF wrenches at both the base and the tip, yielding $m = 12$ control inputs. Since $m \leq n$, the system can now be either fully actuated or underactuated in the generalized coordinates, depending on the discretization order—in contrast with the single-robot case, which is always underactuated regardless of n_{ord} . As in Section 3.1.6, a change of coordinates to a task-space representation is required (Section 3.1.7), now extended to both endpoints.

Kinematics

The recursive kinematics – the free-floating base initialization of Eq. (3.23), the tangent operator, and the Jacobian update of Eq. (3.25) – are identical to those already derived for the single-robot configuration in Section 3.1.6 and are not repeated here. The only difference is that the tip Jacobian $J_t = J(X = 1)$, used in Section 3.1.6 only to compute the (uncontrolled) tip velocity twist, now also enters the actuation map, since a second end-effector acts at $X = 1$.

Equations of motion

The equations of motion extend Eq. (3.26) of Section 3.1.6 with an additional wrench term at the tip; the generalized matrices are assembled identically (Eqs. (3.8)–(3.10)):

$$M(\mathbf{q}) \ddot{\mathbf{q}} + [C(\mathbf{q}, \dot{\mathbf{q}}) + D] \dot{\mathbf{q}} + K \mathbf{q} = \mathbf{Q}(\mathbf{q}) + J_b^T \text{Ad}_{g_b}^* \mathbf{u}_b + J_t^T \text{Ad}_{g_t}^* \mathbf{u}_t, \quad (3.42)$$

where $\mathbf{u}_b, \mathbf{u}_t \in \mathbb{R}^6$ are the spatial wrenches exerted by the two end-effectors at the base and tip of the DLO, respectively.

Collocated change of coordinates

The construction follows the same recipe as Section 3.1.6, extended to both endpoints. Unlike the single-robot case, where only the six rigid base coordinates $\mathbf{q}_r$ were remapped while $\mathbf{q}_s$ was left untouched (since the tip carried no actuation), here *both* endpoints are actuated, so the *full* generalized coordinate vector $\mathbf{q}$ must be mapped onto the twelve-dimensional pose of the base and tip together; no coordinate is left as an untouched complement this time.

Task-space coordinates. The collocated coordinates are the ZYX Euler angles and Cartesian position of the base and tip:

$$\Theta = [\phi_b, \sigma_b, \psi_b, x_b, y_b, z_b, \phi_t, \sigma_t, \psi_t, x_t, y_t, z_t]^T \in \mathbb{R}^{12}, \quad (3.43)$$

extracted from the ZYX decomposition of $g_b = g(0)$ and $g_t = g(1)$.

Task-space Jacobian. The body-frame Jacobians at the base and tip are rotated to the world frame:

$$J_b^s = \text{Ad}_{g_b} J_b, \quad J_t^s = \text{Ad}_{g_t} J_t, \quad (3.44)$$

and stacked into the spatial Jacobian $J_s = \begin{bmatrix} J_b^{s^T} \\ J_t^{s^T} \end{bmatrix} \in \mathbb{R}^{12 \times n}$. Applying the ZYX angular-velocity mapping $P(\sigma, \phi)$ of Eq. (3.31) at each endpoint yields $\Psi_{b/t} = \text{blkdiag}(P_{b/t}, I_3)$ and the full 12×12 matrix $\Psi = \text{blkdiag}(\Psi_b, \Psi_t)$. This approach again follows the work of Feliu-Talegon [8], more specifically what has been described in the cited work in Section 3 and Section 5. The task-space Jacobian is then:

$$J_a = \Psi J_s \in \mathbb{R}^{12 \times n}, \quad \dot{\Theta} = J_a \dot{\mathbf{q}}. \quad (3.45)$$

Coordinate transformation. Differentiating (3.45) gives $\ddot{\Theta} = \dot{J}_a \dot{\mathbf{q}} + J_a \ddot{\mathbf{q}}$, so that $\ddot{\mathbf{q}} = J_a^{-1}(\ddot{\Theta} - \dot{J}_a \dot{\mathbf{q}})$. Substituting into (3.42) and left-multiplying by J_a^{-T} yields the transformed dynamics:

$$M_\Theta \ddot{\Theta} + C_\Theta \dot{\Theta} + D_\Theta \dot{\Theta} + K_\Theta = J_a^{-T} A(\mathbf{q}) \ddot{\mathbf{u}} + \mathbf{Q}_\Theta, \quad (3.46)$$

where:

$$M_{\Theta} = J_a^{-T} M J_a^{-1}, \quad C_{\Theta} = J_a^{-T} (C - M J_a^{-1} \dot{J}_a) J_a^{-1}, \quad D_{\Theta} = J_a^{-T} D J_a^{-1}, \quad K_{\Theta} = J_a^{-T} K \mathbf{q}. \quad (3.47)$$

Collocated input. The generalized actuation matrix in the bimanual case is

$$A(\mathbf{q}) = [J_b^T \text{Ad}_{g_b^{-1}}^*, J_t^T \text{Ad}_{g_t^{-1}}^*].$$

Since this equals J_s^T , and $J_a = \Psi J_s$, it follows:

$$J_a^{-T} A(\mathbf{q}) = (\Psi J_s)^{-T} J_s^T = \Psi^{-T}. \quad (3.48)$$

Defining the transformed input $\tilde{\mathbf{u}} = \Psi^T \tilde{\mathbf{u}}$ therefore yields the collocated dynamics:

$$M_{\Theta} \ddot{\Theta} + (C_{\Theta} + D_{\Theta}) \dot{\Theta} + K_{\Theta} = \mathbf{I} \tilde{\mathbf{u}} + \mathbf{Q}_{\Theta}, \quad (3.49)$$

where the fact that the input matrix is the identity $\mathbf{I}$ confirms that each component of $\tilde{\mathbf{u}}$ directly drives the corresponding task-space coordinate in Θ .

Remark 3.1.2 (Comparison with the single-robot collocated form). In the single-robot case (Eq. (3.41)), the input matrix is $[I_6; \mathbf{0}_{n_{\text{soft}}}]^T$ regardless of n_{soft} , confirming that the system is always underactuated and that $\mathbf{q}_s$ is always a non-collocated subsystem. Here, by contrast, the input matrix is I_n , achieved by mapping *all* $n = 6 + n_{\text{soft}}$ generalized coordinates to the 12-dimensional task-space pose: when $n_{\text{soft}} = 6$ (i.e. $n = 12$), the transformation is square and the system may be fully actuated. The collocated transformation of Eq. (3.45) is therefore generally dense and requires $n_{\text{soft}} = 6$ for exact invertibility, unlike the square, block-diagonal Jacobian of the single-robot case (Eq. (3.34)), which was invertible by construction for any n_{soft} and required no pseudoinverse.

4

Adaptive Control

4.1 Structural Properties and Control Law

When a model-based control strategy is employed, its validity can be established mathematically. However, this comes at the cost of requiring accurate knowledge of the object's physical characteristics, which may be difficult to obtain in our case and may require a separate identification procedure. To address this, we adopt an adaptive control strategy that simultaneously regulates the system and estimates the unknown physical parameters online. Specifically, we exploit the collocated adaptive control of "Underactuated Mechanical Systems", described in [14], which requires a set of structural properties of the dynamic model to hold. In this section, we firstly prove that these properties are satisfied for our robot-DLO system. Then we will describe the various elements present in the control action itself.

4.1.1 Properties proof

The adaptive control framework requires the following five properties of the dynamic model:

- (1) boundedness and positive definiteness of the inertia matrix,
- (2) skew-symmetry of $\dot{M} - 2C$,
- (3) linear boundedness of the Coriolis matrix in velocity,
- (4) boundedness of gravity,
- (5) linear parametrizability in the base parameters π .

We now prove each in turn for the collocated coordinates θ . The following proofs refer to the collocated coordinates, but are related to the same properties for the generalized case that are reported in Appendix B (8.2).

Theorem 4.1.1. *The inertia matrix M_θ is bounded, symmetric and positive definite for any θ , i.e.,*

$$\lambda_3 I_n \leq M_\theta \leq \lambda_4 I_n \quad \forall \theta$$

with λ_3, λ_4 being two strictly positive constants.

Proof. The proof exploits the results from Th.8.2.1:

$$\begin{aligned} M_\theta^T &= \left[J_h^{-T} M_q J_h^{-1} \right]^T = J_h^{-T} M_q^T J_h^{-1} = J_h^{-T} M_q J_h^{-1} = M_\theta \\ y &= J_h^{-1} x \quad \forall x \\ x^T M_\theta x &= x^T J_h^{-T} M_q J_h^{-1} x = y^T M_q y > 0 \end{aligned}$$

This result proves that M_θ is also symmetric positive definite. We need to show the boundedness of the inertia matrix. As in the counterpart of the generalized coordinates, λ_3 exists due to positive definiteness. Instead, for the upper bound

$$\begin{aligned} \|M_\theta\| &\leq \|J_h^{-T}\| \|M_q\| \|J_h^{-1}\| \leq \lambda_2 \|J_h^{-1}\|^2 \leq \lambda_4 \\ \|J_h^{-1}\| &\leq J_M^{-1}, \end{aligned}$$

where J_h

$$\|J_h^{-1}\| \leq \|A_a^{-T}\| + \|A_a^{-T}\| \|A_u^T\| + 1 \leq \|A_a^{-1}\| + \|A_a^{-1}\| \|A_u\| + 1.$$

Hence, the key step is to exploit the boundedness of the actuation matrix, which holds in a finite subspace of the configuration space of the robot. $\square$

Theorem 4.1.2. *The matrix $\dot{M}_\theta - 2C_\theta$ is skew-symmetric, i.e.,*

$$x^T (\dot{M}_\theta - 2C_\theta) x = 0, \quad \forall x \in \mathbb{R}^n.$$

Proof. Proceeding in the same way as in Th.8.2.2

$$\begin{aligned} \dot{M}_\theta &= \dot{J}_h^{-T} M_q J_h^{-1} + J_h^{-T} \dot{M}_q J_h^{-1} + J_h^{-T} M_q \dot{J}_h^{-1} \\ I = J_h J_h^{-1} \quad \dot{I} = 0 &\longrightarrow \frac{dJ_h^{-1}}{dt} = -J_h^{-1} \dot{J}_h J_h^{-1} \\ x &= \dot{\theta} \end{aligned}$$

Then the property can be demonstrated as follows:

$$\begin{aligned} \dot{\theta}^T (\dot{M}_\theta - 2C_\theta) \dot{\theta} &= \dot{q}^T J_h^T (\dot{M}_\theta - 2C_\theta) J_h \dot{q} = \\ &= \dot{q}^T \left(-\dot{J}_h^T J_h^{-T} M_q + \dot{M}_q - M_q J_h^{-1} \dot{J}_h - 2 \left(C_q - M_q J_h^{-1} \dot{J}_h \right) \right) \dot{q} = \\ &= -\dot{q}^T \underbrace{\dot{J}_h^T J_h^{-T} M_q}_{0} \dot{q} + \dot{q}^T (\dot{M}_q - 2C_q) \dot{q} + \dot{q}^T M_q J_h^{-1} \dot{J}_h \dot{q} = 0 \end{aligned}$$

where the last step is justified by Th.8.2.2. and the symmetry of M_q . □

Theorem 4.1.3. *The Coriolis matrix $C(\theta, \dot{\theta})$ satisfies*

$$\|C(\theta, \dot{\theta})\| \leq \alpha_1 \|\dot{\theta}\| \quad \forall \theta$$

for some bounded positive constant α_1 .

Proof. Proceeding with the definition of the Coriolis matrix in the new set of coordinates we obtain

$$C_\theta = J_h^{-T} \left(C(q) - M(q) J_h^{-1} \dot{J}_h \right) J_h^{-1}.$$

Exploiting the triangle inequality, we get

$$\begin{aligned} \|C_\theta\| &\leq \underbrace{\|J_h^{-T} C(q) J_h^{-1}\|}_{V_1} + \underbrace{\|J_h^{-T} M(q) J_h^{-1} \dot{J}_h J_h^{-1}\|}_{V_2} \\ \|V_1\| &\leq \|J_h^{-1}\|^2 \|C_q\| \leq \|J_h^{-1}\|^2 \alpha_0 \|\dot{q}\| \leq \alpha_0 \|J_h^{-1}\|^3 \|\dot{\theta}\| \leq \alpha_0 \|\dot{\theta}\| \\ \|V_2\| &\leq \|J_h^{-1}\|^3 \|\dot{J}_h\| \|M_q\| \leq \lambda_2 \|J_h^{-1}\|^3 \|\dot{J}_h\| \end{aligned}$$

where V_1 exploits the result in generalized coordinates shown in Th.8.2.3. To show the dependence of V_2 on $\|\dot{\theta}\|$, we consider how the derivative of the Jacobian is computed:

$$\begin{aligned} \frac{dJ_h}{dt} &= \frac{d}{dt} \begin{bmatrix} A_a^T & A_u^T \\ 0 & I_{n-m} \end{bmatrix} = \begin{bmatrix} \dot{A}_a^T & \dot{A}_u^T \\ 0 & 0 \end{bmatrix} \\ \|\dot{J}_h\| &\leq \|\dot{A}_q\| = \left\| \frac{\partial A_q}{\partial q} \dot{q} \right\| = \left\| \frac{\partial A_q}{\partial q} \right\| \|\dot{q}\| = \left\| \frac{\partial A_q}{\partial q} \right\| \|J_h^{-1}\| \|\dot{\theta}\| \end{aligned}$$

The key step again is to exploit the boundedness of the Jacobian as in Th.4.1.1, with the additional bound on $\frac{\partial A_q}{\partial q}$, that holds due to the definition of the model and the related finiteness of the actuation matrix. □

Theorem 4.1.4. *The gravity vector $\mathbf{g}(\theta)$ satisfies*

$$\|\mathbf{g}(\theta)\| \leq \gamma_\theta$$

for some bounded positive constant γ_θ .

Proof. Using the principle of virtual work, we can show the effect of changing the coordinates. Indeed, it holds that the work done considering the collocated coordinates or the generalized coordinates must stay the same

$$\begin{aligned} \delta W_g &= \mathbf{g}_q^T \delta q = \mathbf{g}_\theta^T \delta \theta & \delta \theta &= J_h \delta q \\ \mathbf{g}_q^T \delta q &= (\mathbf{g}_\theta J_h)^T \delta q & \mathbf{g}_q &= J_h^T \mathbf{g}_\theta^T \end{aligned}$$

Finally, using Th.8.2.4

$$\|\mathbf{g}_\theta\| \leq \|J_h^{-T}\| \|\mathbf{g}_q\| \leq \gamma_0 \|J_h^{-1}\|$$

which requires the usual bound on the geometric Jacobian. $\square$

Theorem 4.1.5. *The Lagrangian model can be expressed linearly with respect to the system's base parameters π , i.e., there exists a regressor matrix $Y(\cdot)$ such that*

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + \mathbf{g}(\theta) + K_\theta + D_\theta\dot{\theta} = Y(\theta, \dot{\theta}, \ddot{\theta})\pi$$

Proof. To prove the theorem, we can resort to the linearity and distributive properties of the matrix product. We showcase the screw matrix exploited in the construction of the model

$$\bar{M} = \rho \begin{pmatrix} J_x & 0 & 0 & 0 & 0 & 0 \\ 0 & J_y & 0 & 0 & 0 & 0 \\ 0 & 0 & J_z & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathcal{A} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathcal{A} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathcal{A} \end{pmatrix} = \rho \cdot \mathcal{I} \quad (4.1)$$

$$\Upsilon = \eta \begin{pmatrix} J_x & 0 & 0 & 0 & 0 & 0 \\ 0 & 3J_y & 0 & 0 & 0 & 0 \\ 0 & 0 & 3J_z & 0 & 0 & 0 \\ 0 & 0 & 0 & 3\mathcal{A} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathcal{A} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathcal{A} \end{pmatrix} = \eta \cdot \mathcal{U} \quad (4.2)$$

$$\Sigma = \begin{pmatrix} GJ_x & 0 & 0 & 0 & 0 & 0 \\ 0 & EJ_y & 0 & 0 & 0 & 0 \\ 0 & 0 & EJ_z & 0 & 0 & 0 \\ 0 & 0 & 0 & E\mathcal{A} & 0 & 0 \\ 0 & 0 & 0 & 0 & G\mathcal{A} & 0 \\ 0 & 0 & 0 & 0 & 0 & G\mathcal{A} \end{pmatrix} = E \cdot \mathcal{E} \quad (4.3)$$

$$\begin{aligned} M_\theta \ddot{\theta} &= J_h^{-T} M(h^{-1}(\theta)) J_h^{-1} \ddot{\theta} = J_h^{-T} \left(\rho \int_0^L J(h^{-1}(\theta), X)^T \cdot \mathcal{I} \cdot J(h^{-1}(\theta), X) dX \right) J_h^{-1} \ddot{\theta} = \\ &= \rho \left(\underbrace{J_h^{-T} \int_0^L J(q, X)^T \cdot \mathcal{I} \cdot J(q, X) dX J_h^{-1}}_{M_\theta \in \mathbb{R}^{n \times n}} \ddot{\theta} \right) = \rho M_\theta \ddot{\theta} = \\ &= Y_{M_\theta} \rho \end{aligned}$$

In the same way, we can obtain the Coriolis matrix and Eq.8.33

$$\begin{aligned}
C_\theta \dot{\theta} &= J_h^{-T} \left(C(h^{-1}(\theta)) - M(h^{-1}(\theta)) J_h^{-1} \dot{J}_h \right) J_h^{-1} \dot{\theta} = \\
&= J_h^{-T} \left(C(h^{-1}(\theta)) \right) J_h^{-1} \dot{\theta} - J_h^{-T} \left(M(h^{-1}(\theta)) J_h^{-1} \dot{J}_h \right) J_h^{-1} \dot{\theta} = \\
&= J_h^{-T} (\rho C_q) J_h^{-1} \dot{\theta} - J_h^{-T} \left((\rho M_q) J_h^{-1} \dot{J}_h \right) J_h^{-1} \dot{\theta} = \\
&= \rho \left(J_h^{-T} C_q J_h^{-1} \right) \dot{\theta} - \rho \left(J_h^{-T} M_q J_h^{-1} \dot{J}_h J_h^{-1} \right) \dot{\theta} = \\
&= \rho \left(J_h^{-T} C_q J_h^{-1} - J_h^{-T} M_q J_h^{-1} \dot{J}_h J_h^{-1} \right) \dot{\theta} = \\
&= \rho C_\theta \dot{\theta} = Y_{C_\theta} \rho
\end{aligned}$$

And the effect of the gravitational force, considering Eq.8.46 results in

$$\mathbf{g}_\theta = J_h^{-T} g(h^{-1}(\theta)) = \rho Y_{g_\theta}$$

Lastly, we remain to compute the stiffness and damping matrices. But exploiting the result shown in Appendix B, concerning the close form solution of these two matrices, namely Eq.8.43 and Eq.8.38, we get the following results:

$$\begin{aligned}
J_h^{-1} &= \begin{pmatrix} J_1 \\ J_2 \\ \vdots \\ J_6 \end{pmatrix} \quad J_i \in \mathbb{R}^{n_i \times n}, \quad h(\theta)^{-1} = \begin{pmatrix} h(\theta)_1 \\ h(\theta)_2 \\ \vdots \\ h(\theta)_6 \end{pmatrix} \quad h(\theta)_i \in \mathbb{R}^{n_i \times 1} \\
K_\theta &= J_h^{-T} K h(\theta)^{-1} = \\
&= \begin{pmatrix} J_1^T & J_2^T & \dots & J_6^T \end{pmatrix} E \begin{pmatrix} \mathcal{H}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathcal{H}_2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{H}_3 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{H}_4 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{H}_5 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{H}_6 \end{pmatrix} h(\theta)^{-1} = \\
&= E \begin{pmatrix} J_1^T \mathcal{H}_1 & J_2^T \mathcal{H}_2 & \dots & J_6^T \mathcal{H}_6 \end{pmatrix} h(\theta)^{-1} = \\
&= Y_K(\theta) E
\end{aligned}$$

where $J_i^T \mathcal{H}_i \in \mathbb{R}^{n \times n_i}$, so that the dimensions of the matrix are respected.

$$\begin{aligned}
D_\theta \dot{\theta} &= J_h^{-T} D J_h^{-1} \dot{\theta} = \\
&= \begin{pmatrix} J_1^T & J_2^T & \dots & J_6^T \end{pmatrix} \eta \begin{pmatrix} \mathcal{H}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathcal{H}_2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{H}_3 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{H}_4 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{H}_5 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{H}_6 \end{pmatrix} \begin{pmatrix} J_1 \\ J_2 \\ \vdots \\ J_6 \end{pmatrix} \dot{\theta} = \\
&= Y_D(\theta, \dot{\theta}) \eta
\end{aligned}$$

Clearly, the closed form of these last matrices will depend explicitly on the non-linear map $h^{-1}(\theta)$.

Finally, combining all the steps we obtain

$$Y(\theta, \dot{\theta}, \ddot{\theta}) \pi = \begin{bmatrix} Y_{M,C,g} & Y_K & Y_D \end{bmatrix} \begin{pmatrix} \rho \\ E \\ \eta \end{pmatrix}.$$

□

4.1.2 Problem statement

Exploiting the linearization of the model, our aim is to obtain the physical information about the system's (ρ, E, η) . These characteristics are relevant for the object that we are using, and the ability to estimate the value of the parameter is an interesting achievement. In fact, the only other way to obtain this information would be by running specific experiments that are deeply affected by experimental errors. In the following, we will present the Adaptive Control strategy showcased in [14]. Where we introduce an auxiliary control input ξ that can be exploited in both the fully actuated and under-actuated cases for defining the sliding variable $s = \dot{q} - \xi$.

Let us consider the model described by the formulation with regressor and base parameter:

$$Y(q, \dot{q}, \ddot{q}) \pi = \tau \quad (4.4)$$

What follows now are the statement of the Lemma 1 and Lemma 2 presented and proven in the Appendix of Pucci [14].

Lemma 4.1.6. *Assume that Properties 4.1.1-4.1.5 are satisfied and that the reference signal $r(t)$ is bounded in its first and second-derivatives. Apply the following control law to the system (4.4):*

$$\tau = Y(q, \dot{q}, \xi, \dot{\xi}) \hat{\pi} - K s \quad (4.5)$$

where we have:

$$\dot{\xi} = \ddot{r} - \Lambda_1 \dot{e} - \Lambda_2 e \quad (4.6)$$

$$\dot{\tilde{\pi}} = -\Gamma Y(q, \dot{q}, \xi, \dot{\xi})s \quad (4.7)$$

with

$$K, \Lambda_1, \Lambda_2 \in \mathbb{R}^{n \times n}$$

diagonal and positive-definite matrices, and

$$\Gamma \in \mathbb{R}^{p \times p}$$

a positive-definite matrix.

Then, the following results hold:

1. The equilibrium point

$$(e, \tilde{\xi}, s, \tilde{\pi}) = (0_n, 0_n, 0_n, 0_p)$$

of the closed-loop dynamics

$$\begin{cases} \dot{e} = \dot{q} - \dot{r} \\ \dot{\tilde{\xi}} = \dot{\xi} - \ddot{r} \\ \dot{s} = \ddot{q} - \dot{\xi} \\ \dot{\tilde{\pi}} = \dot{\pi} - \dot{\tilde{\pi}} \end{cases}$$

is stable.

2. For any initial condition

$$(e, \tilde{\xi}, s, \tilde{\pi})(0),$$

the trajectories of the closed-loop dynamics are bounded, and the tracking error $e(t) = q(t) - r(t)$ converges to zero.

The stability proof in [14], exploits a Lyapunov candidate function

$$V = \frac{1}{2} \left(s^T M s + \tilde{\pi}^T \Gamma^{-1} \tilde{\pi} + 2e^T K \Lambda_1 e + 2(\xi + \tilde{\Lambda}_1 e)^T \Lambda_1 K \Lambda_2^{-1} (\xi + \tilde{\Lambda}_1 e) \right).$$

Then we can also present the result for the underactuated case that will be exploited in our case.

Lemma 4.1.7. Assume that Properties 4.1.1-4.1.5 hold and that the reference signal is bounded in its first and second-derivatives. Partition the vector ξ as follows:

$$\xi := \begin{bmatrix} \xi_u \\ \xi_a \end{bmatrix} \quad (4.8)$$

where $\xi_u \in \mathbb{R}^k$ and $\xi_a \in \mathbb{R}^m$ are the unactuated and actuated component of the auxiliary input, respectively.

Apply the following control laws to system $Y(q, \dot{q}, \ddot{q})\pi = \begin{bmatrix} \tau \\ 0 \end{bmatrix}$:

$$\bar{\tau} = Y_c(q, \dot{q}, \xi, \dot{\xi})\hat{\pi} - Ks_c \quad (4.9)$$

$$\dot{\xi} = \begin{bmatrix} \dot{\xi}_u \\ \dot{\xi}_a \end{bmatrix} = \begin{pmatrix} M_n^{-1} \left(K_n s_n - Y_n \left(q, \dot{q}, \xi, \begin{bmatrix} 0_k \\ \dot{\xi}_a \end{bmatrix} \right) \hat{\pi} \right) \\ \ddot{r} - \Lambda_1 \dot{e} - \Lambda_2 e \end{pmatrix} \quad (4.10)$$

$$\dot{\hat{\pi}} = -\Gamma Y(q, \dot{q}, \xi, \dot{\xi})s \quad (4.11)$$

with

$$K, \Lambda_1, \Lambda_2 \in \mathbb{R}^{m \times m} \text{ and } K_n \in \mathbb{R}^{k \times k}$$

diagonal, positive-definite matrices, and the matrix M_n defined as the k th-order leading principal minor of the mass matrix M evaluated with estimated base parameters, i.e.,

$$M_n := S_k M(q, \hat{\pi}) S_k^T \quad (4.12)$$

where the selection matrix S_k is given by

$$S_k := \begin{bmatrix} I_k & 0_{k \times (n-k)} \end{bmatrix}. \quad (4.13)$$

Then, the following results hold:

1. The equilibrium point

$$(e, \tilde{\xi}_a, s, \tilde{\pi}) = (0_m, 0_m, 0_n, 0_p)$$

of the closed-loop dynamics

$$\begin{cases} \dot{e} = \dot{q}_a - \dot{r} \\ \dot{\tilde{\xi}} = \dot{\xi}_a - \ddot{r} \\ \dot{s} = \ddot{q} - \dot{\xi} \\ \dot{\tilde{\pi}} = \dot{\pi} - \dot{\hat{\pi}} \end{cases}$$

is stable.

2. Assume that the noncollocated velocities remain bounded, i.e., there exists $\delta > 0$ such that

$$|\dot{q}_n| < \delta \quad \forall t.$$

There exists a neighborhood $\mathcal{I}$ of the origin

$$(0_m, 0_m, 0_n, 0_p)$$

such that if the initial condition

$$(e, \tilde{\xi}_a, s, \tilde{\pi})(0)$$

belongs to $\mathcal{I}$, then the tracking error $e(t)$ converges to zero.

To obtain global asymptotic stability in the underactuated case, an additional element must be considered in the dynamics of the parameters. Indeed, there could be issues related to the zero crossing value of π , which may affect the invertibility of the inertia matrix. Then, to ensure global asymptotic stability we exploit the results from Lemma 2 of [14]:

$$\dot{\hat{\pi}} = -\Gamma (Y(q, \dot{q}, \xi, \dot{\xi})s - \eta\delta)$$

where η and δ are obtained directly from Pucci's work in Eqns. [19a] and [19b] in [14].

4.1.3 Parameter Identification

A central motivation for adopting an adaptive control framework, rather than a fixed-gain strategy, is the ability to retrieve physically meaningful information about the DLO directly from closed-loop operation. Specifically, we select the base parameter vector as

$$\pi = \begin{pmatrix} \rho \\ E \\ \eta \end{pmatrix} \quad (4.14)$$

where ρ denotes the volumetric mass density of the object, E its Young's modulus (characterizing axial and bending stiffness), and η its material damping coefficient. These three quantities fully characterize the mechanical behavior of the DLO within the Cosserat rod framework adopted in this work, and are the parameters that most directly influence the object's dynamic response during manipulation. In the work developed here, it is assumed that the cable material is isotropic, namely that the physical properties do not change by considering different sections of the DLO. However, an even larger effort could be made by considering materials that do not respect this property and exploiting the internal linearity associated with the GVS model and its integral computation.

Role of Each Parameter in the Dynamic Model

To understand the identifiability of each parameter, it is instructive to examine how π enters the terms of the regressor $Y(\theta, \dot{\theta}, \xi, \dot{\xi})$. The inertia matrix M_θ and Coriolis matrix C_θ depend linearly on ρ , as mass distribution governs both kinetic energy and centrifugal effects. The stiffness matrix K_θ depends linearly on E , since the elastic restoring forces are proportional to the Young's modulus through the object's second moment of area. Finally, the damping matrix D_θ depends linearly on η . The shape of the regressor $Y(\theta, \dot{\theta}, \xi, \dot{\xi})$ implies that the three

parameters influence different physical phenomena and appear in distinct terms of the regressor:

$$Y(\theta, \dot{\theta}, \xi, \dot{\xi})\pi = \underbrace{Y_M\ddot{\theta} + Y_C\dot{\theta} + Y_g}_{\rho} + \underbrace{Y_K\theta}_E + \underbrace{Y_D\dot{\theta}}_{\eta} \quad (4.15)$$

Persistent Excitation and Identifiability

A well-known limitation of adaptive control is the requirement of *persistent excitation* (PE) for exact parameter convergence. The PE condition requires that the regressor matrix Y remains sufficiently rich over time, namely:

$$\exists \alpha, T > 0 \quad \text{s.t.} \quad \int_t^{t+T} Y^T(\tau)Y(\tau) d\tau \geq \alpha I \quad \forall t \geq 0. \quad (4.16)$$

this condition means that the information associated to the regressor configuration at a time instant τ has to be in all its components larger than a minimum degree of excitation represented by the value α . When this condition is not satisfied, the update law (4.17) may still drive $s \rightarrow 0$ and $e \rightarrow 0$, but the parameter estimates $\hat{\pi}$ need not converge to the true values π .

In our setting, the degree of excitation, i.e. the amount of excitation associated to each element of the regressor, differs substantially across the three parameters. Since ρ acts on both the inertial and the gravitational contributions to the regressor, it is excited by any non-trivial motion of the object — accelerations, velocity changes, and gravity all carry information about the mass distribution. By contrast, E appears only through the elastic restoring term $Y_K\theta$, which has more information when the object undergoes significant shape changes. Similarly, η is excited only when the object's velocity is non-zero and varying. As a consequence, in tasks with limited dynamic content — such as slow quasi-static repositioning — the regressor may be poorly excited with respect to E and η , leading to slow or incomplete convergence of those estimates.

Parameter Update Law

The adaptive update law from Lemma 4.1.6 drives the parameter estimate $\hat{\pi}$ according to

$$\dot{\hat{\pi}} = -\Gamma Y(\theta, \dot{\theta}, \xi, \dot{\xi}) s \quad (4.17)$$

where $s = \dot{q} - \xi$ is the composite error signal and $\Gamma \in \mathbb{R}^{3 \times 3}$ is a symmetric positive-definite adaptation gain matrix. The parameter estimation error $\tilde{\pi} = \hat{\pi} - \pi$ converges to zero asymptotically, as guaranteed by the stability analysis of Lemma 4.1.6, provided that the regressor Y is persistently exciting over time.

Tuning of the Adaptation Gain

To mitigate the effects of poor excitation, the matrix Γ can be tuned to accelerate convergence in specific parameter directions. In this work, Γ is taken as a diagonal matrix

$$\Gamma = \text{diag}(\gamma_\rho, \gamma_E, \gamma_\eta) \quad (4.18)$$

where each diagonal entry scales the adaptation speed for the corresponding parameter. In tasks with limited excitation of E and η , larger values of γ_E and γ_η are used to compensate for the reduced regressor signal, at the cost of increased sensitivity to noise in those channels. The choice of Γ therefore involves a trade-off between convergence speed and robustness, which is analyzed experimentally in Chapter 5.

Physical Interpretation of the Estimates

Beyond their role in the control law, the time varying estimates $\hat{\rho}$, $\hat{E}$, and $\hat{\eta}$ carry direct physical meaning. The density estimate $\hat{\rho}$ characterizes the object's mass per unit volume, which together with the known geometry determines the total mass and its distribution along the object. The Young's modulus estimate $\hat{E}$ quantifies the object's resistance to bending deformation — a quantity that is notoriously difficult to measure directly for flexible cables without specialized equipment. The damping estimate $\hat{\eta}$ reflects the rate at which the object dissipates energy during oscillation. Obtaining these quantities as a byproduct of closed-loop control — without requiring dedicated offline identification experiments — is one of the principal practical advantages of the adaptive framework presented in this chapter.

5

Simulations

5.1 Results

This chapter presents the simulation results used to validate the adaptive control framework developed in Chapter 4.1. Simulations serve two complementary purposes. First, they allow us to assess the performance of the proposed controller in a safe environment where ground truth parameter values are known exactly, enabling a rigorous evaluation of both tracking accuracy and parameter convergence that would be impossible to obtain from physical experiments alone. Second, they provide a systematic comparison against fixed-gain baselines — namely a PD + model cancellation controller and a PID controller, developed in [15, 17] — to quantify the practical benefit of adaptation.

Three aspects of performance are evaluated across all simulation scenarios:

1. **Tracking accuracy:** the ability of the controller to regulate the DLO endpoint to the desired position and orientation, measured through the error $e(t) = q(t) - r(t)$ over time.
2. **Parameter convergence:** the evolution of the estimates of the physical parameters associated to the material's density, the stiffness and the elastic damping, i.e. the variables $\hat{\rho}$, $\hat{E}$, and $\hat{\eta}$, towards their true values, with particular attention to the different convergence rates discussed in Chapter 4.
3. **Robustness across object types:** the consistency of performance when the physical properties of the DLO are varied, simulating the range of cables considered in this work.

To achieve (3) we will consider the following DLO's characteristics:

DLO	ρ [kg/m^3]	E [Pa]	η	L [m]
DLO1	3162.6	$2.53 \cdot 10^7$	$1.05 \cdot 10^6$	0.77
DLO2	3546.6	$2.207 \cdot 10^7$	$1.25 \cdot 10^6$	0.77
DLO3	4514.6	$4.05 \cdot 10^7$	$1.6 \cdot 10^6$	0.89
DLO4	5161.2	$4.802 \cdot 10^7$	$1.75 \cdot 10^6$	0.89

Table 5.1: Physical characteristics of the simulated Deformable Linear Objects used in the simulations

All simulations were implemented in MATLAB, using the dynamic models derived in Chapter 3. The object parameters used as ground truth span a representative range of stiffness, damping, and volumetric density, reflecting the variability observed across the physical objects DLO1-DLO4. In each case, the controller is initialized with a deliberately mismatched parameter estimate $\hat{\pi}(0) \neq \pi$, to test adaptation from realistic initial conditions. The PD + model cancellation and PID baselines are tuned separately but remain fixed in the different scenarios to showcase the differences in the controllers.

The remainder of this chapter is organized as follows. Section 5.1.1 addresses the fixed single-joint case, where the DLO is attached to a single revolute joint and the underactuated adaptive controller of Lemma 2 is applied. Section 5.1.2 extends the study to a full single-arm manipulator grasping the DLO at one end. Section 5.1.3 extends the analysis to the dual-arm case, where both ends of the object are controlled, corresponding to the fully actuated setting of Lemma 1 and the under-actuated case using Lemma 2.

5.1.1 Fixed-Arm

The first, simplified model considers control only of the revolute joint grasping the DLO, while the unactuated degrees of freedom θ_u evolve according to the system's natural dynamics. This is an underactuated setting for which the controller of Lemma 2 was specifically designed. The tasks considered here are actuated coordinate regulation and endpoint regulation — driving the distal end of the object to a specified position — which requires the controller to indirectly shape the object's equilibrium configuration through the robot's base angle. The scenarios will first be evaluated with adaptive control and then compared with the performance achieved with PD+ and PID.

Desired Actuated Coordinate Control

The gains of the adaptive control in use are

$$K = 0.09, K_n = 10^{-3}, \Lambda_1 = 7, \quad (5.1)$$

$$\Lambda_2 = 9, \Gamma = \text{blkdiag}(4, 20, 500) \quad (5.2)$$

The tuning was informed by both knowledge of the parameter and the regressor, as well as by experimental results. We consider the regulation task:

$$\theta_a(t) = \begin{cases} 0, & 0 \leq t \leq 5s, \\ -\pi/4, & 5s \leq t \leq 10s, \\ \pi/2, & 10s \leq t \leq 15s, \\ -\frac{1}{2}, & 15s \leq t \leq 20s, \end{cases}$$

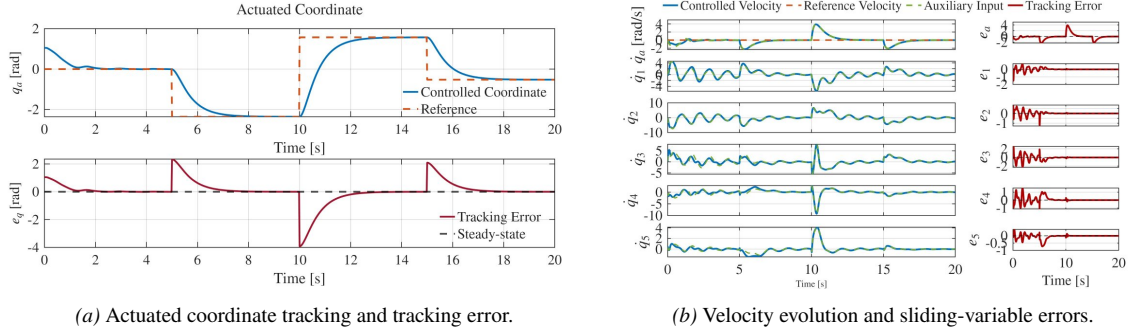


Figure 5.1: Desired actuated-coordinate regulation using the adaptive controller from the initial configuration $\theta_a = \pi/3$. Panel (a) shows the tracking performance of the actuated coordinate, while panel (b) reports the corresponding velocity response and sliding-variable errors.

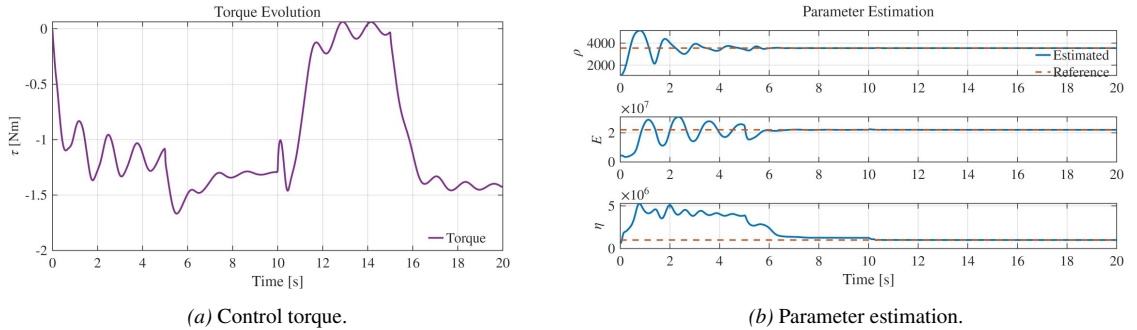


Figure 5.2: Control effort and adaptive parameter estimation for the desired actuated-coordinate regulation task shown in Fig. 5.1a, using the adaptive controller. Panel (a) reports the control torque required to achieve the observed tracking performance, while panel (b) shows the evolution of the estimated dynamic parameters starting from the initial estimate $\pi(0) = [10^3, 4 \cdot 10^5, 6 \cdot 10^5]$.

The results achieved can be seen in Figure 5.1 and Figure 5.2. In Figure 5.1a, the actuated coordinate q_a tracks each new reference step closely, and the tracking error e_a is driven back toward its steady-state band after every transition; consistently with the online nature of the estimation, the largest excursion occurs at the very first step and around 1.8–2 s, when the parameter estimate $\hat{\pi}$ is still far from its true value π , while later steps are tracked with visibly smaller and shorter-lived errors, as the estimate has by then approached the true parameters. The velocity-level behavior in Figure 5.1b confirms the two-stage structure of the underactuated design: the actuated-coordinate channel is smooth and closely follows its reference, while the remaining (unactuated) shape channels exhibit a persistent, higher-frequency oscillation — an

expected consequence of regulating these coordinates indirectly through the collocated variable ξ — yet at every step, the sliding-type errors satisfy

$$\tilde{\xi} = q_a - \xi \rightarrow 0 \quad s = \dot{q} - \xi \rightarrow 0,$$

confirming that the auxiliary and sliding variables are correctly driven to the origin even when the raw velocities keep oscillating.

The torque required to realize this tracking (Figure 5.2a) is dominated by a comparable oscillatory component, with the largest transient appearing around $t \approx 10\text{--}15$ s, in correspondence with the reference step of largest amplitude; outside these transitions the torque remains bounded and free of the sharp spikes that will be seen for the fixed-gain baselines below. Finally, Figure 5.2b shows the three parameter estimates converging toward the ground-truth values indicated by the dashed reference lines, but at markedly different rates: $\hat{\rho}$ is already close to ρ within the first few seconds and only shows small residual ripple, $\hat{E}$ requires a longer transient before settling near E , and $\hat{\eta}$ is the slowest of the three, first overshooting and then relaxing toward η only toward the end of the simulation window. This ordering — $\hat{\rho}$ fastest, $\hat{\eta}$ slowest — is consistent with the excitation-dependent convergence rates discussed in Section 5.3, since the damping term η enters the regressor multiplied by the (comparatively small) strain-rate terms and is therefore the hardest to excite persistently with a single actuated coordinate.

We now show the different performance in the control performance of three different control actions: PID, PD+ and Adaptive control. The control gains of the remaining controllers are:

$$\text{PID: } K_P = 0.6, K_I = 0.3, K_D = 0.15$$

$$\text{PD+: } K_P = 0.8, K_D = 0.45$$

These control strategies have diverse characteristics. PID is a model free solution; however, tuning can affect the performance that is achievable. PD+ is a partially model dependent strategy, thus using a model different from the nominal will introduce an error. Adaptive Control is a model dependent strategy that can correct modeling errors to achieve desired performance.

The characteristics of the DLOs are reported in Table 5.1.

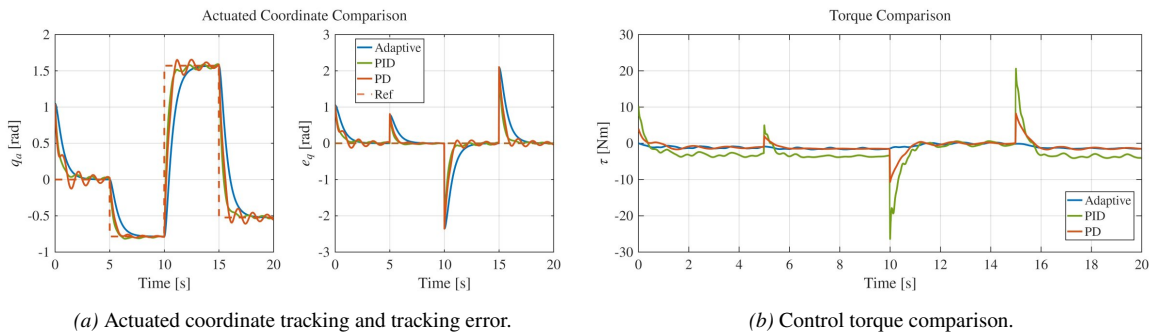


Figure 5.3: Regulation task comparison among the Adaptive (blue), PID (green), and PD+ (red) controllers against the reference (orange), in the nominal case (DLO2). Panel (a) shows the actuated-coordinate tracking together with the tracking error, while panel (b) shows the corresponding control torque for the three strategies.

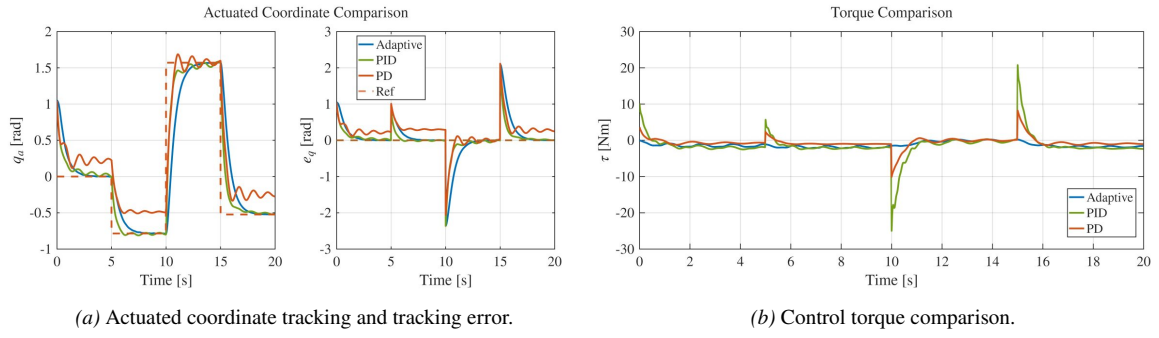


Figure 5.4: Regulation task comparison among the Adaptive (blue), PID (green), and PD+ (red) controllers against the reference (orange), considering DLO3. Panel (a) shows the actuated-coordinate tracking together with the tracking error, while panel (b) shows the corresponding control torque for the three strategies.

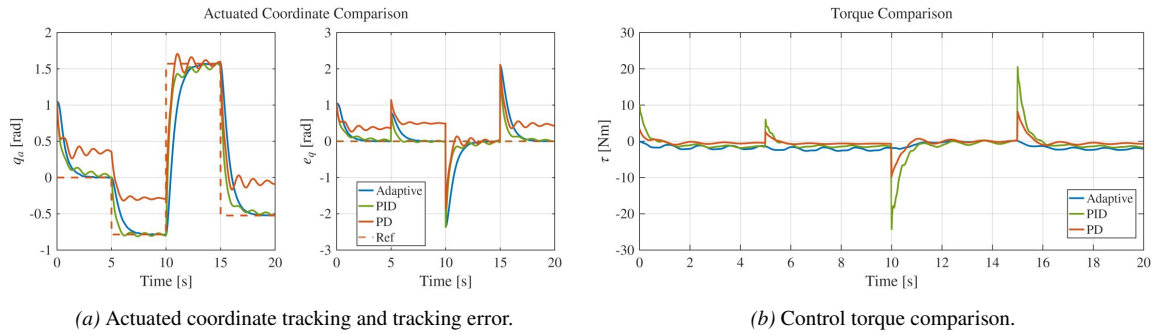


Figure 5.5: Regulation task comparison among the Adaptive (blue), PID (green), and PD+ (red) controllers against the reference (orange), considering DLO4. Panel (a) shows the actuated-coordinate tracking together with the tracking error, while panel (b) shows the corresponding control torque for the three strategies.

As it can be seen in Figure 5.3a, 5.4a, 5.5a, the performance in the nominal case is as we would expect: the PID and PD+ are faster, and both manage to reach steady-state error. Indeed, in the tuning phase, to be able to perform an acceptable comparison between the different control strategies, we have considered as performance metrics the overshoot and the settling time of the actuated coordinate in the nominal case, and the necessary torque to complete the motion. The reached performance were:

$$\text{Adaptive Control : } M_s = 0.9\% \quad t_s = 2.081s$$

$$\text{PID Control : } M_s = 0.8\% \quad t_s = 1.951s$$

$$\text{PD+ Control : } M_s = 3.1\% \quad t_s = 2.041s$$

However, by changing the characteristics of the DLO, all three controllers obtain a decrease in performance to the point where PD+ does not achieve the steady state. Indeed, the compensating action required to achieve steady state error, in this case, acts as a constant error due to the imprecise knowledge of the true model. The PID becomes slower and exhibits larger errors in the transient. Furthermore, the tuning of the controller was developed to achieve the task in the nominal case with similar control effort. The result is a torque evolution whose magnitude

differs by one order of magnitude between the PID–PD+ and adaptive control strategies. As a result, the torque produced by the Adaptive control is quite smaller and does not present uncomfortable peaks.

Adaptability

In this section, we report the true ability of the control implemented in this work.

We now consider the control of 4 different DLOs, building on the table presented in the previous task, Tab.5.1. The test was developed with the same initial conditions and the same tracking task.

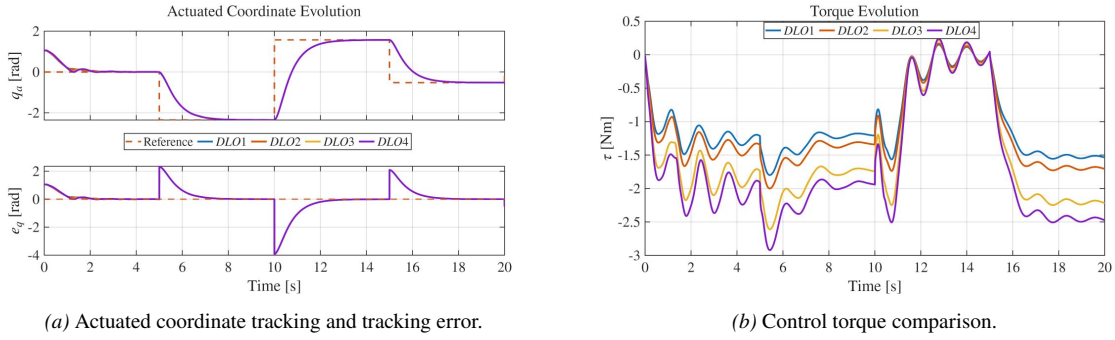


Figure 5.6: Adaptive controller performance across the four DLOs (DLO1–DLO4) considered in this work, completing the set-point tracking task from the same initial condition $\theta_a = \pi/3$. Panel (a) shows the actuated-coordinate tracking together with the tracking error, while panel (b) shows the corresponding control torque for each object.

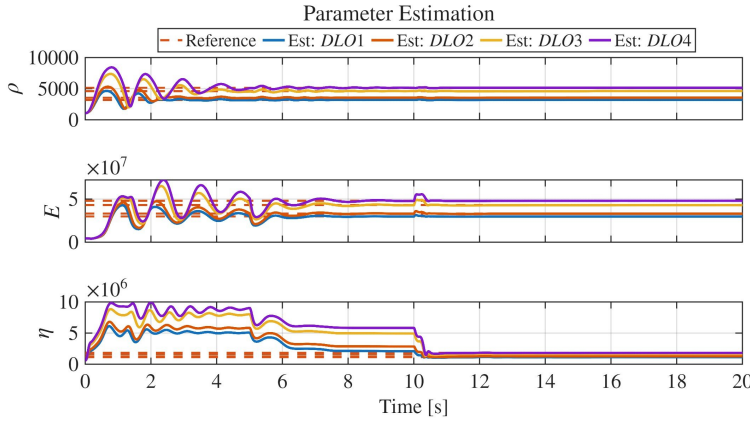


Figure 5.7: Parameter estimation comparison considering the 4 kind of DLO, starting from the initial condition $\pi(0) = [10^3, 4 \cdot 10^5, 6 \cdot 10^5]$

The tracking performance in Figure 5.6a is essentially indistinguishable across the four objects DLO1–DLO4: the actuated coordinate follows the same reference profile with comparable error regardless of the underlying mass, stiffness or damping, confirming that the parameter mismatch is compensated for online rather than relying on a fixed nominal model. The price of this invariance is visible in the torque plot of Figure 5.18b, where the required torque scales directly with the object’s physical properties: the heaviest and stiffest object among the four produces markedly larger torque swings than the lightest one, while all four remain smooth

and free of the peaks that characterize the fixed-gain baselines. The parameter estimates in Figure 5.7 converge toward the correct, object-specific reference value in every case, with $\hat{\rho}$ and $\hat{E}$ settling within the first few seconds for all four objects and $\hat{\eta}$ again the slowest to converge, particularly for the heaviest object, which requires close to the full simulation horizon to reach its steady-state estimate. This confirms that the convergence-rate ordering observed for a single object in the previous section is not an artifact of that specific parameter set, but a structural property of the regressor that persists across the whole range of DLOs considered in this work.

Tip Regulation Control

This paragraph presents a more realistic task, where the problem to be solved is the control of the unactuated part of the model, namely the position control of the moving extremity of the DLO. The interest behind this regulation challenge lies in the possible application to industrial robotics, for example in the motion of cables in industrial facilities.

The plots below showcase the results obtained while solving by means of the same control gain as in Eq.5.1.1.

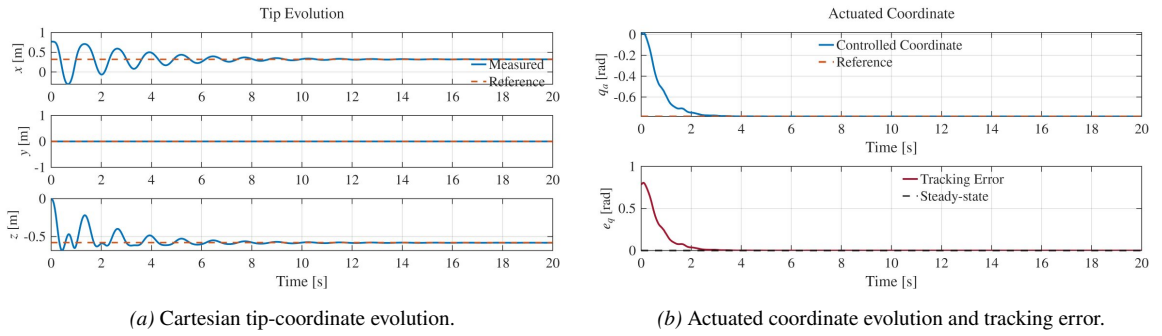


Figure 5.8: Tip-regulation task performance using the adaptive controller with the inverse kinematics algorithm based on the fully known model. Panel (a) shows the evolution of the Cartesian tip coordinates toward the desired position, while panel (b) shows the corresponding actuated-coordinate evolution and tracking error.

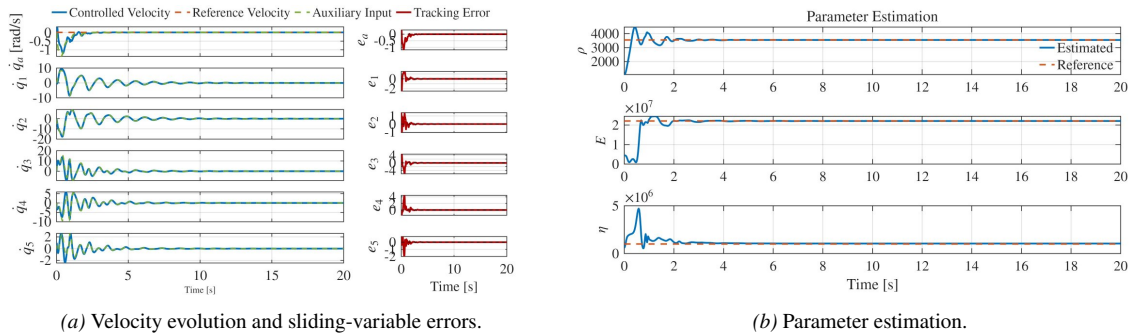


Figure 5.9: Velocity-level behaviour and parameter estimation for the tip-regulation task shown in Fig. 5.8a. Panel (a) reports the actuated and unactuated velocity components together with the corresponding sliding-variable errors, while panel (b) shows the evolution of the estimated dynamic parameters starting from the initial estimate $\pi(0) = [10^3, 4 \cdot 10^5, 6 \cdot 10^5]$.

So the user gives a desired tip position p_d , and through the use of an inverse kinematics algorithm, the controller is fed a desired actuated coordinate. The structure of the algorithm can

be implemented in two different scenarios: one where the model exploits the fully known model, and the other where the model could exploit the evolution of the estimation of the parameters. The algorithm can generally be described by the following minimization problem:

$$\begin{aligned} \min_{q,u} \quad & \|p(q) - p_d\|^2 \\ \text{s.t.} \quad & Kq - \hat{Q}(q) = A(q)u \end{aligned} \quad (5.3)$$

$$\begin{aligned} \min_{q,u} \quad & \|p(q) - p_d\|^2 \\ \text{s.t.} \quad & \hat{K}q - \hat{Q}(q) = A(q)u \end{aligned} \quad (5.4)$$

where the evolution of $p(q)$ is associated with the progress of the configuration, while preserving static equilibrium. Both methods solve this as a nested two-level problem: an outer level, in which the Jacobian of the cost function $\|p(q) - p_d\|^2$ is used to take a Gauss-Newton step that updates the configuration q ; and an inner level, in which a Newton iteration re-enforces the active equilibrium constraint at this updated q , adjusting the remaining generalized coordinates (and recovering u) until $Kq - Q(q) = A(q)u$ holds. The minimization is solved over the generalized coordinates and, upon convergence, delivers the associated actuated configuration θ .

As it can be seen in Figure 5.8a, the actuated coordinate is driven monotonically from its initial value towards the value required to place the tip at the desired position, with the tracking error decaying accordingly. The sliding-type errors again converge to zero, as shown in Fig. 5.9a, and the parameter estimation is achieved almost perfectly, as depicted in Fig. 5.9b, where the lack of excitation in the estimation of the damping η due to poor velocity variation, prevents achieving the perfect estimation of the three parameters.

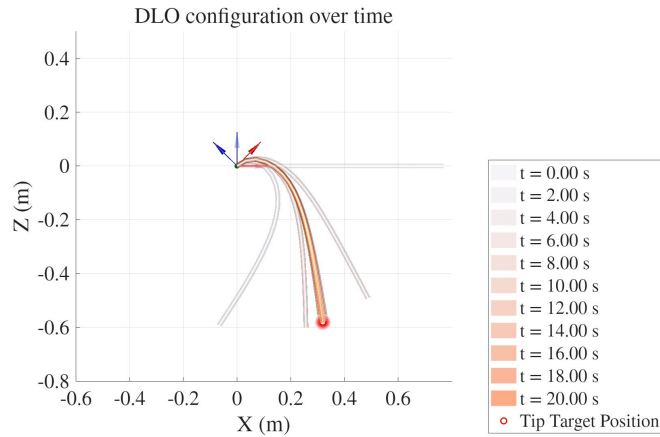


Figure 5.10: DLO configuration evolution in time while achieving the tip's regulation task

Finally, Figure 5.10 shows the actual evolution of the cable in time, up to the point of regulating to the desired configuration.

A similar result is obtained when we consider the time evolving inverse kinematics algorithm described in Eq.5.4. As we can see from Figure 5.11a, the evolution of the DLO has a different development with respect to the previous solution. However, due to the parameter estimation rapid improvement, related to the large excitation resulting from the actuation coordinate requested by the algorithm, the correct configuration is computed in the following iteration of the

algorithm; the final DLO configuration coincides with the previous.

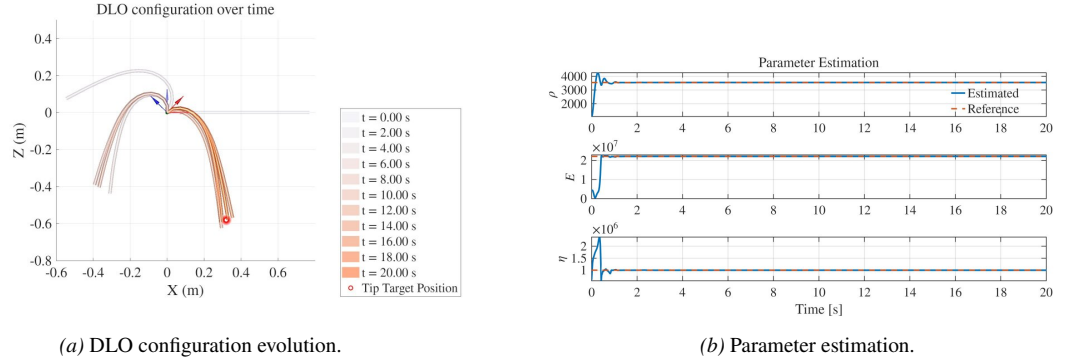


Figure 5.11: DLO configuration evolution and parameter estimation for the tip-regulation task solved with the inverse kinematics algorithm based on the online parameter-estimate update (Eq. 5.4). Panel (a) shows the resulting configuration evolution of the DLO, while panel (b) shows the corresponding parameter estimation starting from the initial estimate $\pi(0) = [10^3, 4 \cdot 10^5, 6 \cdot 10^5]$.

5.1.2 Single-Arm Manipulation

Having validated the adaptive law on the simplified single-revolute-joint test bench of Section 5.1.1, we now consider the more general case of a single-arm manipulator grasping the DLO at one of its ends. This exploits the same structure used in another relevant control work on deformable objects [21]. The additional degrees of freedom of the manipulator allow the grasped end to be driven anywhere within the reachable workspace, rather than along the single rotational coordinate of the fixed-joint case, while the object's shape degrees of freedom θ_u remain unactuated. The setting therefore remains underactuated, and the controller of Lemma 2 is again the natural choice. As in Section 5.1.1, two tasks are evaluated: actuated coordinate tracking and endpoint (tip) regulation, first with the adaptive controller alone and then in comparison with the PD+ and PID baselines, so that the benefit of adaptation can be assessed as the complexity of the plant increases.

Desired Actuated Coordinate Control

The gains of the adaptive controller used in this scenario are

$$K = 1, K_n = 1, \Lambda_1 = 6, \quad (5.5)$$

$$\Lambda_2 = 10, \Gamma = \text{blkdiag}(2, 10, 30) \quad (5.6)$$

As in Section 5.1.1, the tuning was carried out jointly based on the known parameter bounds, the structure of the regressor, and preliminary experimental results.

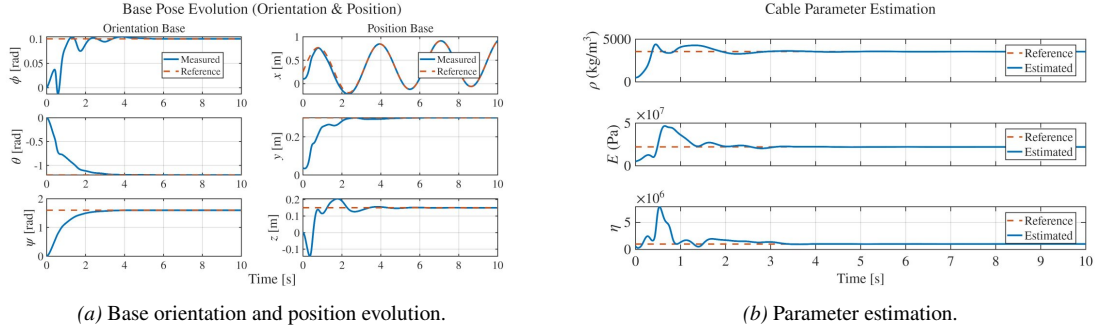


Figure 5.12: Base-pose tracking and parameter estimation for the moving-arm scenario, using the adaptive controller to converge to a sinusoidal trajectory varying along the x-axis. Panel (a) shows the evolution of the base orientation and position, while panel (b) shows the evolution of the estimated dynamic parameters starting from the initial estimate $\pi(0) = [10^3, 4 \cdot 10^5, 6 \cdot 10^5]$.

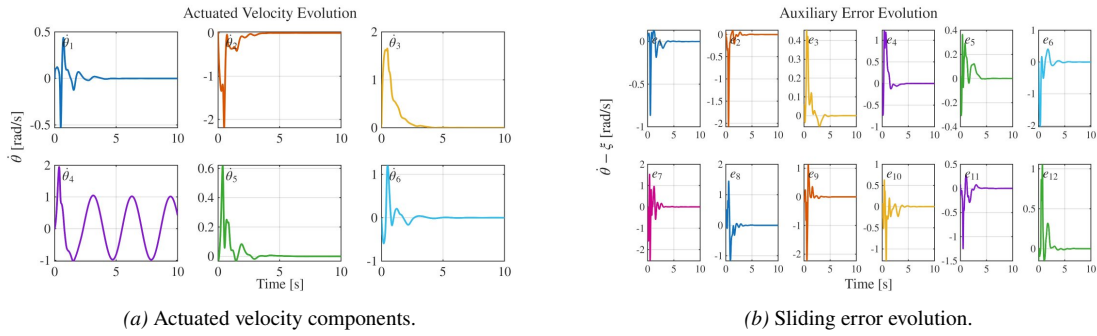


Figure 5.13: Actuated-velocity and sliding-variable behaviour for the moving-arm sinusoidal-tracking task shown in Fig. 5.12a. Panel (a) shows the evolution of the actuated velocity components, while panel (b) shows the corresponding sliding-error evolution for both the actuated and unactuated components.

The results are reported in Figure 5.12a-5.13b. As in the fixed-joint case, Figure 5.12a shows both the base orientation (ϕ_b, θ_b, ψ_b) and the base position (x_b, y_b, z_b) converging smoothly from their initial values to the desired reference within the first few seconds, with only a small overshoot on the position channels and no oscillatory residual — a first indication that, despite the added kinematic complexity of a full manipulator, the actuated boundary condition remains well behaved. The parameter estimates in Figure 5.12b confirm the same convergence-rate ordering identified in Section 5.1.1, with $\hat{\rho}$ settling first and $\hat{\eta}$ requiring the longest transient. The actuated-velocity evolution in Figure 5.13a shows that most of the six shape-related velocity components settle within a few seconds, while one component, in particular, retains a persistent, larger-amplitude oscillation for the whole simulation window as it is associated with the sinusoidal trajectory; the corresponding auxiliary error in Figure 5.13b shows the same component as the slowest (and only partially damped) among the twelve reported, while the remaining errors converge close to zero quickly. This is consistent with the fixed-joint case: as the setting is still underactuated in Θ , 6 internal shape coordinates are only regulated indirectly and are therefore expected to retain some residual motion even at steady state. Consistent with Section 5.1.1, the sliding-variable behavior

$$\tilde{\xi} = q_a - \xi \rightarrow 0 \quad s = \dot{q} - \xi \rightarrow 0$$

is confirmed in Figure 5.12b, and the estimated parameters converge toward their true values as shown in Figure 5.13b.

Comparison of Control Methods The physical characteristics of the simulated DLOs are unchanged with respect to the fixed-joint scenario and are reported in Table 5.1. As before, the comparison is carried out against PID and PD+ baselines with gains

$$\text{PID: } K_P = 40, K_I = 20, K_D = 8$$

$$\text{PD+: } K_P = 6, K_D = 2$$

tuned, as in Section 5.1.1, to achieve comparable performance in the nominal case, so that the effect of varying the object's physical properties can be isolated. The corresponding average nominal-case performance metrics are:

$$\text{Adaptive Control : } \bar{M}_s = 8.92\% \quad \bar{t}_s = 2.46 \text{ s}$$

$$\text{PID Control : } \bar{M}_s = 9.1\% \quad \bar{t}_s = 2.3945 \text{ s}$$

$$\text{PD+ Control : } \bar{M}_s = 12.64\% \quad \bar{t}_s = 2.223 \text{ s}$$

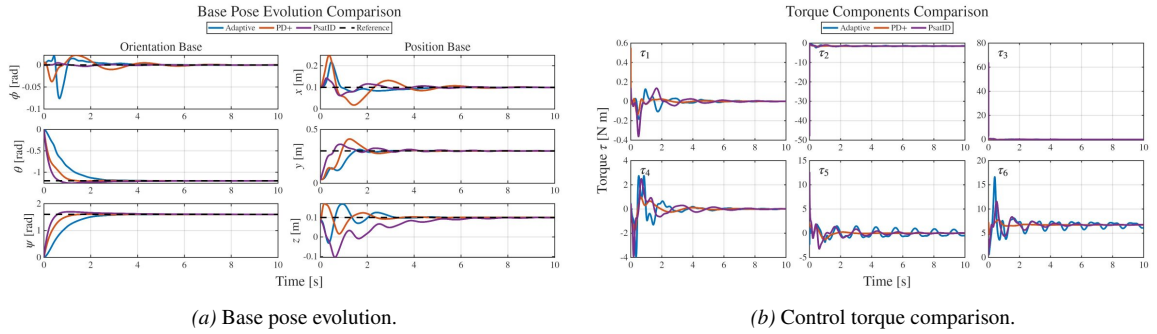


Figure 5.14: Base-pose and control-torque comparison among the Adaptive (blue), PD+ (orange), and PID (purple) controllers completing the configuration-convergence task for the moving-arm scenario, in the nominal case (DLO2). Panel (a) shows the base-pose evolution, while panel (b) shows the corresponding control torque for the three strategies.

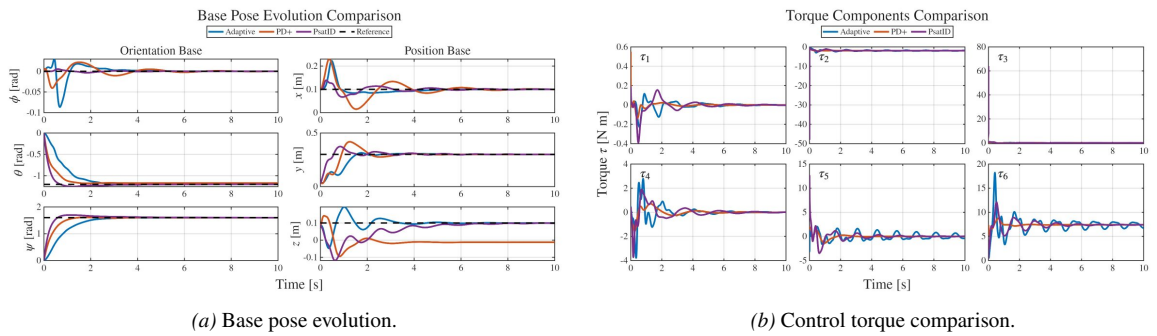


Figure 5.15: Base-pose and control-torque comparison among the Adaptive (blue), PD+ (orange), and PID (purple) controllers completing the configuration-convergence task for the moving-arm scenario, considering DLO3. Panel (a) shows the base-pose evolution, while panel (b) shows the corresponding control torque for the three strategies.

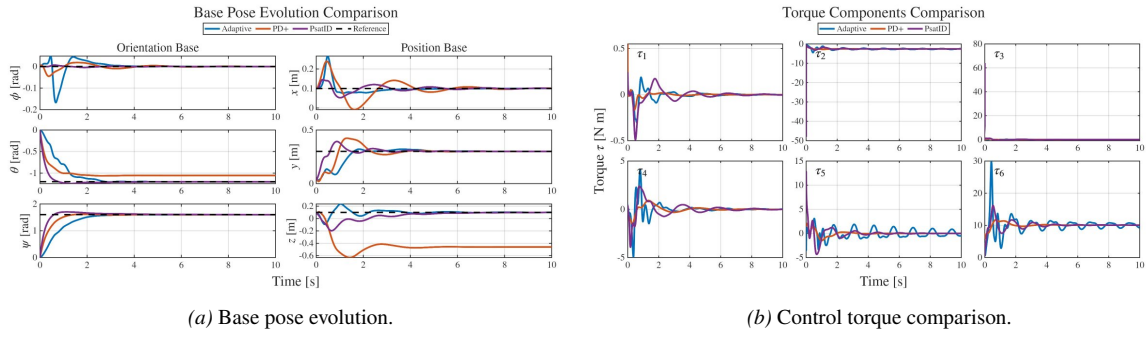


Figure 5.16: Base-pose and control-torque comparison among the Adaptive (blue), PD+ (orange), and PID (purple) controllers completing the configuration-convergence task for the moving-arm scenario, considering DLO4. Panel (a) shows the base-pose evolution, while panel (b) shows the corresponding control torque for the three strategies.

As in Figure 5.14a, the base pose is tracked to essentially the same reference trajectory by all three controllers for DLO2, the nominal object on which the gains were tuned, with only minor differences in the transient. The torque plot in Fig5.14b, however, already reveals a gap at this stage: the Adaptive controller distributes the required effort smoothly across its torque components, while PID and, to a lesser extent, PD+ present sharper transients on the components associated with the base orientation. Moving to DLO3 and DLO4, whose mass and stiffness depart increasingly from the nominal value used to tune the baselines, the pose-tracking curves for PID and PD+ develop a visibly larger overshoot and a slower return to the reference, whereas the Adaptive controller's response changes comparatively little across the three objects. This degradation is even more evident on the torque side, see Fig.5.14b-5.16b; several torque components for PD+ grow markedly in amplitude from DLO2 to DLO4, with the largest components approaching values several times higher than in the nominal case, while the Adaptive controller's torque remains of comparable magnitude throughout, mirroring the order-of-magnitude difference already reported for the fixed-joint case in Section 5.1.1.

Adaptability

As for the fixed-joint case, the base pose evolution is essentially overlapping across the four objects DLO1-DLO4, confirming that the adaptive law compensates for the change in physical parameters rather than depending on knowledge of a nominal object.

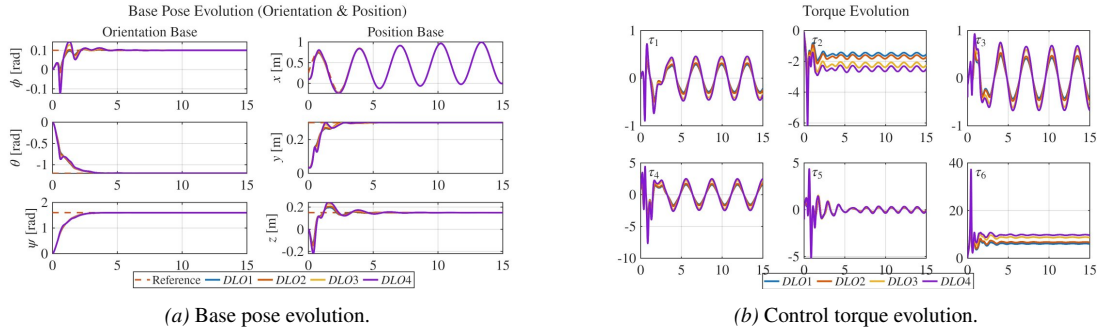


Figure 5.17: Adaptive controller performance across the four DLOs (DLO1–DLO4) considered for the moving-arm scenario. Panel (a) shows the base-pose evolution, while panel (b) shows the corresponding control torque for each object.

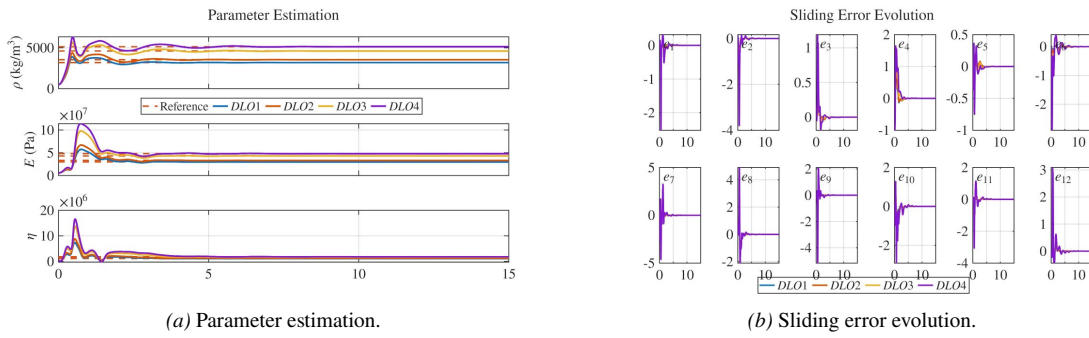


Figure 5.18: Parameter estimation and sliding-variable behaviour across the four DLOs (DLO1–DLO4) considered for the moving-arm scenario. Panel (a) shows the evolution of the estimated dynamic parameters, while panel (b) shows the corresponding sliding-error evolution.

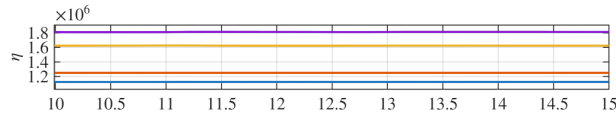


Figure 5.19: Zoom of the Last five second evolution of the estimation of the material's damping in the scenario comparing different DLOs

The torque plots again scale with the object's physical properties, with the heaviest and stiffest object requiring visibly larger torque amplitudes on several components than the lightest one, while remaining free of the sharp transients that characterize the fixed-gain baselines. The parameter estimates converge to their correct, object-specific values for all four DLOs, and the error signals settle close to zero over comparable time-scales regardless of the object. The zoom on the last five seconds of the damping estimate in Figure 5.19 shows that, even $\hat{\eta}$ has fully settled to a constant value for every object.

Tip Regulation Control

As previously done in the revolute joint case, the following paragraph presents the results achieved in the regulation of the tip position. The desired extremity position is mapped to a desired actuated coordinate configuration through the use of the inverse kinematics algorithm

described in Eq.5.3 and then used as p_d .

The problem was solved exploiting the same control gains as in Eq. 5.1.2.

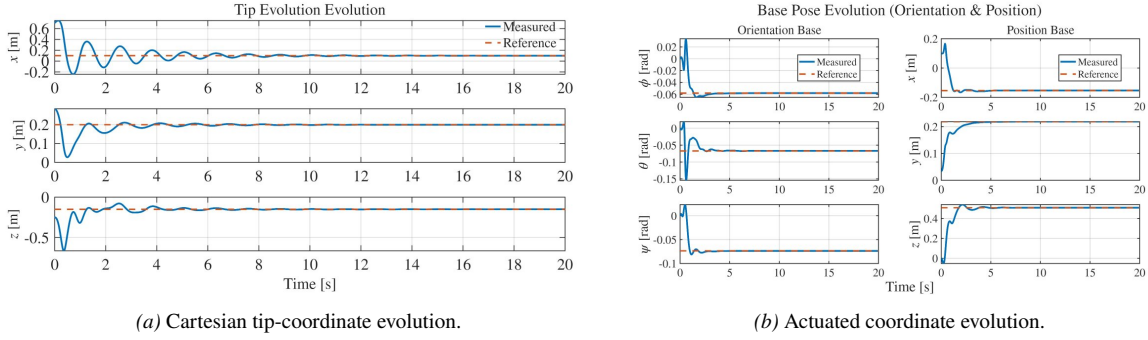


Figure 5.20: Tip-settling task performance for the moving-arm scenario using the adaptive controller with the inverse kinematics algorithm based on the fully known model (Eq. 5.3). Panel (a) shows the evolution of the Cartesian tip coordinates, while panel (b) shows the corresponding actuated-coordinate evolution.

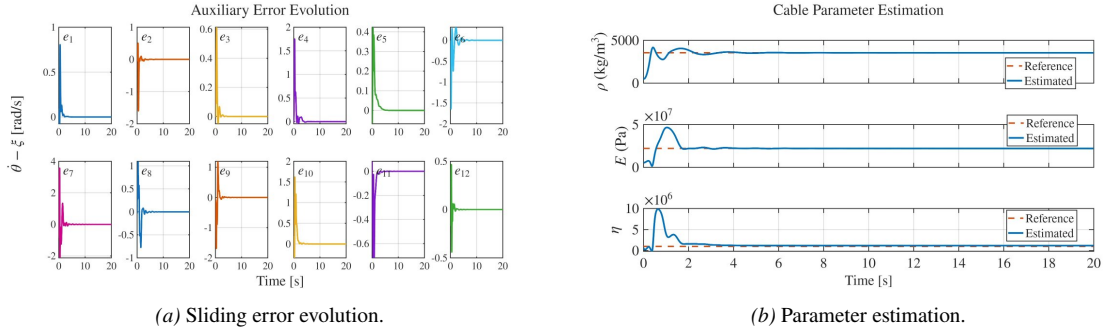


Figure 5.21: Sliding-variable behaviour and parameter estimation for the tip-settling task shown in Fig. 5.20. Panel (a) shows the evolution of the sliding error, while panel (b) shows the corresponding parameter estimation.

As shown in Figure 5.20a, the desired actuated coordinate evolves towards the value required to settle the tip at the desired position, with the sliding variable s converging to zero, as in Figure 5.20b. The parameter estimates again converge correctly, as depicted in Figure 5.21b.

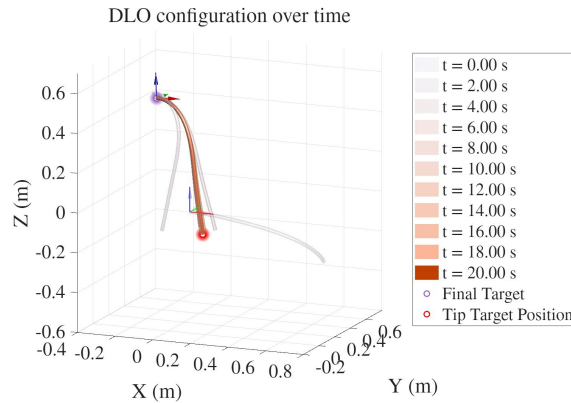


Figure 5.22: Time Evolution of the single arm DLO configuration while achieving tip's regulation

Moving a step further, the inverse kinematics model could be adjusted based on the update of the parameters, as described in Eq.5.4. As shown in the following Figure 5.23a, the task can be easily completed without the need to adjust the actuated coordinate delivered by the inverse kinematics algorithm, which evolves with the convergence of the parameters to the reference model.

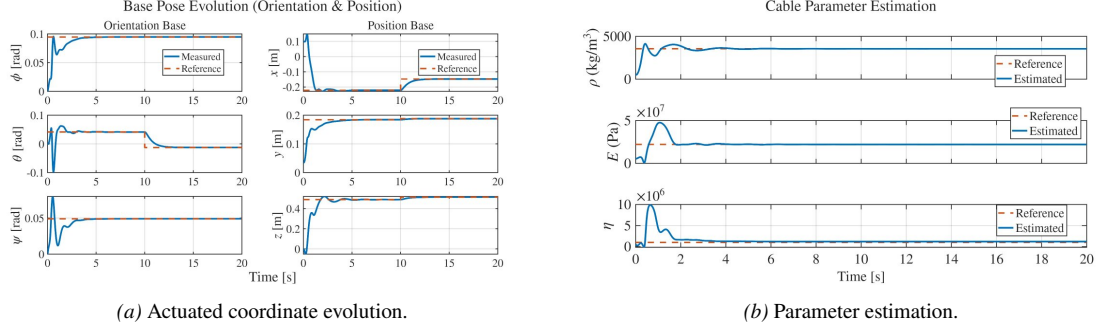


Figure 5.23: Actuated-coordinate evolution and parameter estimation for the tip-regulation task solved with the inverse kinematics algorithm based on the online parameter-estimate update (Eq. 5.4). Panel (a) shows the actuated-coordinate evolution required to reach the desired tip position, while panel (b) shows the corresponding parameter estimation.

Figure 5.24 clearly shows how the cable configuration first misses the desired tip position to complete the task, and once the estimation is completed, it delivers the necessary θ_a to reach the tip's regulation.

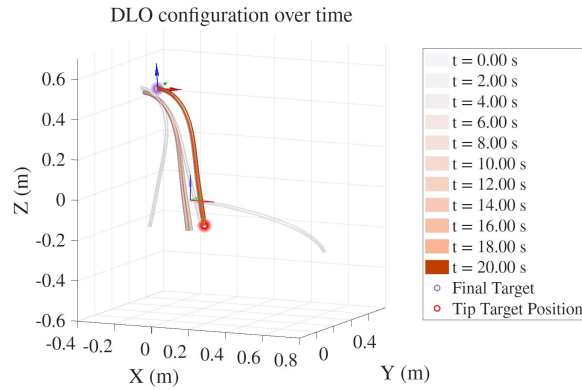


Figure 5.24: Time Evolution of the single arm DLO configuration while achieving tip's regulation, using the estimation of the parameters for computing the actuated coordinate

5.1.3 Dual-Arm

The dual-arm scenario considers two manipulators grasping the DLO at both ends simultaneously. This configuration removes the under-actuation constraint: with both boundary conditions under direct control, the object's shape can be steered more freely and the fully actuated adaptive controller of Lemma 1 applies. Beyond improved controllability, the dual-arm setting is also of practical interest for tasks requiring precise simultaneous control of both the position and the orientation of multiple points along the object — such as cable routing or assembly operations.

The richer excitation provided by two independent control inputs also creates more favorable conditions for parameter convergence, particularly for $\hat{E}$ and $\hat{\eta}$.

Furthermore, to increase the precision of the model with the experimental set, we will also consider the under-actuated case, where we will apply Lemma 2.

Desired Actuated Coordinate Control

As already performed in the tasks with One arm manipulation, we will now show the results of performing a desired pose tracking. The actuated coordinates of this model capture the pose of both ends of the DLO, i.e. the cartesian position and the Euler angle configuration

$$\theta = [\phi_b, \sigma_b, \psi_b, x_b, y_b, z_b, \phi_t, \sigma_t, \psi_t, x_t, y_t, z_t]^T \quad (5.7)$$

We will exploit, in this case, the fully-actuated case, with different control types generating the necessary wrench in the task-space.

The goal of our Adaptive Controller is to obtain comparable performance with the PID and PD+ controllers, while being able to estimate the physical parameters.

Fully-Actuated Case (Lemma 1) The gains of the adaptive controller used for the fully-actuated dual-arm task are

$$K = 1, \Lambda_1 = 6, \quad (5.8)$$

$$\Lambda_2 = 10, \Gamma = \text{blkdiag}(2, 10, 20) \quad (5.9)$$

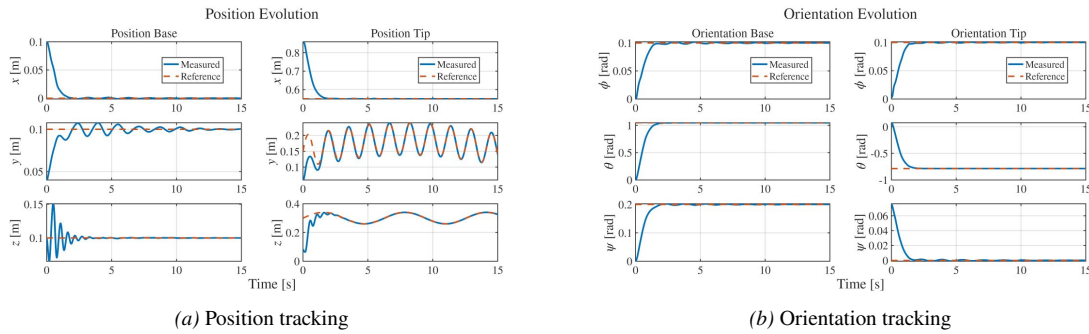


Figure 5.25: Position and orientation evolution of both end-points of the nominal DLO (DLO2). Considering the sinusoidal trajectory tracking with initial configuration $\theta = [0, 0, 0.1, 0.035, 0.09, 0, 0.65, 0.87, 0, 0.1]^T$

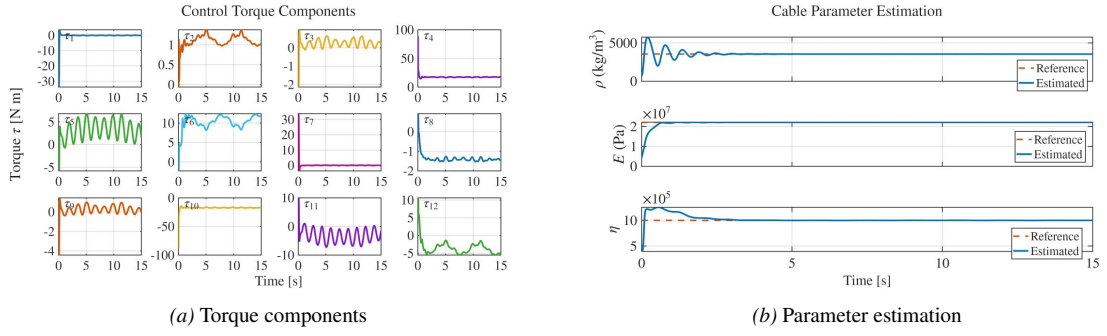


Figure 5.26: Control effort and parameter convergence under the fully-actuated adaptive controller in the nominal case DLO (DLO2), with parameter initial condition $\pi(0) = [10^3, 4 \cdot 10^5, 6 \cdot 10^5]$

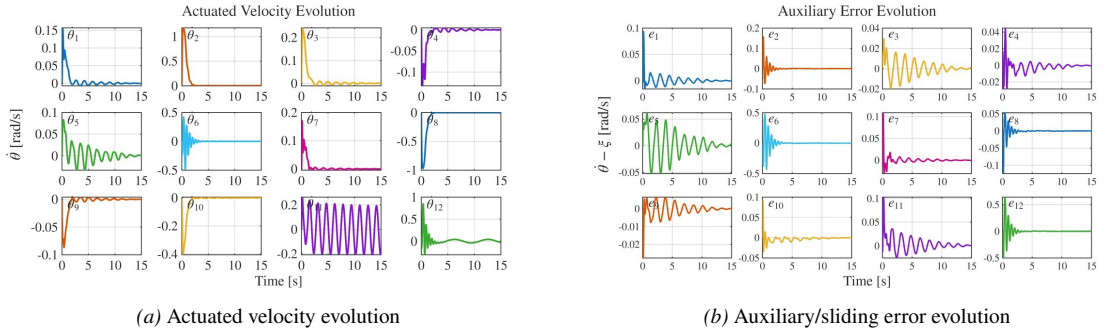


Figure 5.27: Velocity-level tracking and sliding-variable convergence under the fully-actuated adaptive controller in the nominal case DLO (DLO2).

As shown in Figure 5.25, both boundary poses – the base and the tip – are driven smoothly to their reference trajectories, with the position and orientation channels of the two ends converging over a comparable time horizon and without the persistent, higher-frequency residual that characterized the actuated-coordinate tracking in the underactuated cases of Sections 5.1.1–5.1.2. This is the expected signature of full actuation: because both ends are now directly commanded, the controller no longer has to shape the object’s equilibrium indirectly through a single boundary condition, and the coupling between the base and tip channels shows up only as a mild, well-damped transient on each pose component immediately after a reference change, rather than as a sustained oscillation. The velocity-level behaviour in Figures 5.27a–5.27b confirms this: essentially all actuated-velocity channels settle within the same short transient, and the sliding-variable convergence

$$\tilde{\xi} = q_a - \xi \rightarrow 0 \quad s = \dot{q} - \xi \rightarrow 0$$

is confirmed as in the previous scenarios, but now uniformly across channels rather than with one or two lagging components. The torque distribution in Figure 5.26a is correspondingly smooth, with the control effort shared between the two arms rather than concentrated on a single actuator.

The parameter estimates in Figure 5.26b converge toward the reference values with the same qualitative ordering seen throughout this chapter ($\hat{\rho}$ fastest, $\hat{\eta}$ slowest), but the gap between the

fast and slow estimates is visibly narrower than in the one-arm and fixed-joint cases: $\hat{E}$ settles noticeably sooner, and $\hat{\eta}$, while still the last to converge, shows less overshoot and a shorter transient than its one-arm counterpart. This is consistent with the richer excitation argument of Section 4.1.3: with two independently actuated boundary conditions exciting the object's internal strain more persistently than a single grasp point, the regressor entries associated with E and η are better excited, which directly translates into faster and cleaner adaptation of these two harder-to-identify parameters.

Comparison of Control Strategies The physical characteristics of the simulated DLOs remain those of Table 5.1. The comparison baselines are tuned as

$$\text{PID: } K_P = 45, K_I = 25, K_D = 5$$

$$\text{PD+: } K_P = 6, K_D = 2$$

Where the average nominal-case performance metrics are:

$$\text{Adaptive Control : } \bar{M}_s = 0.62\% \quad \bar{t}_s = 1.36 \text{ s}$$

$$\text{PID Control : } \bar{M}_s = 22.1\% \quad \bar{t}_s = 0.937 \text{ s}$$

$$\text{PD+ Control : } \bar{M}_s = 4.71\% \quad \bar{t}_s = 0.907 \text{ s}$$

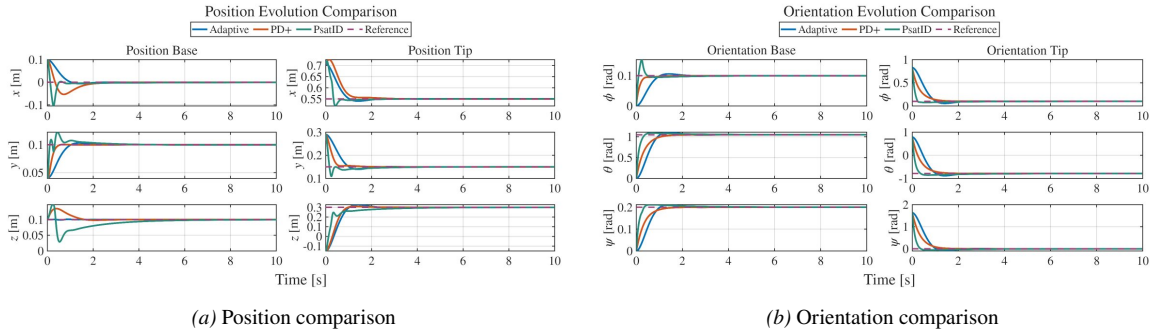


Figure 5.28: Pose tracking comparison for DLO2 considering the three different control actions– Adaptive (Blue), PD+cancellation (Orange), PsatID(Green), Reference(Purple).

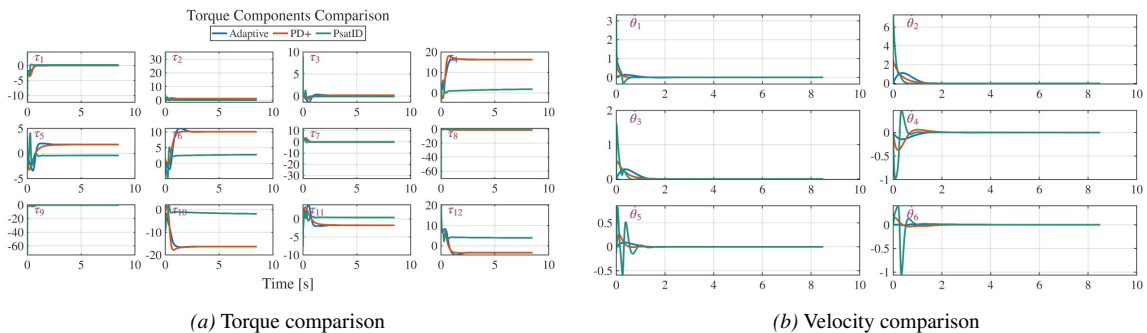


Figure 5.29: Control-effort and velocity comparison for DLO2 considering the three different control actions– Adaptive (Blue), PD+cancellation (Orange), PsatID(Green).

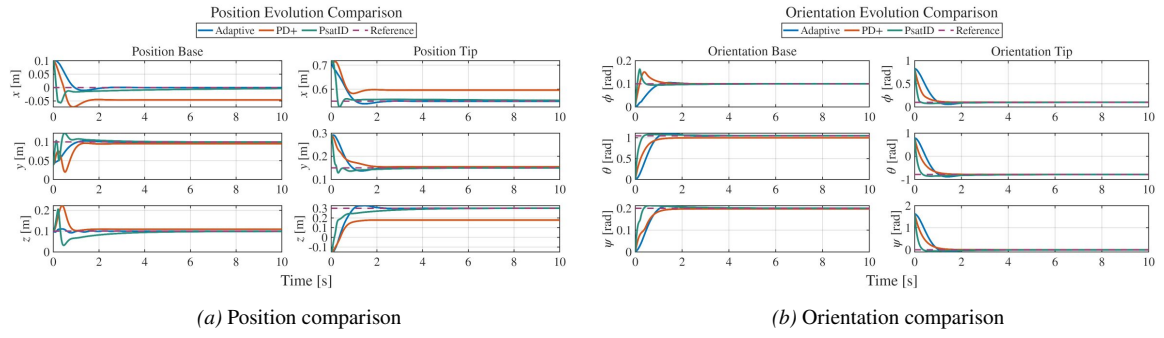


Figure 5.30: Pose tracking comparison for DLO3 considering the three different control actions– Adaptive (Blue), PD+cancellation (Orange), PsatID(Green), Reference(Purple).

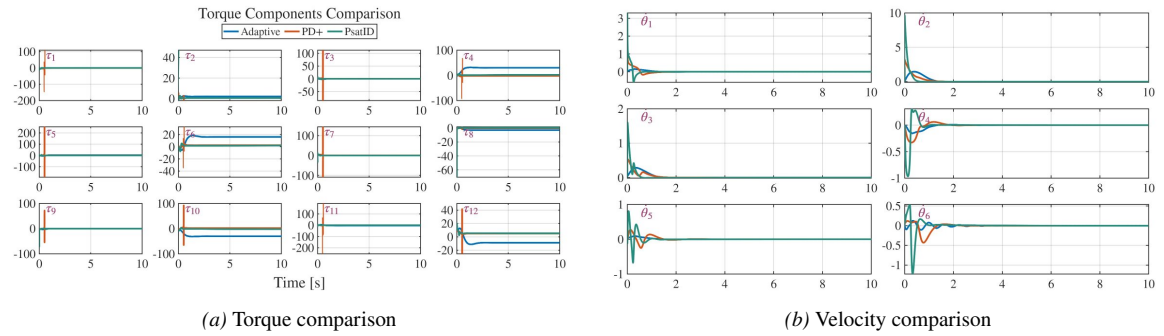


Figure 5.31: Control-effort and velocity comparison for DLO3 considering the three different control actions– Adaptive (Blue), PD+cancellation (Orange), PsatID(Green).

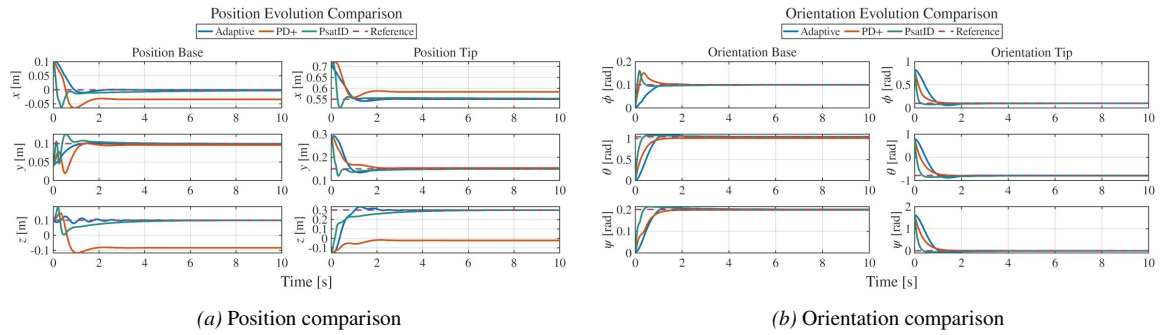


Figure 5.32: Pose tracking comparison for DLO4 considering the three different control actions– Adaptive (Blue), PD+cancellation (Orange), PsatID(Green), Reference(Purple).

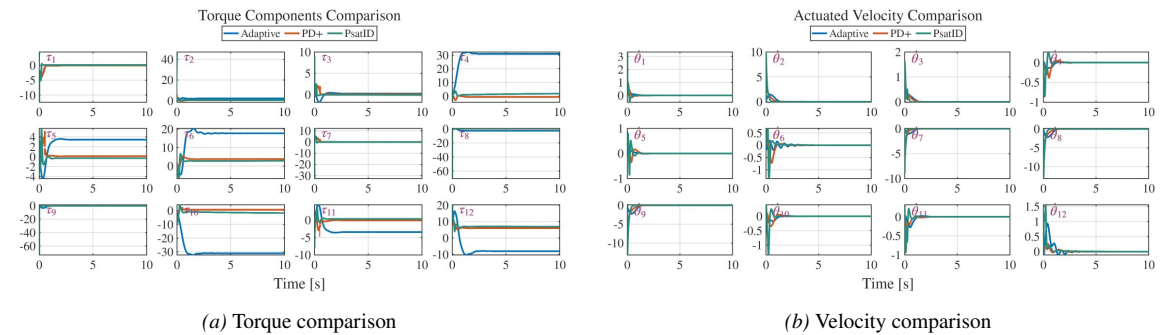


Figure 5.33: Control-effort and velocity comparison for DLO4 considering the three different control actions– Adaptive (Blue), PD+cancellation (Orange), PsatID(Green).

As in Figures 5.28a–5.32a and 5.28b–5.32b, the three controllers achieve comparable pose-tracking accuracy on DLO2, the nominal object used to tune PID and PD+, with the base and tip trajectories nearly overlapping across strategies. As with the previous two scenarios, this agreement degrades once the DLO’s physical parameters move away from the nominal case: for DLO3 and, more markedly, DLO4, the PID and PD+ pose trajectories in Figures 5.30a–5.32a show a larger overshoot and a slower return to the reference on both the base and tip channels, while the Adaptive controller’s trajectories change only slightly across the three objects. The corresponding torque and velocity comparisons (Figures 5.29a–5.33b) make the difference even clearer: several torque components for PID and PD+ grow substantially in amplitude and develop sharper transients from DLO2 to DLO4, whereas the Adaptive controller maintains a comparatively flat and bounded torque profile across all three objects, consistent with the order-of-magnitude gap already reported for the one-arm and fixed-joint scenarios. Interestingly, the gap between Adaptive and the two baselines opens more gradually here than in the underactuated cases, a plausible consequence of the richer, fully-actuated excitation making even the fixed-gain baselines comparatively more robust to parameter mismatch than they were when only one boundary condition could be commanded.

Adaptability Proceeding in the same order as for the two other simulation scenarios, we now show the ability of the adaptive controller to handle the four different objects DLO1–DLO4, all starting from an identical initial configuration.

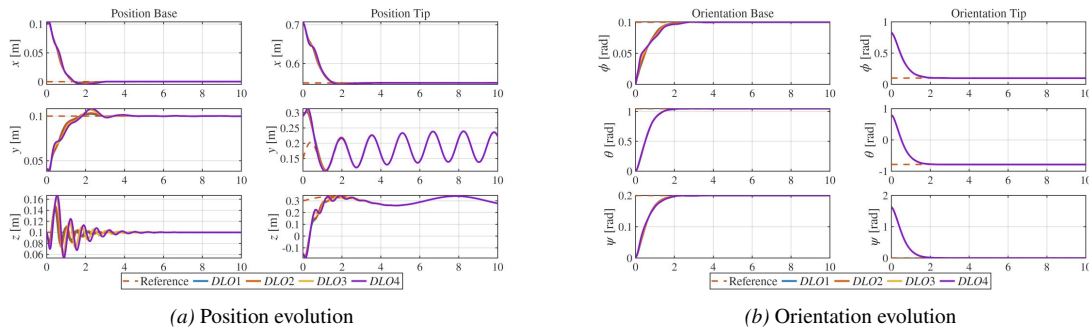


Figure 5.34: Adaptive controller performance across the four DLOs (DLO1–DLO4) considered for the dual-arm scenario. Panel (a) shows the position evolution of base and tip, while panel (b) shows the orientation evolution for each object.

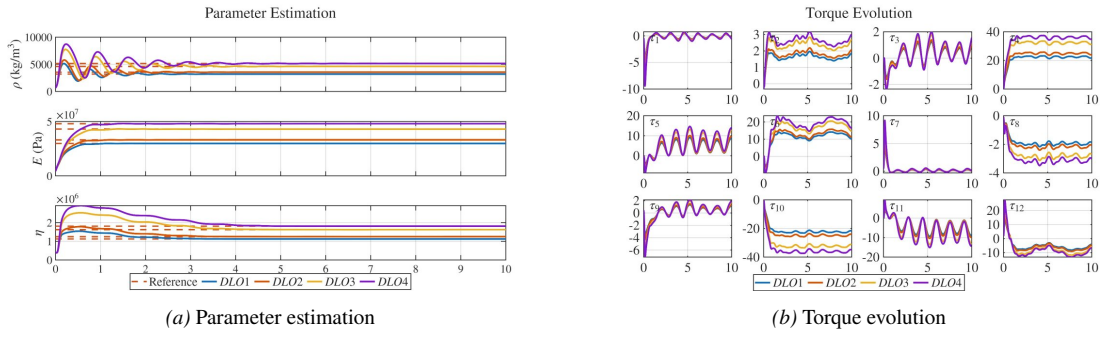


Figure 5.35: Parameters convergence and control effort across the four DLOs (DLO1–DLO4) considered for the dual-arm scenario. Panel (a) shows the parameters evolution, while panel (b) shows the corresponding control torque for each object.

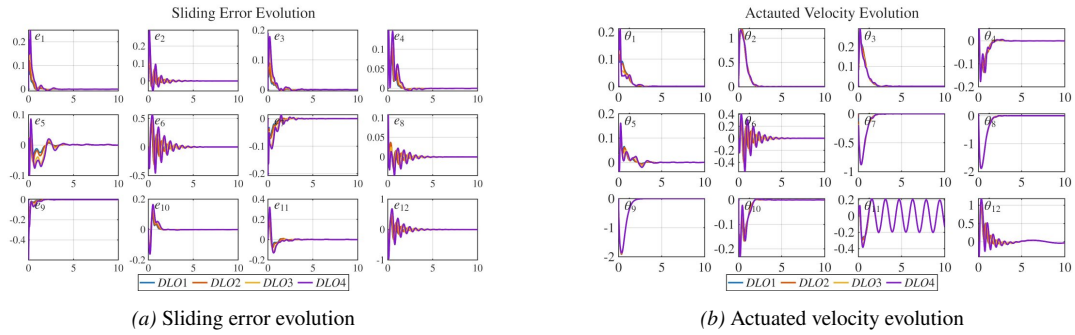


Figure 5.36: Sliding-errors convergence and velocities across the four DLOs (DLO1–DLO4) considered for the dual-arm scenario. Panel (a) shows the sliding errors evolution, while panel (b) shows the corresponding actuated velocities for each object

The base pose in Figure 5.34 is tracked consistently across all four objects: the base position and orientation channels converge to their reference within the first 2–3 s regardless of the object, with DLO4 – the heaviest and stiffest of the four – showing only a slightly slower initial rise before catching up to the other three. The tip channels tell a more nuanced story. The tip orientation and the tip x -position converge cleanly and, once again, essentially independently of the object. The tip y and z positions settle into their sinusoidal reference, with excellent results, only differentiated by small initial transient evolutions.

The parameter estimates in Figure 5.35a converge toward each object’s own reference values, confirming that the estimator correctly distinguishes between the four different DLOs rather than converging to a single average model. The convergence is fastest and cleanest for DLO1–DLO3, whose $\hat{\rho}$ and $\hat{E}$ settle within about 2–3 s with only minor ripple; DLO4, being the most extreme object in the set, instead exhibits a pronounced damped oscillation on both $\hat{\rho}$ and $\hat{E}$ before settling near its reference around $t \approx 7\text{--}8\text{ s}$. The damping estimate $\hat{\eta}$ is, consistently with every other scenario in this chapter, the slowest of the three: for DLO4 it overshoots early and then decreases monotonically for almost the entire simulation, only approaching its reference value toward the end of the horizon, while for DLO1–DLO3 it settles closer to (though still visibly biased with respect to) the reference within the same window. The torque evolution in Figure 5.35b mirrors this ordering: torque magnitude and steady-state offset scale directly with the object’s physical properties, with several components (e.g. τ_4 , τ_6 , τ_{10}) reaching amplitudes

for DLO4 that are two to three times larger than the corresponding value for DLO1, while all four objects share essentially the same qualitative torque profile.

Finally, the sliding-variable and actuated-velocity behaviour in Figure 5.36 shows that most of the twelve components converges to zero within about 2–5 s for all four objects, with DLO4 again producing the largest initial transient in almost every component – consistent with the fact that it requires the largest parameter correction. Due to the persistently-oscillating reference trajectory associated to the tip’s motion, a pair of components correctly settle to the sinusoidal value, while the rest settles to zero.

Under-Actuated Case

To complete and justify the use of the control law presented in Chap.4, this paragraph will showcase some of the most interesting results related to under-actuated control in bimanual manipulation of the DLO.

To achieve the underactuation in the bimanual setup, which has 12 DOF of actuation, we need to increase the degree of the Legendre polynomials. This results in the model’s ability to complete the task by performing different deformations and returning different information.

By comparing the performance obtained in the fully-actuated case and the under-actuated case while controlling the DLO’s extremities, we obtain the following.

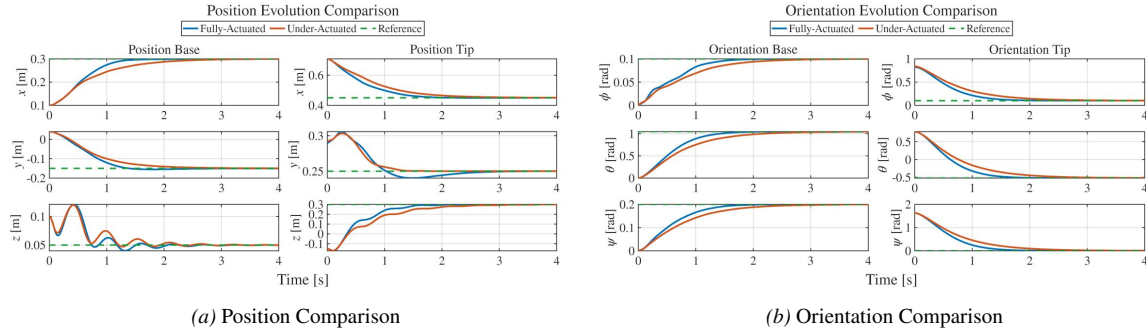


Figure 5.37: Pose evolution comparison considering the under-actuated case(Orange) and the fully-actuated case(blue) while completing a regulation task.

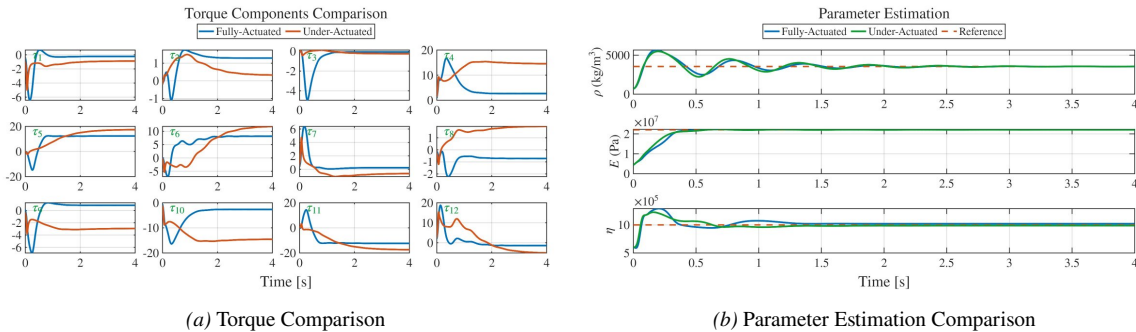


Figure 5.38: Comparison of the Torque evolution and the Parameter estimation considering the under-actuated case(Orange) and the fully-actuated case(blue) while completing a regulation task.

As it can be noted, generally the fully-actuated model is faster to complete the task; as mentioned before, this must be related to the fact that the deformations obtained are smaller in

the FA case as they follow a 2nd order Legendre polynomial instead of the 3rd order exploited by the UA model. However, the fact that the UA case has a greater number of generalized coordinates used in the model results in a larger amount of information reaching the parameter estimation evolution. This can be seen in Fig. 5.38b where the estimation of both E and η settles faster to the reference value. While, as per the construction of the model, ρ depends heavily on the free moving base, we cannot appreciate a faster evolution.

This result could motivate an interest in increasing the order of the model to achieve the best trade off between convergence in the task space and parameter estimation. To this point, we showcase the difference between a 2nd, 3rd and 5th order model.

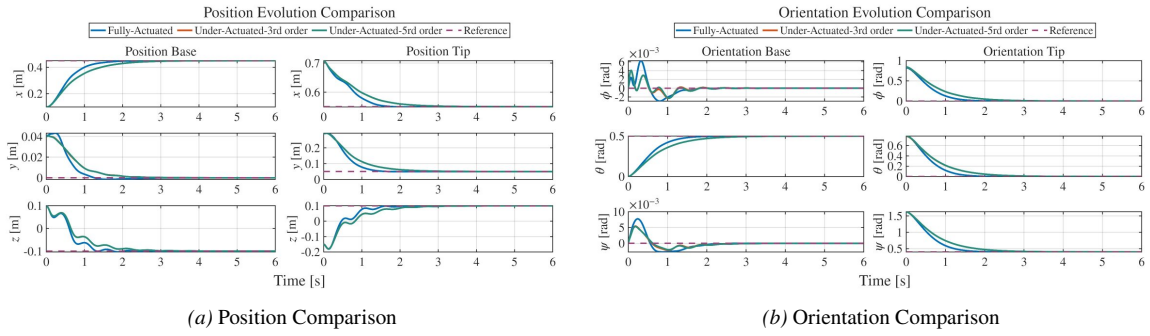


Figure 5.39: Pose comparison considering the fully-actuated case (blue), the under-actuated case 3rd order (Orange) and the under-actuated case 5th order (Green) while completing a regulation task.

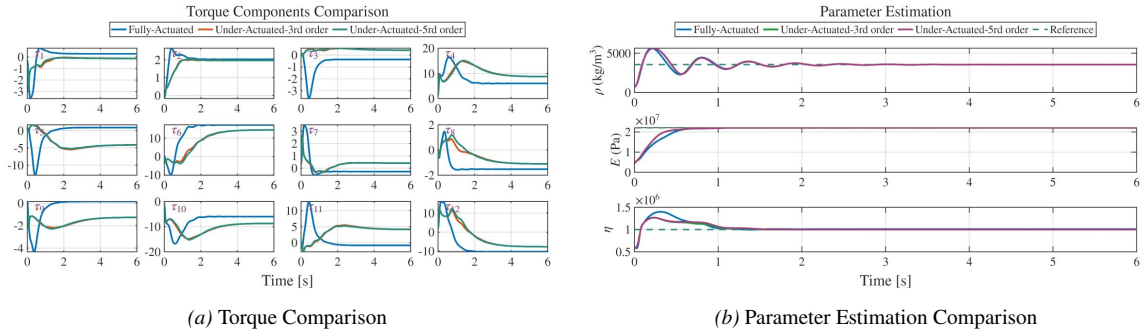


Figure 5.40: Comparison of the Torque evolution and the Parameter estimation considering the fully-actuated case (blue), the under-actuated case 3rd order (Orange) and the under-actuated case 5th order (Green) while completing a regulation task.

As it can be seen, the difference in the performance in the underactuated cases is imperceptible. Some slight differences can be noted in the torque produced, Fig. 5.40a. Instead, the figure that was key to this evaluation didn't produce the desired effect. Indeed in Figure 5.40b, the estimation does not improve significantly enough to justify an increase in the order of the model description, which also requires a bigger computational effort.

Shape Control

Lastly, exploiting the numerous actuation degrees of freedom and trying to complete a task useful in a real world implementation, we consider the control of the pose at different points from the

base and tip while keeping the actuation in these points. The strategy follows the minimization problem on the achievable configuration by considering the realizable deformations $\mathcal{R}$:

$$\min_{q \in \mathcal{R}} \|p(q) - p_d\|^2 \quad (5.10)$$

This minimization function can be used for the regulation of one or more points in the cable. The achieved results shown in what follows consider the pose regulation of two points in the cable.

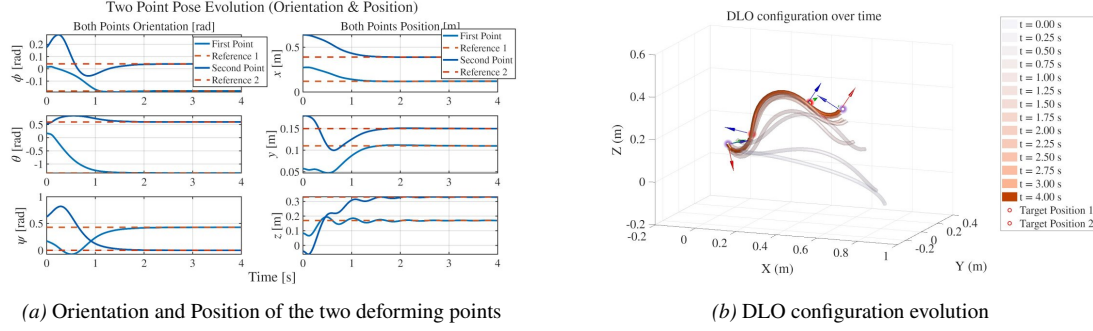


Figure 5.41: Actuated Coordinate associated to the deforming points and Configuration evolution of the nominal case DLO (DLO2) in the dual arm model

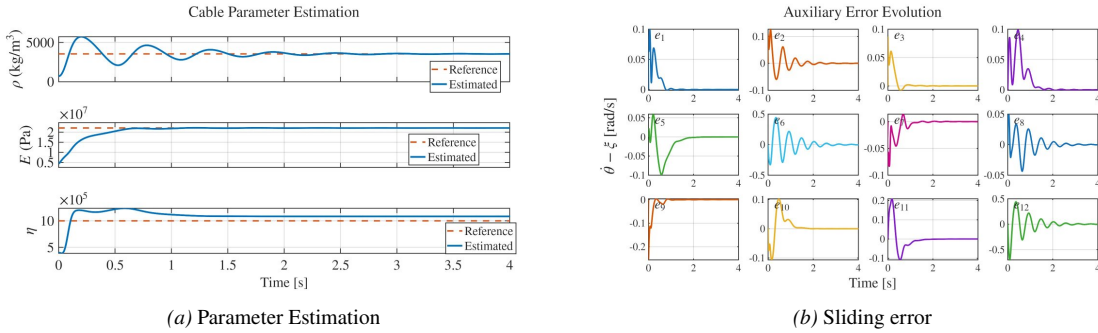


Figure 5.42: Parameter estimation and sliding error evolution in time related to the configuration control in the dual arm model. Panel (b) we see the known result of convergence to zero of the sliding error.

As depicted in Fig.5.41 the desired task was correctly achieved by the Adaptive control action. Even more, the problem allowed the perfect estimation of the mass density and stiffness of the DLO. However, the usual issue related to the limited excitation in the damping is present and it precludes from the perfect estimation of the elastic damping coefficient.

6

Experimental Validation

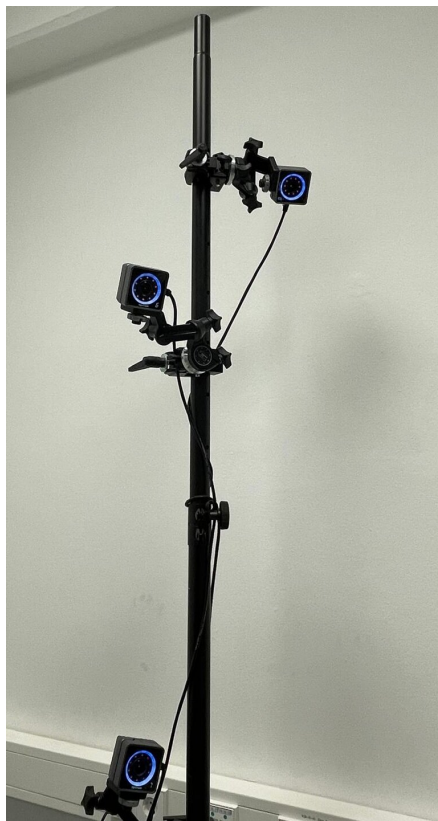
To complete the work developed in this thesis, experiments regarding single-arm manipulation were carried out. The choice to proceed with single-arm manipulation only is justified by two key motives: first, the limited time available for access to the robot, and second, the internal complexity of coupling a compliant system (the cable) with a rigid system (the Franka manipulator).

6.0.1 Experimental Setup

The setup is composed of the following elements:

1. **Manipulator:** The Franka Panda is a 7-DOF collaborative robotic arm developed by Franka Robotics (formerly Franka Emika). Each joint is equipped with torque sensors, enabling direct measurement and control of joint torques rather than relying solely on motor current estimation. The robot has a rated payload of 3 kg and a repeatability of ± 0.1 mm. It supports low-level, real-time control at up to 1 kHz via the `libfranka` C++ library and its ROS integration (`franka_ros`), which allow direct access to joint torque, velocity, and Cartesian impedance control interfaces. In this work, the robot was controlled using the Cartesian impedance interface.
2. **Deformable Linear Object (DLO):** A cable was used, with the physical characteristics associated with DLO2 in Tab. 5.1, namely length: 0.77 m, radius: 0.009 m, bending: 3546.6 kg/m^3 , and stiffness: $2.2 \cdot 10^6$. The cable was fixed at one end to the manipulator's end-effector, while the other end was free, reflecting the single-arm manipulation scenario described above.

3. **Tracking system:** OptiTrack was used in this work; it is a high-precision, marker-based optical motion capture system that uses infrared cameras to triangulate the 3D position of reflective markers in real time, typically achieving sub-millimeter positional accuracy. The setup used in this work consisted of 6 cameras operating at 240 fps. 4 reflective markers were placed along the cable at regular intervals of 18 cm, allowing the reconstruction of its centerline via the open-loop inverse kinematics algorithm and providing ground-truth data for validation of the model. Also, 4 reflective markers on the end effector to compute the base's pose of the DLO with respect to the end effector pose.



(a) Motion capture system



(b) 1. shows the Franka Panda robot; 2. shows the DLO considered

Figure 6.1: The experimental setup used in the validation cases proposed.

Each component has a specific role in the setup. The manipulator is the element that performs the physical action; the high-level controller acts on it to translate the desired task-space action into robot-space commands. The DLO is the principal object of study, as it is the element being acted upon: its motion is driven by the manipulator, but knowledge of its dynamic evolution is required to compute the low-level control action needed to complete the task. Finally, the tracking system is essential to the experiments, as OptiTrack is used to measure the cable's evolution in space; combined with an ad-hoc open-loop inverse kinematics algorithm, it reconstructs the cable's kinematics, which is necessary to obtain its shape and dynamics during the experiments.

6.0.2 Control Law

The control law described in Sec. 4 was adjusted to use an impedance control strategy, rather than directly using force and torque to control position and force. This strategy is typically used when the manipulator must interact with the environment. To give the full picture, this section briefly explains the dynamics of the whole system, described as

$$M(\mathbf{q})\ddot{\mathbf{q}} + c(\mathbf{q}, \dot{\mathbf{q}}) + g(\mathbf{q}) = \tau \quad (6.1)$$

where M, c, g are, respectively, the inertial, Coriolis/centrifugal, and gravitational components of the system; the generated torque is related to the wrench components via $\tau = J^T(q) \begin{pmatrix} f \\ \mu \end{pmatrix}$.

Since the experiments were carried out using two different control strategies – PD and PD + cancellation – a unified framework was developed to express all control actions consistently:

$$\tau = J^T(q)u = J^T(q) \left(K_P(\theta_a - \theta_a^d) + K_D(\dot{\theta}_a - \dot{\theta}_a^d) + u_c \right) \quad (6.2)$$

where u_c depends on the controller used:

$$u_c = \begin{cases} 0 & \text{PD} \\ J_b^{-1} A d_{g_b}^{*-1} (K_\theta + Q_\theta) & \text{PD + cancellation} \\ J_b^{-1} A d_{g_b}^{*-1} (Y\hat{\pi} + G_k \Lambda_2 \left(\int (\theta_a - \theta_a^d) dt \right)) & \text{Adaptive control} \end{cases}$$

6.0.3 Results

The section will be divided considering the different experiments developed; namely, we will first report on a regulation task of the actuated coordinate by generating a multi step trajectory. Then, we will discuss a slight modification that considers a multi step task which also includes large rotations. Finally, we will address trajectory tracking. However, the end goal is the control of the unactuated components; in this case, the pose in space of the free tip of the DLO.

Multi-step regulation The goal of this task is to achieve the regulation at different setpoints. The task was analyzed at different initial condition of the parameters value, to showcase the robustness of completing the with the adaptive controller.

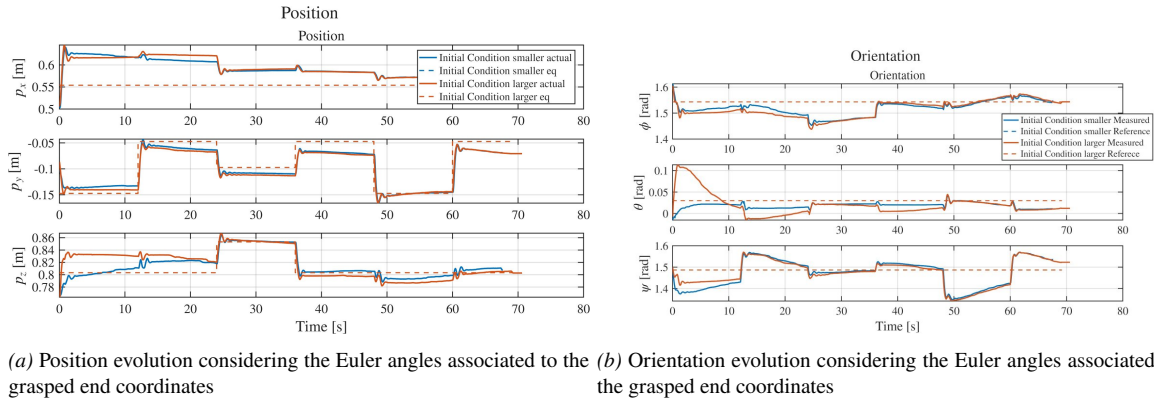


Figure 6.2: Actuated coordinate performance considering different initial conditions of the parameter values- Blue for the parameter values with nominal ratio 0.1, Orange for the parameter values with nominal ratio 2.0

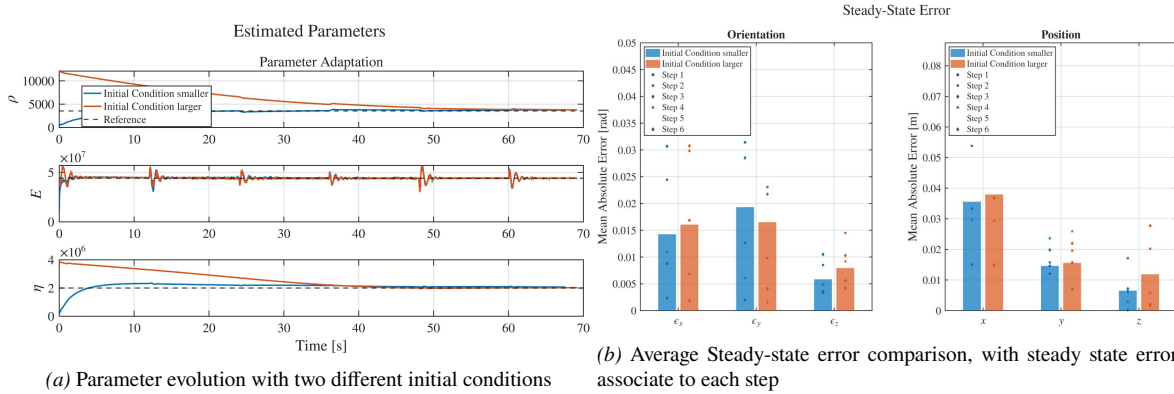


Figure 6.3: Parameter evolution and steady state error behaviour analysis- Blue for the parameter values with nominal ratio 0.1, Orange for the parameter values with nominal ratio 2.0

The plots showcase the desired property of adaptive control, indeed, the differences in the two control performances are quite small, as per Fig.6.3b. Both control actions achieve parameter estimation to the nominal value of the cable characteristic. As it will be analyzed with more depth later, we can see that the achieved tracking results are off from the desired regulation task. This could be related to the deconstruction of the control law for the experimental implementation; indeed, the search for a better set of gains to complete the task has resulted in more complicated work compared to what was obtained in the simulation. The principal goal of this paragraph is to demonstrate the comparability of this adaptive control with other control strategies.

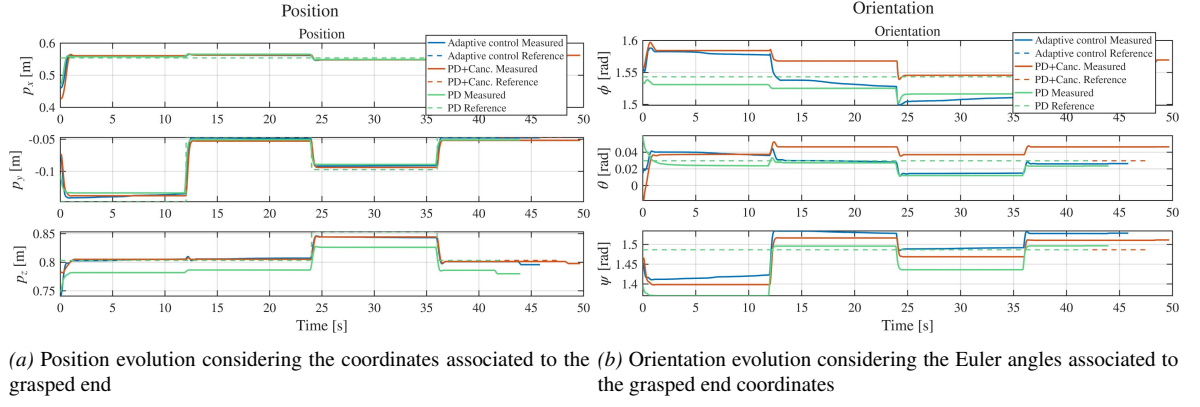


Figure 6.4: Actuated coordinate performance considering different control strategies-Blue for the adaptive control action, Orange for the PD+ Cancellation method, and Green for the PD control action

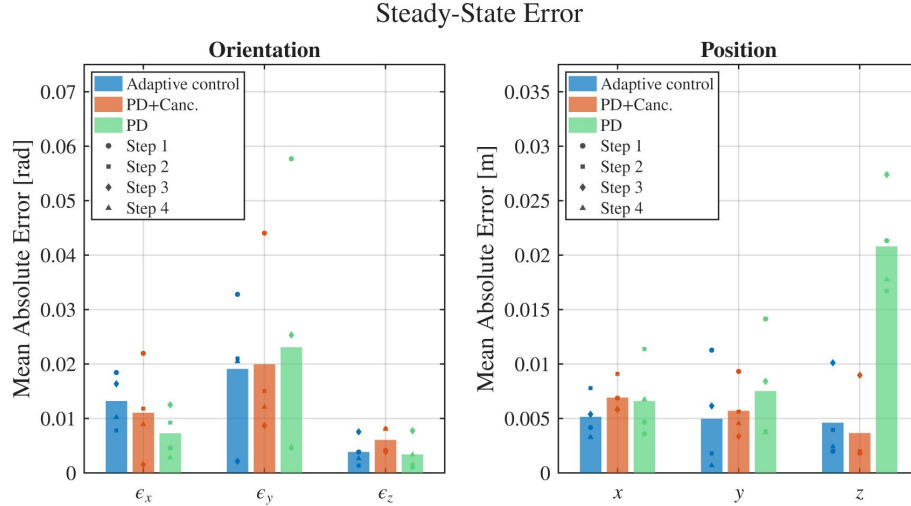


Figure 6.5: Average Steady-state error comparison, with steady state error associate to each step of the three different control strategies

As previously done in the comparison plots, the most revealing figure is associated with the average steady state error, in Fig.6.5. It can be seen that the performance in the different actuated coordinates is relatively comparable; however, the large difference in performance is achieved in the control of the z coordinates, which is directly affected by gravity. This is correctly compensated for by the cancellation component of the second control action presented, and with slightly worse performance, it is compensated for by the adaptive control action. Lastly, to complete the task of justifying the choice of this control strategies, we illustrate in the next figure 6.6 the importance of having an estimation evolution. Indeed, as it was mentioned at the beginning of this work, in the general case, the information available of the DLO are quite limited and most of all time consuming to fetch.

Steady-State Error

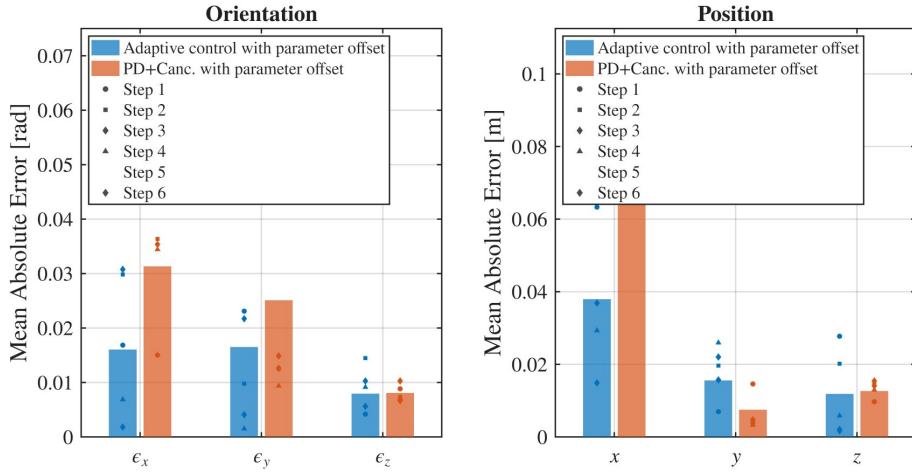
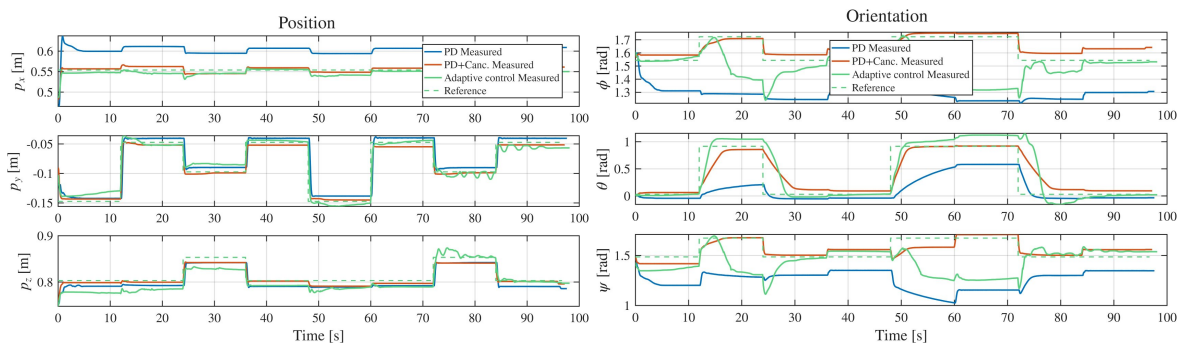


Figure 6.6: Average steady state error comparison between an wrongly model DLO, namely, an offset in the characteristics of the initial condition of the physical parameters

Figure 6.6 shows the effect of a wrongly known system, and the strength of the adaptive control action to reduce the steady-state error generated in the multi-steps regulation task. This is an awaited result as the mismodeled cancellation acts as an additional error to the desired task; instead, the correction imposed by the adaptive control strategy acts against the error, due to its ability of stirring the mismodeled parameters to their correct values.

Multi-step regulation with the addition of the rotations A different kind of control action task results when we consider also the additional motion of rotation while translating.



(a) Position evolution considering the coordinates associated to the grasped end (b) Orientation evolution considering the Euler angles associated to the grasped end coordinates

Figure 6.7: Actuated coordinate performance considering different control strategies-Blue for the adaptive control action, Orange for the PD+ Cancellation method, and Green for the PD control action

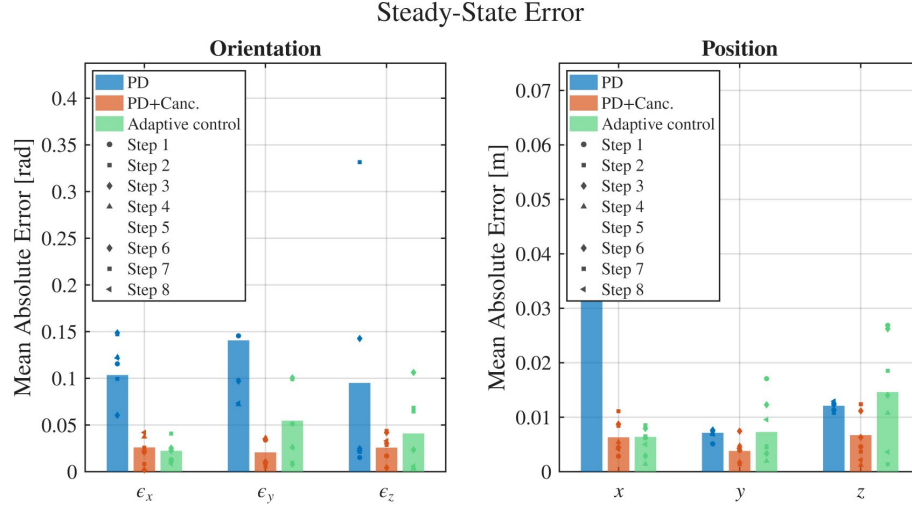


Figure 6.8: Average Steady-state error comparison, with steady state error associate to each step of the three different control strategies

The performance depicted above represents again the expected result, the overall best performance belongs to the PD+Cancellation, while the Adaptive control has an average worse performance that is however related to the initial behaviour. Indeed, by following the symbols associated to the steady state error at each step: it can be seen that in the first three steps the steady state error generated is larger than the average value represented by the column of the histogram. Instead, in the final steps when the adaptive controller has learned the model, much smaller steady state errors can be noticed. Lastly, other than the big difference in controlling the rotations variable, we see again the effect of gravity that this time affects the x-coordinate due to the fact that now the rotations influences the momentum of the whole system, resulting in a larger error production in this coordinate, in particular.

Trajectory tracking The last task covered is tracking of a sinusoidal trajectory, which is done to generate the maximum excitation to achieve the parameter estimation. As it was done in the Simulation chapter 5, this paragraph begins by comparing the performance of different mismodel systems.

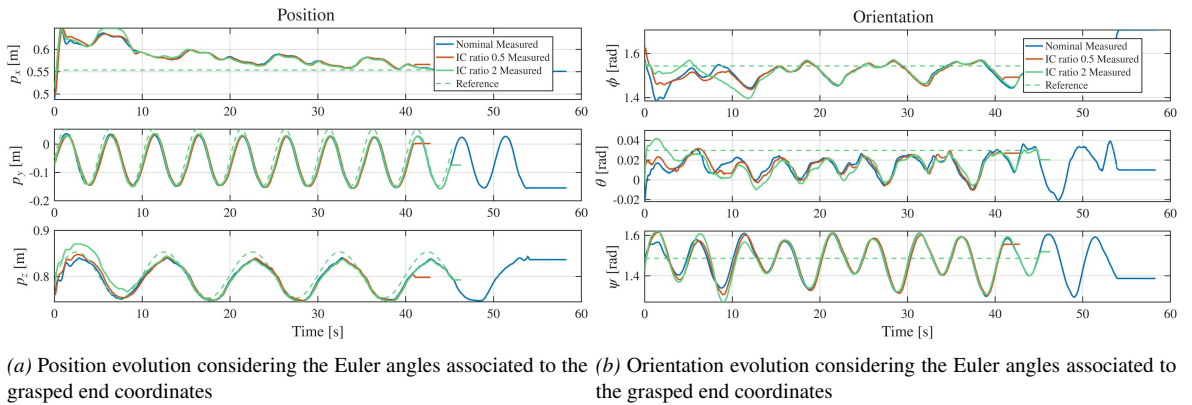


Figure 6.9: Actuated coordinate performance considering different initial conditions of the physical parameter values, while completing the trajectory tracking task.

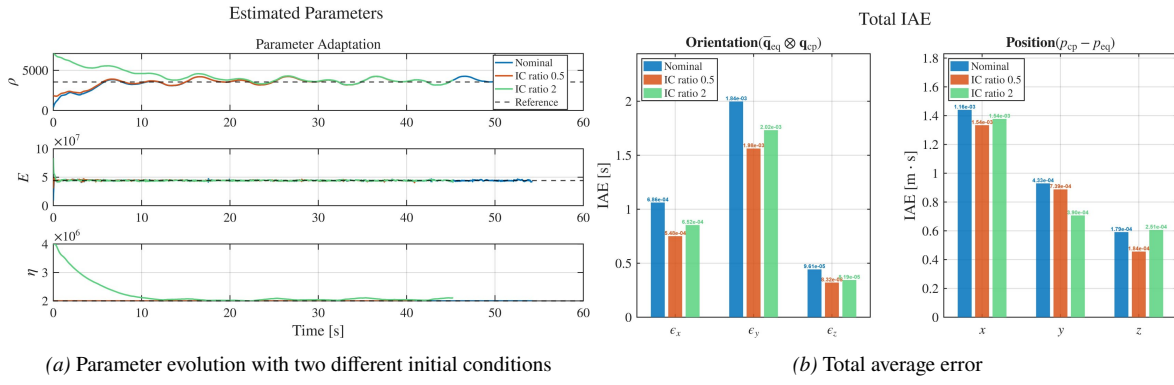


Figure 6.10: Parameter estimation evolution and total error generated considering different initial conditions of the physical parameter values, while completing the trajectory tracking task.

As it can be seen, the robustness of the adaptive control is once again showcased, where the overall best performance is obtained considering the nominal value of the parameter estimation, while the other two initial conditions do not produce performance that are so different. The estimation is achieved in all three parameters at different rates.

The usual comparison in the control strategies follows.

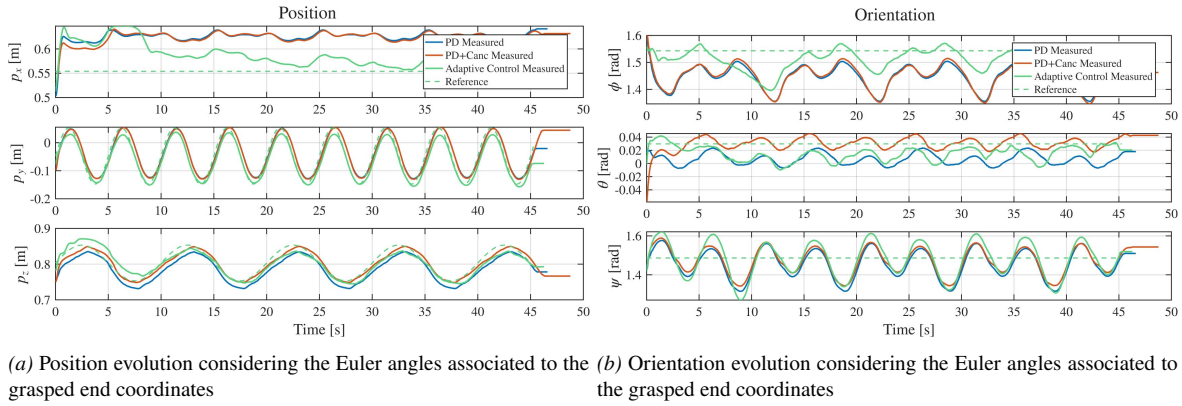


Figure 6.11: Actuated coordinate performance considering different control strategies while completing the trajectory tracking task.

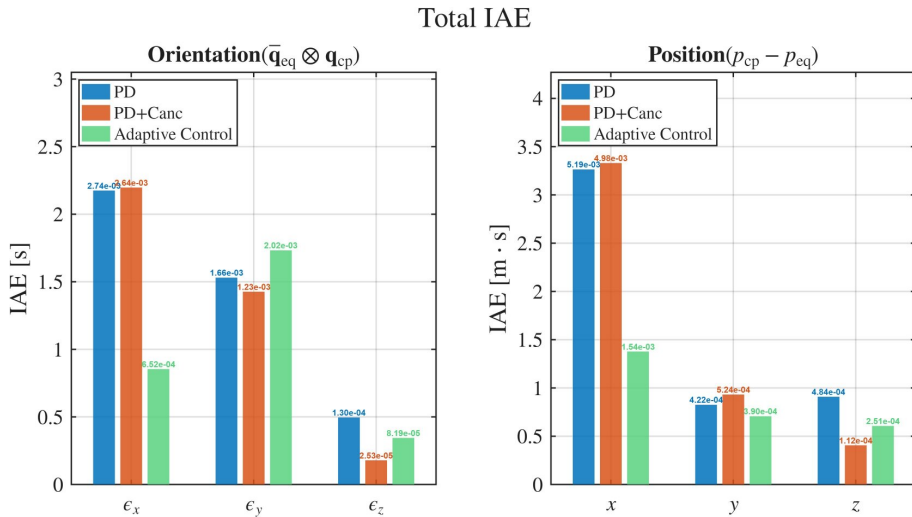


Figure 6.12: Total error generated by each control strategy while completing the trajectory tracking task.

Here, more than in the previous cases, we can see the ability of the adaptive controller to reduce the error and achieve the best performance among the different strategies. The result in Figure 6.12 illustrates the total error generated in the full run by each strategy, as a standard way of fairly comparing the control performances.

The end goal was to achieve the inverse kinematics; thus, the regulation of the unactuated component was also possible. Due to the lack of time, few experiments were conducted. However, there are promising results on the cable used up to this point. The result can be observed in the following plot, which shows the evolution over time of the unactuated pose.

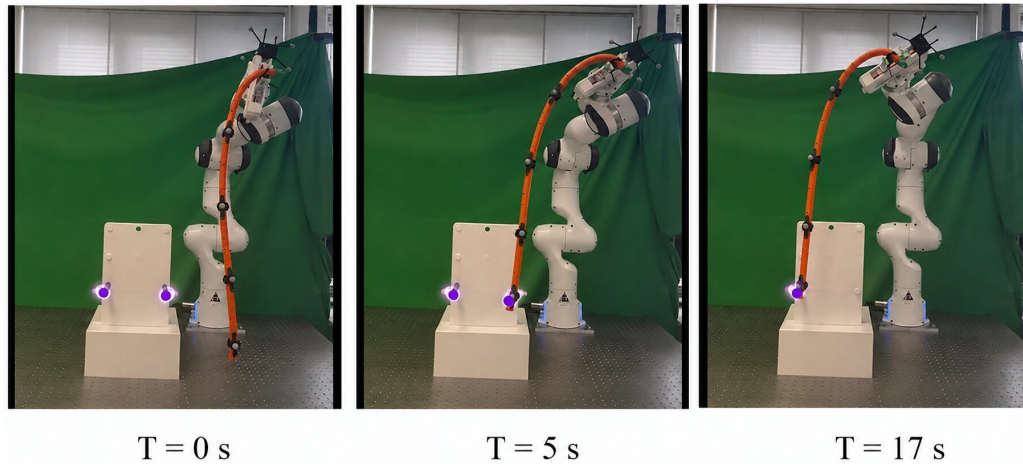


Figure 6.13: Unactuated component control evolution in time, based in the inverse kinematics algorithm

7 Conclusion

Considering the state of the art in the manipulation of deformable linear objects, a gap was found regarding the implementation of model based strategies that allowed for compensation of uncertainties and retrieval of key physical information. This research filled this gap by exploiting a realistic modeling technique in the GVS approach while attempting to control and compensate using an underactuated adaptive control strategy. The use of a model based technique gave an additional robustness and light computational effort to the standard category of adaptive control. In Chap.5-6, different results were showcased, from the ability to operate on different DLOs to achieve control actions comparable in both control effort and performance with different commonly used control strategies, as well as the ability to execute model dependent tasks based on the information returned to the control law. This was achieved for 3 different models, each with its own specifics and useful characteristics.

This thesis broadens the scope of DLO research and its real-world applications by adopting a framework that differs from those typically considered. In particular, the proposed manipulation task enables experiments with different materials while maintaining the same standard work frame, thereby providing valuable insights into the approach.

While the initial results in the experimental setup were promising, the same experiments should be performed in the bimanual setup. This system could provide a greater amount of real-world application with harder tasks to complete. In fact, future research should focus on greater experimental validation and the introduction of grasping by exploiting the manipulation of deformable objects. Alternatively, an even greater challenge could be the use of dynamic behavior to achieve complex tasks like passing cables or manual abilities such as knotting.

8 Appendices

8.1 GVS Model derivation

8.1.1 Key structural elements

This appendix gives the complete derivation of the Geometric Variable Strain (GVS) kinematics and dynamics summarized in Section 3.1, following Mathew and Feliu-Talegon [13]. It is organized in the same three blocks as the main chapter – strain parametrization, recursive kinematics, and Lagrangian dynamics.

Strain parametrisation and kinematic preliminaries

Each link of the kinematic chain – rigid or soft – is described along a curvilinear abscissa X_i by a pose curve

$$g(\cdot) : X_i \in [0, L_i] \rightarrow g(X_i) = \begin{bmatrix} \mathbf{R}_i & \mathbf{r}_i \\ \mathbf{0} & 1 \end{bmatrix} \in \text{SE}(3), \quad (8.1)$$

where X_i is the position along the backbone of link i and g fully describes the pose of the local frame attached at X_i . For a *rigid* link, the same description holds with $X_i \in [0, 1]$ reinterpreted as an evolution variable (0 for the initial pose, 1 for the final pose) rather than a physical length.

The spatial and time derivatives of g define, respectively, the strain twist $\hat{\xi}_i$ and the (right,

body-frame) velocity twist $\hat{\eta}_i^r$:

$$g'(X_i) = \frac{\partial g(X_i)}{\partial X_i} = g_i \hat{\xi}_i = g_i \begin{bmatrix} \tilde{k}_i & p_i \\ \mathbf{0} & 0 \end{bmatrix}, \quad (8.2)$$

$$\dot{g}(X_i) = \frac{\partial g(X_i)}{\partial t} = g_i \hat{\eta}_i^r = g_i \begin{bmatrix} \tilde{\omega}_i^r & v_i^r \\ \mathbf{0} & 0 \end{bmatrix}. \quad (8.3)$$

Forward kinematics requires the spatial integration of Eq. (8.2); the two cases (rigid, soft) are treated separately below.

Rigid integration. For a rigid joint, $\hat{\xi}_i$ is constant along X_i (it is simply the joint twist), so the integration is exact:

$$g_i(1) = \exp(\hat{\xi}_i). \quad (8.4)$$

Soft integration. For a soft link, $\hat{\xi}(X_i)$ varies continuously, and the integral

$$g(X_i) = \exp\left(\int_0^{X_i} \hat{\xi}(s) ds\right) = \exp(\hat{\Omega}(X_i)) \quad (8.5)$$

has no closed form in general, since $\hat{\Omega}$ is a Lie-algebra valued integral series of nested Lie brackets of $\hat{\xi}(X_i)$ (the Magnus series). The series is truncated to the fourth order term (Zanna quadrature), which is accurate provided the integration step h_j is kept small; the rod is therefore subdivided into short intervals, and the pose is propagated recursively:

$$g(X_i + h_j) = g(X_i) \exp(\hat{\Omega}(h_j)), \quad \hat{\Omega}(h_j) = \frac{h_j}{2} (\hat{\xi}_1 + \hat{\xi}_2) + \frac{\sqrt{3} h_j^2}{12} \text{ad}_{\hat{\xi}_1} \hat{\xi}_2, \quad (8.6)$$

with $\hat{\xi}_l = \hat{\xi}\left(X_i + \frac{h_j}{2} \mp \frac{\sqrt{3}}{6} h_j\right)$, $l = 1, 2$, the two Zanna collocation points.

The corresponding map $\exp(\hat{\Omega}) \in \text{SE}(3)$ admits the closed-form Rodrigues-like expression

$$\exp(\hat{\Omega}) = I_4 + \hat{\Omega} + \frac{1 - \cos \|k\|}{\|k\|^2} \hat{\Omega}^2 + \frac{\|k\| - \sin \|k\|}{\|k\|^3} \hat{\Omega}^3, \quad (8.7)$$

where $\|k\|$ corresponds to the magnitude of the angular strain that is computed via Magnus expansion $\hat{\Omega}$.

8.1.2 Recursive kinematics: from strain to velocity and acceleration

The pose recursion of the previous section gives $g(X)$. This section derives the corresponding recursion for the velocity twist η and acceleration twist $\dot{\eta}$, culminating in the Jacobian relation Eq. (3.5) used throughout the main chapter.

Relating the strain twist to the velocity twist

Differentiating $g' = g \dot{\xi}$ with respect to time and $\dot{g} = g \dot{\eta}^r$ with respect to X , and equating mixed partials gives

$$\dot{\eta}^{r'} = \dot{\xi} - \text{ad}_{\xi} \eta^r, \quad \dot{\eta}^{r'} = \ddot{\xi} - \text{ad}_{\xi} \eta^r - \text{ad}_{\xi} \dot{\eta}^r. \quad (8.8)$$

where the Lie-Algebra operators are:

$$\text{ad}_{\xi}(X) = \begin{bmatrix} \tilde{k} & \mathbf{0}_{3 \times 3} \\ \tilde{p} & \tilde{k} \end{bmatrix} \in \mathbb{R}^{6 \times 6}, \quad \text{ad}_{\xi}(X) = \begin{bmatrix} \tilde{\omega} & \mathbf{0}_{3 \times 3} \\ \tilde{v} & \tilde{\omega} \end{bmatrix} \in \mathbb{R}^{6 \times 6}, \quad (8.9)$$

Using the identity between the Adjoint and adjoint maps, $\text{Ad}'_g = \text{Ad}_g \text{ad}_{\xi}$ and $\dot{\text{Ad}}_g = \text{Ad}_g \text{ad}_{\eta^r}$, and integrating Eq. (8.8) spatially via the Leibniz rule, the $\text{ad}_{\xi} \eta^r$ term cancels against the Ad'_g contribution, leaving the compact result

$$\eta^r(X_i) = \text{Ad}_g^{-1} \int_0^{X_i} \text{Ad}_g \dot{\xi} ds, \quad (8.10)$$

$$\dot{\eta}^r(X_i) = \text{Ad}_g^{-1} \int_0^{X_i} (\text{Ad}_g \ddot{\xi} + \dot{\text{Ad}}_g \dot{\xi}) ds. \quad (8.11)$$

where the Adjoint map of SE(3) was exploited :

$$\text{Ad}_g(X) = \begin{bmatrix} R & \mathbf{0}_{3 \times 3} \\ \tilde{r}R & R \end{bmatrix} \in \mathbb{R}^{6 \times 6}. \quad (8.12)$$

with R being the rotation matrix associated to the pose curve and $\tilde{r}$ the SE(3) transformation of the position vector r .

Rigid and soft cases

Specializing Eqs. (8.10)–(8.11) to each link type requires the *tangent operator* of the exponential map,

$$T(\xi) = \int_0^1 \exp(s \text{ad}_{\xi}) ds, \quad (8.13)$$

so that:

$$\textbf{Rigid case:} \quad \eta_i(1) = \text{Ad}_{\exp(\hat{\xi}_i)}^{-1} T(\xi_i) \dot{\xi}_i, \quad \dot{\eta}_i(1) = \text{Ad}_{\exp(\hat{\xi}_i)}^{-1} (\dot{T}(\xi_i) \dot{\xi}_i + T(\xi_i) \ddot{\xi}_i), \quad (8.14)$$

$$\textbf{Soft case:} \quad \eta^r(h_j) = \text{Ad}_{\exp(\hat{\Omega})}^{-1} T(\Omega) \dot{\Omega}(h_j), \quad \dot{\eta}_i(1) = \text{Ad}_{\exp(\hat{\Omega})}^{-1} (\dot{T}(\Omega) \dot{\Omega}(h_j) + T(\Omega) \ddot{\Omega}). \quad (8.15)$$

8.1.3 Recovering absolute velocity and acceleration

The above quantities are velocity and acceleration twists relative to the previous body-frame, to recover the absolute values of this twists a composition across integration steps must be used:

$$\eta(X_i + h) = \text{Ad}_{\exp(\hat{\Omega})}^{-1} \eta(X_i) + \eta_i^r(h), \quad (8.16)$$

$$\dot{\eta}(X_i + h) = \text{Ad}_{\exp(\hat{\Omega})}^{-1} \left(\dot{\eta}(X_i) + \text{ad}_{\eta(X_i)} \text{Ad}_{\exp(\hat{\Omega})} \eta_i^r(h) \right) + \dot{\eta}_i^r(h). \quad (8.17)$$

8.1.4 Discretized strain and the Jacobian recursion

The continuous strain field is approximated in finite dimensions as

$$\xi(X) = \Phi_\xi(X) q + \xi^*(X), \quad (3.3)$$

which, since Φ_ξ is independent of q and t , gives $\dot{\xi} = \Phi_\xi \dot{q}$ and $\ddot{\xi} = \Phi_\xi \ddot{q}$. Substituting into Eqs. (8.10)–(8.11) and grouping terms in $\dot{q}$, $\ddot{q}$ recovers the familiar robotics-like Jacobian form:

$$\eta^r = \text{Ad}_g^{-1} \int_0^{X_i} \text{Ad}_g \Phi_\xi(s) ds \dot{q} = J_i \dot{q}, \quad (8.18)$$

$$\dot{\eta}^r = \text{Ad}_g^{-1} \int_0^{X_i} \dot{\text{Ad}}_g \Phi_\xi(s) ds \dot{q} + \text{Ad}_g^{-1} \int_0^{X_i} \text{Ad}_g \Phi_\xi(s) ds \ddot{q}. \quad (8.19)$$

For the discretized Magnus expansion, the same substitution gives

$$\Omega(X)|_{\text{const.}} = X_i \xi = \Phi_\Omega q + X_i \xi^*, \quad (8.20)$$

$$\dot{\Omega}(h_k) = \left[\frac{h_k}{2} (\Phi_{\xi_1^k} + \Phi_{\xi_2^k}) + \frac{\sqrt{3}}{12} h_k^2 (\text{ad}_{\xi_1^k} \Phi_{\xi_2^k} - \text{ad}_{\xi_2^k} \Phi_{\xi_1^k}) \right] \dot{q} \triangleq \mathbf{B}_\Omega(h_k) \dot{q}, \quad (8.21)$$

so that, combining Eqs. (8.14)–(8.15) with the discretized Magnus expansion, the discretized velocity and acceleration twists become:

$$\eta^r = \text{Ad}_{\exp(\hat{\Omega}(h_k))}^{-1} T(\Omega(h_k)) \mathbf{B}_\Omega(h_k) \dot{q} = J_i \dot{q}, \quad (8.22)$$

$$\dot{\eta}^r = \text{Ad}_{\exp(\hat{\Omega}(h_k))}^{-1} \left(\dot{T}(\Omega(h_k)) \mathbf{B}_\Omega(h_k) \dot{q} + T(\Omega(h_k)) \mathbf{B}_\Omega(h_k) \ddot{q} \right), \quad (8.23)$$

and the absolute recursion of Eqs. (8.16)–(8.17) becomes exactly the Jacobian update used in the main chapter (Eq. (3.5) and, in the model-specific form, Eqs. (3.18) and (3.45)):

$$\eta_i(X_i + h) = J(X_i) \dot{q}, \quad \dot{\eta}_i(X_i + h) = J(X_i) \ddot{q} + \dot{J}(X_i) \dot{q}. \quad (3.5 \text{ revisited})$$

8.1.5 Lagrangian dynamics

With the Jacobian relation established, the generalized dynamics follow from projecting the free dynamics of each link onto the generalized coordinates via D'Alembert's principle.

Free dynamics – rigid link.

$$\mathcal{M}_i \dot{\eta}_i + \text{ad}_{\eta_i}^* \mathcal{M}_i \eta_i = \mathcal{F}_{J_i} - \text{Ad}_{g_{ij}g_j}^* \mathcal{F}_{J_j} + \mathcal{F}_{e_i}, \quad (8.24)$$

where $\mathcal{M}_i$ is the (constant) rigid-body inertia matrix, $\mathcal{F}_{J_i}$ is the wrench transmitted across the joint connecting link i to its parent, and $\mathcal{F}_{e_i}$ is any concentrated external load on link i .

Where the coadjoint operator associated to the velocity twist has the following formulation that depends on the linear and angular velocities:

$$\text{ad}_\eta(X) = \begin{bmatrix} \tilde{\omega} & \mathbf{0}_{3 \times 3} \\ \tilde{v} & \tilde{\omega} \end{bmatrix} \in \mathbb{R}^{6 \times 6} \quad \text{ad}_\eta^*(X) = \begin{bmatrix} -\tilde{\omega} & -\tilde{v} \\ \mathbf{0}_{3 \times 3} & \tilde{\omega} \end{bmatrix} \in \mathbb{R}^{6 \times 6}. \quad (8.25)$$

Free dynamics – soft link.

$$\begin{cases} \bar{\mathcal{M}}_i \dot{\eta}_i + \text{ad}_{\eta_i}^* \bar{\mathcal{M}}_i \eta_i = (\mathcal{F}_{i_i} - \mathcal{F}_{a_i})' + \text{ad}_{\xi_i}^* (\mathcal{F}_{i_i} - \mathcal{F}_{a_i}) + \bar{\mathcal{F}}_{e_i}, \\ (\mathcal{F}_{i_i} - \mathcal{F}_{a_i})(0) = -\mathcal{F}_{J_i}, \\ (\mathcal{F}_{i_i} - \mathcal{F}_{a_i})(L_i) = -\text{Ad}_{g_{ij}}^* \mathcal{F}_{J_j}, \end{cases} \quad (8.26)$$

where $\bar{\mathcal{M}}_i$ is the inertia density matrix of the cross-section, $\bar{\mathcal{F}}_{e_i}$ is the distributed external load, and $\mathcal{F}_{i_i}$, $\mathcal{F}_{a_i}$ are the internal wrenches due to material elasticity and distributed actuation, respectively. The two boundary conditions state that the internal wrench at each end of the soft link must balance the wrench transmitted through the joints connecting it to its neighbours.

Internal elastic and damping wrench. Using a Hooke-like linear constitutive law, the internal wrench of a soft link is the derivative of its elastic energy potential:

$$\mathcal{F}_{i_i}(X_i) = \Sigma_i(\xi_i - \xi_i^*) + \Upsilon_i \dot{\xi}_i = \Sigma_i \Phi_\xi q_i + \Upsilon_i \Phi_\xi \dot{q}_i, \quad (8.27)$$

where Σ_i and Υ_i are the screw elasticity and screw damping matrices of the cross-section. It is precisely through Eq. (8.27) that the material parameters ρ (via the inertia density $\bar{\mathcal{M}}_i$), E (via Σ_i) and η (via Υ_i) enter the model – linearly in q and $\dot{q}$, as already noted in Section 3.1.

Assembled equations of motion. Projecting Eqs. (8.24)–(8.26) onto the generalised coordinates via the Jacobian J (Section 8.1.2) and integrating over the length of each link recovers the standard robotics-like dynamic equation used throughout this thesis,

$$M(q)\ddot{q} + [C(q, \dot{q}) + D]\dot{q} + Kq = A(q)u + Q(q), \quad (3.7 \text{ revisited})$$

with M , C , D , K , Q assembled by Gaussian quadrature exactly as given in Eqs. (3.8)–(3.10) of the main chapter. The material matrices $\bar{M}$, Σ and Υ used there are the screw inertia density, screw elasticity, and screw damping matrices of the (uniform circular, in this work) cross-section introduced above.

8.2 Properties demonstration with generalized coordinates

Theorem 8.2.1. *The inertia matrix M is bounded and symmetric positive definite for any q , i.e., there exist two positive constants λ_1 and λ_2 such that*

$$\lambda_1 I_n \leq M(q) \leq \lambda_2 I_n \quad \forall q.$$

Proof. Exploiting [13], Eq.[37a], and considering just a single link (for lighter computations), as we can exploit the linearity of the integral to generalize this result to N links, we obtain:

$$M(q) = \int_0^L J^T \bar{M} J dX$$

Let us start by proving the symmetry and positive definiteness: $0 < M(q) = M(q)^T$. If we consider the formula that defines the inertia matrix, we get:

$$\begin{aligned} M(q) &= \int_0^L J(q, X)^T \bar{M} J(q, X) dX = \int_0^L \left(J(q, X)^T \bar{M} J(q, X) \right)^T dX \\ &= \int_0^L J(q, X)^T \bar{M} J(q, X) dX = M(q)^T \end{aligned}$$

To prove positive definiteness, we first point out that $\bar{M}$ is a positive definite diagonal matrix composed of area and the second moment of area, which are even functions. Then, $\forall y \in \mathbb{R}^n$ we need to prove that $0 < y^T \bar{M} y$; it follows that

$$\begin{aligned} y &= Jx \quad \forall x \in \mathbb{R}^n \\ y^T \bar{M} y &= (Jx)^T \bar{M} (Jx) = x^T J^T \bar{M} J x > 0 \end{aligned}$$

positive definiteness holds for $Jx = 0$ if and only if $x = 0$, this corresponds to state that the only generalized velocity producing zero twist throughout the entire model is $\dot{q} = 0$.

To ensure boundedness of the inertia matrix:

- given $M(q)$ positive definite, we can exploit the fact that there exists a minimum eigenvalue so that $\lambda_1 I_n \leq M(q)$, where $\lambda_1 = \min_i \text{eig}_i(M(q))$
- $\|M(q)\| \leq \int_0^L \|J\|^2 \|\bar{M}\| dX$ where clearly $\bar{M}$ is bounded by beam finiteness; equally, the Jacobian term is bounded as $\|J\| \leq J_{Max}$, as the material is finite, and thus, length dependent values are finite.

□

Theorem 8.2.2. *The matrix $\dot{M} - 2C$ is skew-symmetric, i.e.,*

$$x^T (\dot{M} - 2C)x = 0 \quad \forall x \in \mathbb{R}^n$$

Proof. By definition and exploiting [13], Eq. [37a],Eq. [37b] :

$$\begin{aligned}\dot{M}(q) &= \frac{d}{dt} \int_0^L J(q, X)^T \bar{M} J(q, X) dX = \int_0^L \dot{J}(q, X)^T \bar{M} J(q, X) + J(q, X)^T \bar{M} \dot{J}(q, X) dX \\ C &= \int_0^L J^T (\text{ad}_\eta^* \bar{M} J + \bar{M} \dot{J}) dX\end{aligned}$$

Putting the two equations together, we get

$$\begin{aligned}\dot{M} - 2C &= \int_0^L \dot{J}^T \bar{M} J + J^T \bar{M} \dot{J} - 2J^T \text{ad}_\eta^* \bar{M} J - 2J^T \bar{M} \dot{J} dX = \\ &= \int_0^L \dot{J}^T \bar{M} J - J^T \bar{M} \dot{J} - 2J^T \text{ad}_\eta^* \bar{M} J dX.\end{aligned}$$

Let us apply $x = \dot{q} \in \mathbb{R}^n$ on the left and right of the previous expression to show the desired property:

$$\begin{aligned}\dot{q}^T (\dot{M} - 2C) \dot{q} &= \int_0^L \dot{q}^T \dot{J}^T \bar{M} J \dot{q} - \dot{q}^T J^T \bar{M} \dot{J} \dot{q} - 2\dot{q}^T J^T \text{ad}_\eta^* \bar{M} J \dot{q} dX = \\ &= \int_0^L \underbrace{(\dot{\eta}(X) - J\ddot{q})^T}_{\phi^T} \bar{M} \eta(X) - \eta(X)^T \bar{M} \underbrace{(\dot{\eta}(X) - J\ddot{q})}_{\phi} - 2\eta(X)^T \text{ad}_\eta^* \bar{M} \eta(X) dX = \\ &= \int_0^L \phi^T \bar{M} \eta(X) - \eta(X)^T \bar{M} \phi - 2\eta(X)^T \text{ad}_\eta^* \bar{M} \eta(X) dX\end{aligned}$$

where $\eta(X) = J\dot{q}$. We have that the first two addenda of the integral are scalars, as $\eta(x)$ and $\phi(X)$ are both vectors, hence, the difference between the two is zero as they cancel each other. Finally,

$$\eta(X)^T \text{ad}_\eta^* \bar{M} \eta(X) = (\bar{M} \eta(X))^T \text{ad}_\eta \eta = 0$$

is derived from the property of the Lie algebra $[\eta, \eta] = 0$, where we obtain the inversion through the use of the co-adjoint property $\langle \eta, \text{ad}_\eta^* F \rangle = \langle F^T, -\text{ad}_\eta \eta \rangle$.

We can conclude that

$$x^T (\dot{M} - 2C) x = 0.$$

□

Theorem 8.2.3. *The Coriolis matrix $C(q, \dot{q})$ satisfies*

$$\|C(q, \dot{q})\| \leq \alpha_0 \|\dot{q}\| \quad \forall q \in \mathbb{R}^n$$

for some bounded constant α_0 .

Proof. Starting from the definition of the Coriolis matrix ([13],Eq.(37b)) :

$$C = \int_0^L J^T (\text{ad}_\eta^* \bar{M} J + \bar{M} \dot{J}) dX$$

we exploit the Cauchy-Schwarz inequality and linearity of the integral. We consider the first addendum of the integration function $J^T \text{ad}_\eta^* \bar{\mathcal{M}} J$. It results

$$\|\text{ad}_\eta^*\| \leq c \|J\| \|\dot{q}\| \quad (8.28)$$

$$\|J\| \leq J_M \quad (8.29)$$

$$\|J^T \text{ad}_\eta^* \bar{\mathcal{M}} J\| \leq \|J\|^2 c J_M \|\dot{q}\| \leq c J_M^2 m_M \|\dot{q}\| \quad (8.30)$$

where equation (8.28) builds on the definition of the Adjoint operator used in Appendix A, as the operator is linear to $\eta = (\omega^T, v^T)^T$ and thus we obtained the following relationship $\|\text{ad}_\eta^*\| \leq c \|\eta\|$.

Instead, Eq.(8.29) exploits the previously used result of the boundedness of the Jacobian, as the model is finite, and so it results that $\|J\| \leq J_M$.

Moreover, we used the following approximation to bound the screw inertia density matrix: $\|\bar{\mathcal{M}}\| \leq m_M$, which holds for the finiteness of the beam.

For what concerns the second addendum $J^T \bar{\mathcal{M}} \dot{J}$, we must prove the boundedness of the derivative of the geometric Jacobian. From Appendix B of [13], we get

$$\begin{aligned} J(X + h_k) &= \text{Ad}_{\exp(\Omega)}^{-1} [J(X) + [0_{6 \times n^-} (ad_\eta T_\Omega + \dot{T}_\Omega) B_\Omega \ 0_{6 \times n^+}]] \\ \|J(X + h_k)\| &\leq \|Ad_{\exp(\Omega)}^{-1}\| \|J\| + V \leq J_M \|\dot{q}\| \end{aligned}$$

where we have

$$\begin{aligned} \|Ad_{\exp(\Omega)}^{-1}\| &\leq c_g \\ V &= Ad_{\exp(\Omega)}^{-1} (ad_\eta T_\Omega + \dot{T}_\Omega) B_\Omega \quad \|V\| \leq \|ad_\eta\| \|T_\Omega\| \|B_\Omega\| + \|\dot{T}_\Omega\| \|B_\Omega\| \\ \|ad_\eta\| &\leq c \|J\| \|\dot{q}\| \leq c J_M \|\dot{q}\| \end{aligned}$$

$\|T_\Omega\|$: following the definition shown in Appendix A of Mathew's [13], the tangent operator is bounded if $\|\Omega\| \neq 0$, which is true by construction

$$\|\dot{T}_\Omega\| \leq k_T \|B_\Omega\| \|\dot{q}\| : \text{ as it linearly depends on } \dot{\theta} = \frac{\Omega^T \dot{\Omega}}{\|\Omega\|} \text{ and on } \dot{\text{ad}}_\Omega = \text{ad}_{\dot{\Omega}} \leq c_{ad} \|\dot{\Omega}\| \leq c_{ad} \|B_\Omega\| \|\dot{q}\|$$

Putting the above implications together, we get

$$\|J^T \bar{\mathcal{M}} \dot{J}\| \leq J_M m_M J_M \|\dot{q}\|.$$

Conclusively, we can put together the results for the two terms to get

$$\|C(q, \dot{q})\| = \left\| \int_0^L J^T \text{ad}_\eta^* \bar{\mathcal{M}} J + J^T \bar{\mathcal{M}} \dot{J} dX \right\| \leq \int_0^L c J_M^2 m_M + J_M^2 m_M dX \|\dot{q}\|$$

□

Theorem 8.2.4. *The gravity vector $g(q)$ satisfies*

$$\|g(q)\| \leq \gamma_0$$

for some bounded constant γ_0 .

Proof. To prove the boundedness of gravity, we must first keep into account that in the GVS model, gravity is included in the external forces $F(q, \dot{q})$.

We find that the gravitational effect is

$$\mathbf{g}(q) = \int_0^L J^T \begin{pmatrix} 0 \\ \rho \mathcal{A}(X) \mathbf{g} \end{pmatrix} dX.$$

We use $\mathbf{g}$ to identify gravity in the body frame, obtained exploiting the inverse kinematics of the local effect of $\mathbf{g}$. Then we get the following inequality

$$\begin{aligned} \|\mathbf{g}(q)\| &\leq \int_0^L \|J^T\| \left\| \begin{pmatrix} 0 \\ \rho \mathcal{A}(X) \mathbf{g} \end{pmatrix} \right\| dX \\ &\leq \int_0^L J_M \rho \mathcal{A}_M \|\mathbf{g}\| dX, \end{aligned}$$

where we leveraged the usual approximation of the geometric Jacobian and exploit the existence of the maximal area due to material finiteness. We can conclude that the gravity vector is bounded by a finite scalar that depends on the geometric parameters:

$$\|\mathbf{g}(q)\| \leq J_M g_0 \rho \mathcal{A}_M L.$$

□

Theorem 8.2.5. *The Lagrangian model can be expressed linearly with respect to the system's base parameters π , i.e., there exists a regressor matrix $Y(\cdot)$ such that*

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + \mathbf{g}(q) + Kq + D\dot{q} = Y(q, \dot{q}, \ddot{q})\pi.$$

Proof. We demonstrate the result just for the single link case; one can exploit linearity to

generalize it. Starting from the mass matrix:

$$M(q) = \int_0^L J(q, X)^T \bar{M} J(q, X) dX \quad (8.31)$$

$$\bar{M} = \rho \underbrace{\begin{pmatrix} J_x & 0 & 0 & 0 & 0 & 0 \\ 0 & J_y & 0 & 0 & 0 & 0 \\ 0 & 0 & J_z & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathcal{A} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathcal{A} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathcal{A} \end{pmatrix}}_I \quad (8.32)$$

where the diagonal entries of the screw density matrix are $J_{x,y,z}(X)$, i.e., the second moment of area of each component, $A(X)$, i.e., the area of the beam, and ρ , which is the mass density of the beam. We can consider these values as constant. This assumption is reasonable since, even if the properties were considered position-dependent, the linearity of the GVS link description and of the integral operation would preserve the linearity of the model.

Exploiting (8.32) and assuming that the aforementioned parameters are constant, we obtain the following mass matrix

$$M(q) = \rho \underbrace{\int_0^L J(q, X)^T \cdot I \cdot J(q, X) dX}_{M_q} = \rho M_q$$

where ρ is a scalar and $M_q \in \mathbb{R}^{n \times n}$.

Now, the same strategy must be applied to the Coriolis matrix

$$C(q, \dot{q}) = \int_0^L J(q, X)^T \text{ad}_\eta^* \bar{M} J(q, X) + J(q, X)^T \bar{M} \dot{J}(q, \dot{q}, X) dX.$$

We can resort to the linearization (8.32), as before, to obtain

$$C(q, \dot{q}) = \rho \underbrace{\int_0^L \left(J(q, X)^T \text{ad}_\eta^* \cdot I \cdot J(q, X) + J(q, X)^T \cdot I \cdot \dot{J}(q, \dot{q}, X) \right) dX}_{C_q} = \rho C_q \quad (8.33)$$

Let us proceed with the damping force described by the relation below:

$$D = \int_0^L B_\xi(X)^T \Upsilon B_\xi(X) dX$$

$$\Upsilon = \eta \underbrace{\begin{pmatrix} J_x & 0 & 0 & 0 & 0 & 0 \\ 0 & 3J_y & 0 & 0 & 0 & 0 \\ 0 & 0 & 3J_z & 0 & 0 & 0 \\ 0 & 0 & 0 & 3\mathcal{A} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathcal{A} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathcal{A} \end{pmatrix}}_{\mathcal{U}} \quad B_\xi \in \mathbb{R}^{6 \times n} \text{ basis function dependent on } X \quad (8.34)$$

We use the same strategy as above, the only change being the base scalar η , characterizing the material damping.

Due to the dependence on the value of X , we obtain the product of the basis matrix $B_\xi(X)$ times a diagonal matrix. Hence, the matrix describing the damping effect can be computed as:

$$B_\xi = \begin{pmatrix} \mathbf{p}_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{p}_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{p}_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{p}_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{p}_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{p}_6 \end{pmatrix} \in \mathbb{R}^{6 \times (\sum_{i=1}^6 n_i)} \text{ where } \mathbf{p}_k = \sum_{i=0}^{n_i} X^i$$

where each zero symbol in the matrix $B_\xi(X)$ represents a row vector whose dimension is equal to the number of coefficients of the polynomial $\mathbf{p}_i$ belonging to the same column block.

$$D = \int_0^L B_\xi(X)^T \Upsilon B_\xi(X) dX = \eta \int_0^L B_\xi(X)^T \cdot \mathcal{U} \cdot B_\xi(X) dX \quad (8.35)$$

As the matrix $\mathcal{U} \in \mathbb{R}^{6 \times 6}$ is diagonal and $B_\xi(X)$ is block diagonal, the product of the corresponding diagonal blocks in $B_\xi(X)^T \mathcal{U} B_\xi(X)$ results in a polynomial characterized by the description of $\mathcal{H}_k$:

$$\mathcal{H}_k = \int_0^L \mathbf{p}_k^T(X) \cdot k_U \cdot \mathbf{p}_k(X) dX \quad \mathcal{H}_k \in \mathbb{R}^{n_k \times n_k} \leftarrow (\text{ as } \mathbf{p}_k^T \text{ is a column vector of size } n_k), \quad (8.36)$$

$$[\mathcal{H}_k]_{jl} = a_k \frac{L^{j+l+1}}{j+l+1} \quad \forall k \in [1, 6]; \forall j, l \in [0, n_k] \quad (8.37)$$

$$D = \eta \begin{pmatrix} \mathcal{H}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathcal{H}_2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{H}_3 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{H}_4 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{H}_5 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{H}_6 \end{pmatrix} \quad (8.38)$$

With the same strategy, we can obtain the stiffness matrix

$$K = \int_0^L B_\xi(X)^T \Sigma B_\xi(X) dX \quad (8.39)$$

$$\Sigma = \begin{pmatrix} GJ_x & 0 & 0 & 0 & 0 & 0 \\ 0 & EJ_y & 0 & 0 & 0 & 0 \\ 0 & 0 & EJ_z & 0 & 0 & 0 \\ 0 & 0 & 0 & E\mathcal{A} & 0 & 0 \\ 0 & 0 & 0 & 0 & G\mathcal{A} & 0 \\ 0 & 0 & 0 & 0 & 0 & G\mathcal{A} \end{pmatrix} \quad B_\xi \in \mathbb{R}^{6 \times n} \text{ basis function dependent on } X \quad (8.40)$$

where Σ is the screw elasticity matrix, whose diagonal entries depend on G, E shear modulus and Young modulus, respectively. We must exploit the relationship between stiffness and shear modulus, so that the screw elasticity matrix can be rewritten as a linear function of the Young's modulus. To do so the following relationship based on the assumption of a constant Poisson modulus $\nu = 0.5$ has been used:

$$E = 2G(1 + \nu).$$

We can now implement the linearization of the whole matrix, as follows:

$$\Sigma = E \underbrace{\begin{pmatrix} \frac{1}{2(1+\nu)}J_x & 0 & 0 & 0 & 0 & 0 \\ 0 & J_y & 0 & 0 & 0 & 0 \\ 0 & 0 & J_z & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathcal{A} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2(1+\nu)}\mathcal{A} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2(1+\nu)}\mathcal{A} \end{pmatrix}}_{\mathcal{E}} \quad (8.41)$$

$$K = \int_0^L B_\xi(X)^T \Sigma B_\xi(X) dX = E \int_0^L B_\xi(X)^T \cdot \mathcal{E} \cdot B_\xi(X) dX \quad (8.42)$$

Exploiting the same reasoning about the dependence on X of the discretization base, we can

compute the closed form solution of the stiffness

$$K = E \begin{pmatrix} \mathcal{F}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathcal{F}_2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{F}_3 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{F}_4 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{F}_5 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{F}_6 \end{pmatrix} \quad (8.43)$$

$$\mathcal{F}_k = \int_0^L \mathbf{p}_k^T(X) \cdot k_E \cdot \mathbf{p}_k(X) dX \in \mathbb{R}^{n_k \times n_k}, \quad [\mathcal{F}_k]_{jl} = b_k \frac{L^{j+l+1}}{j+l+1} \quad \forall k \in [1, 6]; \forall j, l \in n_i \quad (8.44)$$

So, we can see that to compute the stiffness and damping matrices, we just need the length of the beam and the base parameters.

Lastly, we demonstrate the linearizability of the gravitational force:

$$\mathbf{g}(q) = \int_0^L J(q, X)^T \mathcal{F}_g dX = \int_0^L J(q, X)^T \begin{pmatrix} \mathbf{0} \\ \rho \mathcal{A}(X) \mathbf{g} \end{pmatrix} dX = \int_0^L J(q, X)^T \rho \mathcal{A}(X) \begin{pmatrix} \mathbf{0} \\ R^T(q, X) \mathbf{g}_0 \end{pmatrix} dX \quad (8.45)$$

$$= \rho \int_0^L J(q, X)^T \mathcal{A}(X) \begin{pmatrix} \mathbf{0} \\ R^T(q, X) \mathbf{g}_0 \end{pmatrix} dX \quad (8.46)$$

The model has been developed using the following base parameters:

$$p = \begin{pmatrix} \rho \\ E \\ \eta \end{pmatrix}$$

To conclude the proof, we want now to put everything together to build the linearized model

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + \mathbf{g}(q) + Kq + D\dot{q} = Y(q, \dot{q}, \ddot{q})\pi$$

Hence starting from the assumption of constant parameters along the whole DLO and exploiting the algebraic relationships described above between the physical parameters and the matrices in which they appear, we can obtain the formulation of the regressor in the generalized coordinates.

$$M(q)\ddot{q} = \underbrace{\rho}_{\in \mathbb{R}^{n \times 1}} M_q \ddot{q} = \left[\int_0^L J(q, X)^T I J(q, X) dX \right] \rho = Y_M(q, \ddot{q}) \rho$$

$$\begin{aligned}
C(q, \dot{q})\dot{q} &= \rho \underbrace{C_q \dot{q}}_{\in \mathbb{R}^{n \times 1}} = \left[\int_0^L J(q, X)^T (\text{ad}_\eta^* J J(q, X) + I \dot{J}(q, \dot{q}, X)) dX \right] \dot{q} \cdot \rho = \\
&= Y_C(q, \dot{q})\rho
\end{aligned}$$

$$\mathbf{g}(q) = \int_0^L \rho J(q, X)^T \mathcal{A}(X) \begin{pmatrix} \mathbf{0} \\ R^T(q, X)g_0 \end{pmatrix} = \left[\int_0^L J(q, X)^T \mathcal{A}(X) \begin{pmatrix} \mathbf{0} \\ R^T(q, X)g_0 \end{pmatrix} \right] \rho$$

Exploiting the fact that the three matrices above depend on the same constant parameter, they can be easily summed to obtain the regressor's element related to the mass density ρ .

Now we remain to show the linearized form of the stiffness and damping components. From the results above, we know that D and K are matrices that depend only on the length of the DLO and can therefore be computed offline.

$$K \cdot q = E \begin{pmatrix} \mathcal{F}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathcal{F}_2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{F}_3 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{F}_4 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{F}_5 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{F}_6 \end{pmatrix} \begin{pmatrix} q_1 \\ q_2 \\ \vdots \\ q_6 \end{pmatrix} = \underbrace{\begin{pmatrix} \mathcal{F}_1 q_1 \\ \mathcal{F}_2 q_2 \\ \mathcal{F}_3 q_3 \\ \mathcal{F}_4 q_4 \\ \mathcal{F}_5 q_5 \\ \mathcal{F}_6 q_6 \end{pmatrix}}_{Y_K \in \mathbb{R}^{n \times 1}} E$$

where $\mathcal{F}_k q_k \in \mathbb{R}^{n_k \times 1}$. In fact, by exploiting the block diagonal structure of the matrix that defines K we can divide the generalized coordinates $\mathbf{q}$ in 6 sub-vectors, each of these is multiplied by a single block diagonal polynomial $\mathcal{F}_k$.

Using the same strategy implemented for the stiffness component, we obtain the following structure for the linearized version of the damping element:

$$D \cdot \dot{q} = \underbrace{\begin{pmatrix} \mathcal{H}_1 \dot{q}_1 \\ \mathcal{H}_2 \dot{q}_2 \\ \mathcal{H}_3 \dot{q}_3 \\ \mathcal{H}_4 \dot{q}_4 \\ \mathcal{H}_5 \dot{q}_5 \\ \mathcal{H}_6 \dot{q}_6 \end{pmatrix}}_{Y_D \in \mathbb{R}^{n \times 6}} \eta$$

Finally, we can put everything together, obtaining the desired result

$$Y(q, \dot{q}, \ddot{q})p = \begin{bmatrix} Y_{M,C,g} & Y_K & Y_D \end{bmatrix} \begin{pmatrix} \rho \\ E \\ \eta \end{pmatrix}$$

with $Y(\cdot) \in \mathbb{R}^{n \times 3}$.

□

8

Bibliography

- [1] Costanza Armanini, Frédéric Boyer, Anup Mathew, Christian Duriez, and Federico Renda. Soft robots modeling: A structured overview. *IEEE Transactions on Robotics*, 39:1728–1748, 06 2023.
- [2] Veronica E. Arriola-Rios, Puren Guler, Fanny Ficuciello, Danica Kragic, Bruno Siciliano, and Jeremy L. Wyatt. Modeling of deformable objects for robotic manipulation: A tutorial and review. *Frontiers in Robotics and AI*, Volume 7 - 2020, 2020.
- [3] Aude Billard and Danica Kragic. Trends and challenges in robot manipulation. *Science*, 364(6446):eaat8414, 2019.
- [4] Alessio Caporali, Ignacio Cuiral-Zueco, Gonzalo López-Nicolás, and Gianluca Palli. Robotic perception and manipulation of deformable linear objects: A survey. *The International Journal of Robotics Research*, 0(0):02783649261432253, 2026.
- [5] Alessio Caporali, Piotr Kicki, Kevin Galassi, Riccardo Zanella, Krzysztof Walas, and Gianluca Palli. Deformable linear objects manipulation with online model parameters estimation. *IEEE Robotics and Automation Letters*, 9(3):2598–2605, 2024.
- [6] J.J Craig. *Introduction to robotics : mechanics & control / John J. Craig*. Addison-Wesley Pub. Co.,, 1986.
- [7] Cosimo Della Santina, Christian Duriez, and Daniela Rus. Model-based control of soft robots: A survey of the state of the art and open challenges. *IEEE Control Systems*, 43:30–65, 06 2023.
- [8] Daniel Feliu-Talegon and Cosimo Della Santina. Manipulation with stability guarantees: Linear deformable objects with non-negligible physical response grasped at multiple location, 2026.

- [9] Manuel Gnucci and Riccardo Marino. Adaptive tracking control for underactuated mechanical systems with relative degree two. *Automatica*, 129:109633, 07 2021.
- [10] Pascal Jung, Sigrid Leyendecker, Joachim Linn, and Michael Ortiz. A discrete mechanics approach to the cosserat rod theory—part 1: static equilibria. *International Journal for Numerical Methods in Engineering*, 85(1):31–60, 2011.
- [11] Aohua Liu, Kun Qian, Zhaokun Yue, Zhuoxuan Wang, Boyi Duan, and Shan Luo. Learning physics-aware sensorimotor model with visual–tactile sensing for dlo manipulation. *IEEE/ASME Transactions on Mechatronics*, pages 1–12, 2026.
- [12] Fei Liu, Entong Su, Jingpei Lu, Ming Li, and Michael C. Yip. Robotic manipulation of deformable rope-like objects using differentiable compliant position-based dynamics. *IEEE Robotics and Automation Letters*, 8:3964–3971, 2023.
- [13] Anup Teejo Mathew, Daniel Feliu-Talegon, and et al. Reduced order modeling of hybrid soft-rigid robots using global, local, and state-dependent strain parameterization. *The International Journal of Robotics Research*, 44(1):129–154, 2025.
- [14] Daniele Pucci, Francesco Romano, and Francesco Nori. Collocated adaptive control of underactuated mechanical systems. *IEEE Transactions on Robotics*, 31(6):1527–1536, 2014.
- [15] Pietro Pustina, Pablo Borja, Cosimo Della Santina, and Alessandro De Luca. P-sati-d shape regulation of soft robots. *IEEE Robotics and Automation Letters*, PP:1–8, 11 2022.
- [16] Pietro Pustina, Cosimo Della Santina, Frédéric Boyer, Alessandro De Luca, and Federico Renda. Input decoupling of lagrangian systems via coordinate transformation. *IEEE Transactions on Robotics*, PP:1–13, 2024.
- [17] Pietro Pustina, Cosimo Della Santina, and Alessandro De Luca. Feedback regulation of elastically decoupled underactuated soft robots. *IEEE Robotics and Automation Letters*, 7:4512–4519, 04 2022.
- [18] Pietro Pustina, Cosimo Della Santina, Frédéric Boyer, Alessandro De Luca, and Federico Renda. Input decoupling of lagrangian systems via coordinate transformation: General characterization and its application to soft robotics. *IEEE Transactions on Robotics*, 40:2098–2110, 2024.
- [19] Bruno Siciliano and Oussama Khatib. *Springer Handbook of Robotics*. Springer, 2nd edition, 2016.
- [20] Jean-Jacques E Slotine and Weiping Li. *Applied Nonlinear Control*. Prentice-Hall, Englewood Cliffs, NJ, 1991.

- [21] Sebastien Tiburzio, Tom´as Coleman, Daniel Feliu-Talegon, and Cosimo Santina. Controlling deformable objects with nonnegligible dynamics: A shape-regulation approach to end-point positioning. *IEEE Transactions on Robotics*, PP:1–16, 01 2025.
- [22] Matthias Tummers, Vincent Lebastard, Frédéric Boyer, Jocelyne Troccaz, Benoît Rosa, and M. Taha Chikhaoui. Cosserat rod modeling of continuum robots from newtonian and lagrangian perspectives. *IEEE Transactions on Robotics*, 39(3):2360–2378, 2023.
- [23] Hang Yin, Anastasia Varava, and Danica Kragic. Modeling, learning, perception, and control methods for deformable object manipulation. *Science Robotics*, 6(54):eabd8803, 2021.