

GEOMETRIC SOFT ROBOTICS: A FINITE ELEMENT APPROACH

A Dissertation
Presented to
The Academic Faculty

By

Stanislao Grazioso

In Partial Fulfillment
of the Requirements for the Degree of
Doctor of Philosophy in Industrial Engineering



Università degli studi di Napoli Federico II
Department of Industrial Engineering
December 2017

Advisors:
Prof. Giuseppe Di Gironimo
Prof. Bruno Siciliano

The Chair of the Doctoral Programme:
Prof. Michele Grassi

*“If you are not failing your time,
you are not doing R & D.
If everything works,
you are not asking tough enough questions”
B.S.*

Preface

It was a mid-december 2014 day in my lab, and, as usual, I was working on 3D vision, the main topic of my doctorate work. I remember that Domenico was writing a proposal for a position of the 2015 EUROfusion Engineering Grants call, the most prestigious founding opportunity for early-stage doctoral and post-doctoral researchers supported by EUROfusion, the European consortium for the development of fusion energy. He asked me and Mario, at that time working on his master thesis, to help him in writing a proposal for another position, which was focused on remote maintenance inside fusion reactors. The problem was that the deadline was that same night. And, EUROfusion aimed at found one project for each position. Excited from the idea, we started to work on it. In the afternoon, Professor Giuseppe Di Gironimo came in the lab, and, on the advice of Domenico, he decided to submit the proposal with me as principal investigator. Among Mario and me, I was selected only because an eligibility criterion of the call required the applicant to possess a master degree in Engineering. In one day, we wrote that proposal. Since the objective was the realistic simulation of flexible robotic manipulators, we focused it in closing the gap between continuum mechanics and robotics. This has always represented a great challenge for me, inspired from the robotics class of Professor Bruno Siciliano and the computational mechanics class of Professor Luciano Rosati. After two months, a mail from EUROfusion invited me in Garching, München, for the interview, in front of the most important representatives of the fusion research worldwide. One month later, the results stated that our project was funded. I remember Professor Vincenzo Coccoresse, the president of CREATE Consortium, coming in our lab to personally congratulate for the successful outcome. I would never forget the emotions of that moment. As a result, after almost one year of activities, the topic of my doctorate work was shifted to the new one. The challenge to enable safer remote applications in challenging environments gave me the opportunity to work in the field of continuum robotics, very exciting for a mechanical engineer in love with robotics.

It is my pleasure to acknowledge the many people who have been the sources of help and inspiration for this work. First of all, I would like to gratefully thank my advisors, Professor Giuseppe Di Gironimo and Professor Bruno Siciliano. Professor Giuseppe Di Gironimo has been a model for me since we met, in my master thesis period, for his special scientific and personal values. He has given me a solid foundation on simulation and virtual prototyping of mechanical systems. Our discussions represented a fundamental element for my growth, and I will always thank him for the opportunities he has given me. Doing robotics research under the supervision of Professor Bruno Siciliano is a privilege reserved to the Neapolitan community. His teachings have been essential for me in developing a system-oriented perspective of Engineering; every talk with him has always been deeply inspiring. I sincerely thank Professor Antonio Lanzotti, who has been important for me since my undergraduate education. His passion in doing research, combined with his leadership, is a meaningful example for young people.

During this research, I have had the opportunity to interact with Daniel Iglesias at RACE/CCFE, Professor Olivier A. Bauchau at University of Maryland and Professor William Singhose at Georgia Institute of Technology. I am really grateful to Daniel Iglesias, who introduced me the main problems of simulating maintenance and inspection procedures in fusion reactors. Professor Oliver A. Bauchau transmitted me the passion on the mathematical beauty behind the modeling of mechanical systems. From Professor William Singhose I learned that the applications of greater impact come from simple basic ideas. Further, I would like to acknowledge Professor Jon Selig and Professor Sarthak Misra, for the time they spent to review this manuscript, as well as for their valuable comments. The beauty of doing research is that it offers a possibility to make new friends. The University of Naples Federico II has been a wonderful environment. I would like to thank my friends Domenico M., Domenico C., Teodorico, Stefano, Andrea, Rocco as well as Luca, Andrea for the never boring time that we spent together. During my visiting period, I have been lucky to meet great people, that helped to make that time extraordinary. Among all, I would like to mention Valentin, who, besides being crucial for my training on beam theory, has always been a dear friend. Further, many thanks go to the members of the research groups I visited as well as the friends I met during my stays in Oxford, College Park and Atlanta. In each of these places I felt as home. A special thank goes to my dear friends of "life and research", Mario and Gennaro. Although we have often been in different places, we have always had the opportunity to meet and exchange ideas. Heartfelt thanks to all my other friends, in particular to Salvatore, Achille, Peppe, Raffaele, Mateusz, as well as Giovanni, Giuseppe, Alfonso, Vincenzo, Fabio. Nothing could express the feeling of love and gratitude towards my family, my parents and my sister Anna. Without their support, this thesis would not exist. My last words go to Angela. Her unconditional love is my daily source of energy and equilibrium. Finally, I want to conclude with an inspiring thought of Professor Wayne J. Book, mentor of life farther than robotics, expressed during his lifetime talk at IROS 2016: *"In robotics we always try to reject external disturbances, but we should begin to accept and use them. In life, disturbances are opportunities: they put us in front of choices and lead us in being who we are"*.

Stanislao Grazioso

Napoli, December 2017

This research has been partially supported by the EEG-2015/21 project. This work has been partially carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission. The authors are solely responsible for its content.

Abstract

Enabling remote semi-autonomous operations in hazardous environments is a challenging technological problem, given the difficulty to access in confined and constrained spaces using classical robotic systems. Inspired by biological trunks and tentacles, soft continuum robots constitute a possible solution to this problem, for their ability to traverse confined spaces, manipulate objects in complex environments, and conform their shape to nonlinear curvilinear paths. The need of reaching difficult-to-access industrial sites for maintenance and inspection procedures or anatomical sites for less invasive robotic surgery mainly motivates the current research. Despite the recent advances in the design and fabrication of soft robots, the community still suffers for the lack of a consolidate modeling framework for simulating their mechanical behavior. Such a modeling framework is the necessary condition for developing new physical design and control strategies, as well as path planning algorithms. Indeed, despite their appreciable features, soft robots usually generate undesired vibrations during normal procedures. This is one of the main reasons which still limits their potentially wide use in real scenario. Realistic modeling frameworks might leverage the development of model-based predictive controllers to compensate for the undesired vibrations, as well as design concepts and optimized trajectories to avoid the excitation of the vibration modes of the mechanical structure.

The main objective of the thesis is to develop a unified mathematical framework for simulating the mechanical behavior of soft continuum robotic manipulators, which can also accommodate the dynamic simulation of classical rigid robots. The computer implementation of this theoretical framework leads to the development of SimSOFT, a physics engine for soft robots. The formulation has been validated through literature benchmark and some applications are presented. One of the major strengths of the framework is that it can accommodate the realistic simulation of kinematic trees or loops constituted either by rigid or soft arms connected by rigid or flexible joints. The simulation of hybrid mechanisms, composed by classical rigid kinematic chains and soft continuum manipulators, which can be used to have larger dexterity in smaller workspaces, as they are easily to miniaturize, is thus possible. To the best of the author's knowledge, the mathematical models developed in the thesis constitute the first attempt in the robotics community towards a unified framework for the dynamics of soft continuum multibody systems.

Contents

Introduction	1
1 The geometric formalism	7
1.1 Rigid body transformations	8
1.2 Homogeneous transformation matrices	9
1.3 Twists	10
1.4 Screws: a geometrical description of twists	11
1.5 Exponential coordinate representation	11
1.6 Wrenches	13
1.7 Equations of motion of a rigid body	14
1.7.1 Variations	14
1.7.2 Kinetic energy	15
1.7.3 Potential energy	15
1.7.4 Hamiltonian formulation	15
1.8 Geometric time integration	17
1.8.1 Global parametrization of rotation	18
1.8.2 Global parametrization-free equations of motion	20
1.9 Summary	21
2 Kinematics	22
2.1 Continuum arm	22
2.1.1 Geometry of the reference curve	22
2.1.2 Position field	24
2.1.3 Deformation field	24
2.2 Piecewise constant deformation soft arm	24
2.2.1 Forward kinematics	25
2.2.2 Inverse kinematics	27
2.2.3 Special case: constant deformation soft arm	29
2.3 Summary	32
3 Differential kinematics and statics	33
3.1 Continuum arm	33
3.1.1 Velocity field	33
3.1.2 Compatibility equations	34
3.1.3 Acceleration field	34

3.1.4	Strain energy	34
3.1.5	Statics	35
3.1.6	Example	36
3.2	Piecewise constant deformation soft arm	38
3.2.1	Soft geometric Jacobian	38
3.2.2	Inverse differential kinematics	45
3.2.3	Statics	45
3.2.4	Special case: constant deformation soft arm	47
3.2.5	Example	49
3.3	Summary	50
4	Dynamics	51
4.1	Continuum arm	51
4.1.1	Kinetic energy	51
4.1.2	Hamiltonian formulation	52
4.1.3	Equations of motion	52
4.1.4	Example	53
4.2	Piecewise constant deformation soft arm	55
4.2.1	Hamiltonian formulation	55
4.2.2	Special case: constant deformation soft arm	57
4.2.3	Example	58
4.3	Summary	58
5	Constrained dynamics	59
5.1	The coupling approach	59
5.2	Relative transformations	60
5.3	Rigid joint	61
5.3.1	Kinematic constraints	63
5.4	Flexible joint	65
5.4.1	Joint deflection	66
5.4.2	Internal forces of the joint	66
5.5	The constrained dynamical system	67
5.5.1	Kinematic configuration	67
5.5.2	Hamiltonian formulation	69
5.5.3	Equations of motion	69
5.5.4	Time integration	69
5.6	Test examples	72
5.6.1	Trees	72
5.6.2	Loops	73
5.6.3	A generic robotic mechanism	74
5.7	Summary	77

6	SimSOFT: a physics engine for soft robots	78
6.1	Description	78
6.2	Benchmark problems	80
6.3	Applications	90
6.3.1	Dynamic simulations in three scenarios: nuclear fusion, surgical, rehabilitation	90
6.3.2	Learning from simulation	105
6.3.3	Vibration control for remote procedures in challenging environments	109
6.4	Summary	114
	Conclusion	115
A	Lie group framework	116
A.1	Lie group fundamentals	116
A.2	The special Orthogonal group $SO(3)$	117
A.3	The special Euclidean group $SE(3)$	118
B	The exponential map	119
B.1	Exponential map fundamentals	119
B.2	The exponential map on $SO(3)$	120
B.3	The exponential map on $SE(3)$	121
C	Lie subgroup framework	122
C.1	Lie subgroup fundamentals	122
C.2	Subgroups of the special Orthogonal group $SO(3)$	123
C.3	Subgroups of the special Euclidean group $SE(3)$	123
D	Time integration on a Lie group	124
D.1	Implicit time integration on a Lie group	124
D.2	Time integration on $SO(3)$	126
D.3	Time integration on $SE(3)$	126

List of Figures

1	Classification of robots based on degrees of freedom. (Images: from left, © 2017 Universal Robots A/S; © 2017 Kuka Robotics Corp.; Choset’s Modsnake; © 2017 Hansen Medical Inc.)	2
1.1	Rigid body transformation	9
1.2	A screw axis \mathcal{S} represented by a point \mathbf{q} , a unit direction vector \mathbf{s} and a pitch p	12
2.1	Geometric description of the reference curve for the continuum arm.	23
2.2	Kinematics of the piecewise constant deformation model	26
2.3	Constant deformation soft arm in its initial and current configuration.	29
2.4	Geometry of the interpolated reference curve.	31
3.1	Continuum arm subject to a torque τ at its free end. Solid line: initial configuration. Dashed line: deformed configuration.	37
3.2	Cantilevered soft arm subject to a torque τ at its free end. The initial configuration is a straight vertical line, while the deformed configuration is a planar curve of constant curvature.	49
4.1	Schematic model of the free-rotating soft arm.	53
4.2	Tip displacements in x and z -directions for the free soft arm rotating at three different velocities w	58
5.1	A kinematic joint which connects two generic arms of a robotic manipulator.	62
5.2	Schematic representation of a revolute elastic joint.	66
5.3	A generic robotic mechanism. \mathbf{H}_I are nodal frames, $\mathbf{H}_{J,I}$ are relative frames and (\mathbf{H}_I) are optional frames.	71
5.4	A tree structured mechanism.	72
5.5	Motion of the central joint and node 4 of the tree structured mechanism.	73
5.6	A four-bar mechanism.	73
5.7	Motion of the third joint and node 2 of the loop structured mechanism.	74
5.8	Schematic model of a generic robotic mechanism composed by rigid and soft arms structured in a parallel and serial topology.	75

5.9	Motion of the free joint and the tip node of the generic manipulator. Blue: joint; red: x; green: y. Dotted: rigid joints; Dashdotted: adding of spring elements; Solid: adding of spring and damper elements.	76
6.1	Conceptual architecture of SimSOFT.	79
6.2	Displacements of the tip-position of the two-link rigid manipulator in the inertial reference frame.	80
6.3	Flexible double pendulum in its initial configuration.	81
6.4	Relative angular position of the revolute joint for the flexible double pendulum. Solid: 1D FEM, dashdotted: 3D FEM	82
6.5	The spatial flexible manipulator, constituted by two links with two active joints, one universal, one revolute.	83
6.6	Tip displacements of the spatial flexible manipulator for move 1 .	84
6.7	Tip displacements of the spatial flexible manipulator for move 2 .	84
6.8	Tip displacements of the spatial flexible manipulator for move 3 .	85
6.9	Schematic model of the Princeton experiment, a cantilevered arm subjected to the load P	86
6.10	Tip displacements of the Princeton cantilevered arm subjected to the end load P with three different intensities and seven configurations. Blue: P_1 , red: P_2 , green: P_3 . Scatter: experimental data, dotted: numerical predictions from SimSOFT.	87
6.11	Rod-driven, two-segment prototype from Rone and Ben-Tzvi. . .	88
6.12	Static equilibrium for the multisegment rod-driven continuum manipulator.	89
6.13	A qualitative picture of the DEMO vertical maintenance system architecture.	91
6.14	Hybrid kinematic mechanism for DEMO MMS remote handling .	92
6.15	Snapshot of the hybrid kinematic mechanism within SimSOFT . .	92
6.16	Kinematic model of the hybrid kinematic mechanism	93
6.17	Displacements of end-effector of the hybrid kinematic mechanism in the test trajectory.	98
6.18	Reaction forces of the hybrid kinematic mechanism at the boundaries with the vertical transport system interface, during the test trajectory. Red: 01cl; Green: 02cl; Blue: 03cl. Solid: RF; Dotted: FR.	99
6.19	Geometric data for the Magellan Robotic Catheter 10Fr	101
6.20	Schematic model (not in scale) of the the Magellan Robotic Catheter 10Fr in the configuration of minimum extension for the leader . .	102
6.21	The Magellan Robotic Catheter 10Fr in the configuration of minimum extension for the leader, as it appears in SimSOFT	102
6.22	Displacements of the Magellan Robotic Catheter 10Fr in the test simulation. As we can see, the trajectory induces a three-dimensional out-of-plane motion.	102
6.23	Example of soft bending actuators	103

6.24	Snapshot of the soft bending actuator in SimSOFT.	104
6.25	Tip displacements of the soft bending actuators along the test trajectory	104
6.26	Learned continuum arm model at different load conditions.	107
6.27	Learned continuum arm model at different rotating velocities.	108
6.28	Zero vibration input shaping process	109
6.29	Input predictive shaping process.	110
6.30	Nonlinear pendulum.	111
6.31	Experimental setup, the Georgia Tech bridge crane	111
6.32	Velocity commands for the ZV-IPS experiments.	113
6.33	ZV-IPS: comparison of experimental and simulated response.	113
6.34	Comparison of input shapers. Experimental data.	113

List of Tables

3.1	Geometric Jacobian of the piecewise constant deformation soft manipulator.	44
5.1	Joint definition.	62
5.2	Initial configuration and inertia properties in principal axes for the tree structured mechanism (position in [m], mass in [kg], rotation inertia in [kgm]).	72
5.3	Initial configuration and inertia properties in principal axes for the four bar mechanism (position in [m], mass in [kg], rotation inertia in [kgm]).	74
5.4	Initial configuration and inertia properties in principal axes for the generic robotic mechanism (position in [m], mass in [kg], rotation inertia in [kgm]).	75
5.5	Mass matrices for the generic robotic mechanism.	76
5.6	Stiffness matrices for the generic robotic mechanism.	76
6.1	Physical parameters for the two-link rigid manipulator (length in [m], mass in [kg], rotation inertia in [kg m ²]).	80
6.2	Initial configuration of the flexible double pendulum. Position in [m].	81
6.3	Kinematic joint definition of the flexible double pendulum.	81
6.4	Initial configuration of the spatial flexible manipulator. Measurements in [m].	82
6.5	Kinematic joint definition of the spatial flexible manipulator.	83
6.6	Joint initial and final positions for the three simulations. Measurements in [rad]	83
6.7	Cross-section properties of the Princeton experiment.	86
6.8	Cross-section properties of the central backbone of the multisegment rod-driven manipulator.	88
6.9	Inertia properties in principal axes for the two segments of the multisegment rod-driven manipulator (mass in [kg] and rotation inertia in [kgm])	89
6.10	Geometric linear data of the hybrid kinematic mechanism	93
6.11	Geometric angular data of the hybrid kinematic mechanism	94
6.12	Initial configuration of the hybrid kinematic mechanism	94

6.13	Inertia properties in principal axes for the rigid bodies of the hybrid kinematic mechanism (mass in [kg] and rotation inertia in [kgm])	94
6.14	Geometric data and material properties for the soft elements of the hybrid kinematic mechanism	95
6.15	Cross-section mass matrices for the soft elements of the hybrid kinematic mechanism	95
6.16	Cross-section stiffness matrices for the soft elements of the hybrid kinematic mechanism.	95
6.17	Kinematic joint definition for the hybrid kinematic mechanism . .	96
6.18	Initial configuration of the Magellan Robotic Catheter 10Fr	100
6.19	Cross-section mass matrices for the elements of the Magellan Robotic Catheter 10Fr	101
6.20	Cross-section stiffness matrices for the elements of the Magellan Robotic Catheter 10Fr.	101
6.21	Dataset for the constant deformation soft arm.	105
6.22	Dataset for the free-rotating soft arm.	106
6.23	Dataset for the ZV-IPS for a nonlinear pendulum.	111
6.24	ZV-IPS for a nonlinear pendulum	112

Introduction

Soft robots refer to robotic systems which exhibit an inherent compliant behavior. Their key component is represented by the continuously deformable *mechanical structure*, which provides them new capabilities relative to traditional robots. The most commonly used traditional robots are kinematically non-redundant. They have been used since many years to repetitively perform a prescribed motion in well-known environments. When a robot has more degrees of freedom (DOF) than the necessary to execute a task (e.g., a 7-DOF arm with a 6-DOF task space) it is said to be redundant, or, in extreme cases, hyper-redundant. Hyper-redundant robots have a large number of links and joints; they have the potential to work in unstructured environments and provide high dexterity. When the number of DOF approaches infinity, the robot approaches what is known as a continuum robot. The shape of a continuum robot is defined by an infinite-DOF elastic member. Since an elastic structure has a distributed deformation and it is inherently compliant, we use the term soft robots to indicate a infinite-DOF compliant robotic manipulator. When multiple infinite-DOF elastic structures are connected through joints in a generic kinematic chain, the soft robot approaches the continuum multibody structure. A qualitative picture on classification of robots based on the DOF is shown in Fig. 1. The configuration space of soft robots is theoretically infinite: the robot tip (tips) can reach every point (points) in the three-dimensional workspace with an infinite number of robot configurations. Exploiting large strain deformation, soft robots can adapt their shape to nonlinear path and squeeze through openings smaller than their nominal dimensions. All these features enable soft robots to perform delicate tasks in cluttered and/or unstructured environments, as well as to investigate novel grasping and manipulation possibilities. Further, the compliance of their underlying material makes them ideal for applications which require a safer physical human-robot interaction.

Mathematical models, i.e., idealized mathematical descriptions in the form of equations for real-world systems, play a crucial role in the soft robotics context. In this thesis we will present a unified mathematical framework for serial/parallel kinematic chains constituted by rigid/soft arms connected through rigid/flexible joints. In the remaining part of this Chapter we present an overview on the current formulations for multibody systems as well as for soft robotic arms. Then, we formulate the objectives that will be discussed in the thesis, as reported in the outline.

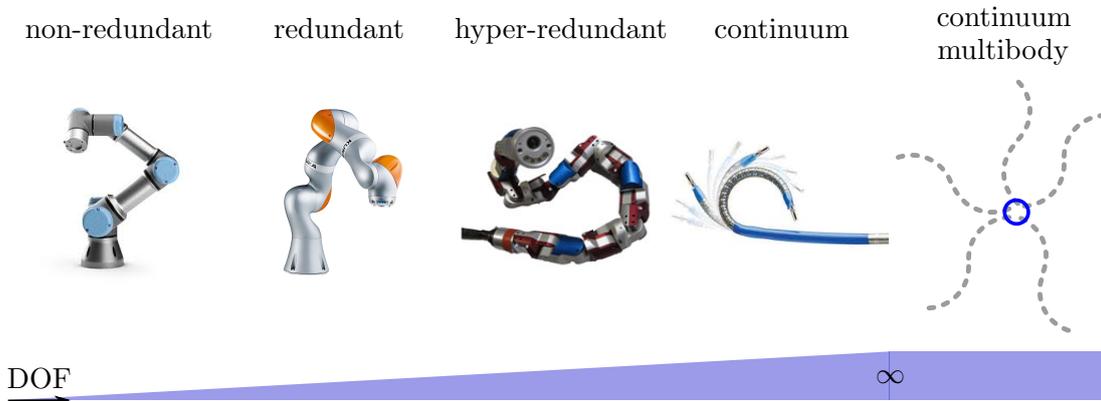


Figure 1: Classification of robots based on degrees of freedom. (Images: from left, © 2017 Universal Robots A/S; © 2017 Kuka Robotics Corp.; Choset’s Modsnake; © 2017 Hansen Medical Inc.)

Formulation for multibody systems

In the multibody dynamics community, many different formulations are available to describe the motion of a generic mechanical systems. Two classifications are mainly adopted for multibody systems, one based on *variables*, one based on *coordinate systems* [1].

Variables classification

Different variables can be chosen for the representation of large amplitude motions and connections between bodies.

- *Relative variables* [2]. The configuration of a dynamical system is expressed in a recursive way; the relative variables describe the relative configurations between the bodies as imposed by the joints. Usually, the relative variables are chosen to be the relative coordinates inside the joints, as the relative angle in a revolute joint or the relative displacement in a prismatic joint. For open-loop rigid topologies, this approach is appealing since a minimal number of variables is necessary to fully describe the system [3]. However, for closed-loop topologies, relative variables are no longer independent and complicated nonlinear algebraic constraint equations must be formulated to represent closure conditions.
- *Absolute variables* [4]. Each body of the system is regarded a-priori as a free body and its configuration is described by a set of variables expressed with respect to a fixed inertial reference frame. The absolute variables are redundant and a set of nonlinear algebraic constraint equations is introduced to describe the kinematic connections between the bodies. This approach

leads to a larger number of variables, but the system of equations of the model can be generated in a systematic way by an assembly procedure, is loosely coupled and its sparsity can be exploited for an efficient computer implementation.

- *Mixed variables.* The description of the system involves at the same time absolute and relative variables, which are related through algebraic constraints. The number of variables is increased, but it is still sparse.

Coordinate system classification

Different coordinate system, i.e., different coordinate frames, can be used to formulate the equations of motions.

- *Floating frame of reference* (FFR) formulation [2, 5, 6]. A floating frame is defined for each flexible component. The overall motion of a component is described by superimposing a local motion to a mean rigid body motion of a floating frame. The rigid body motion can be described using absolute or relative variables. The local motion is represented by additional local variables, which are usually referred to as modal amplitudes or elastic coordinates. The FFR approach reproduces exactly the rigid body motion, and it represents the natural way to extend rigid multibody dynamics to flexible multibody systems. One drawback of this formulation is that it is not able to account for geometrically nonlinear effects of deformations, such as the geometric stiffening occurring under fast rotations.
- *Corotational frame* (CF) formulation [7–10]. A corotational frame is defined for each (finite) element. The corotational frame of each element follows a mean rigid body motion of the component. This method is usually based on absolute nodal variables and aims at generalizing the geometrically linear finite element tools to the geometrically nonlinear case, i.e., including large amplitude motions.
- *Inertial frame* (IF) formulation [4, 11–14], also called geometrically exact (finite) element formulation. The global inertial reference frame is used as a reference frame for all motions. This approach is usually based on absolute variables, which include position and finite rotation. The exact representation of the rigid body motions is achieved by deriving frame invariant deformation measures from continuum mechanics. Since the motion of a component is viewed as a whole, without a priori decomposition into a mean rigid body and a superimposed flexible motion, the formulations based on the geometrically exact nonlinear beam theory are the most general and accurate to account for flexibility [15].

In the current work, we use a *geometrically exact finite element* formulation which makes use of *mixed variables*, namely absolute variables for the representation of large amplitude motion, and relative variables for the connection between bodies.

Framework for soft robotic arms

Two different frameworks are mainly adopted in the literature to model soft robotics arms, the kinematics and the mechanics frameworks [16].

Kinematics framework

A kinematic-based framework can follow one of the following approaches.

- *Discrete rigid links* [17, 18]. The simplest approach to approximate a continuous elastic structure is to consider a series of n rigid links connected by conventional revolute, universal or spherical joints.
- *Constant-curvature* [19], the most adopted approach in the soft robotics community. It represents the soft robotics arm as a finite collection of mutually tangent circular arcs, which can be described by only three parameters: radius of curvature, angle of the arc, bending plane. The configuration of the arm is full described by these three parameters. Notice that "constant" here refers to invariance with respect to arc length, not time. In this context, the homogeneous transformations along a constant curvature robotic arm have been derived from a variety of perspectives, such as Denavit-Hartenberg parameters [20, 21], and exponential coordinates [22, 23].
- *Variable-curvature* [24–27]. The configuration of the backbone, i.e. the position and the orientation of the infinite-DOF elastic member, is obtained as a function of the arc length along the robot's shape. The approach to model continuum robots using variable-curvature frameworks dates back to the works of Chirikjian on hyper-redundant serial manipulators [28–30].

Mechanics framework

Mechanics (or dynamics) frameworks can be combined with various kinematics frameworks to achieve a mechanics-based representation of the governing equations for the robot configuration.

- *Lumped parameters models* [31, 32]. They arise almost automatically as an extension of discrete-link kinematic frameworks, but they can also be imposed on top of constant and variable curvature frameworks. The approach consists of attaching discrete mechanical elements such as point masses, springs and dampers, to the kinematics framework in order to approximate the mechanical behavior of a continuous elastic and/or viscous element. The governing equations for lumped-parameters model can be obtained by energy methods or classical Newton-Euler equations describing how forces and moments propagate from link to link.

- *Energy methods* [33]. These methods belong to a powerful class of tools that have been used for a large variety of purposes in continuum robotics research. Energy minimization has been used to derive both constant and non-constant curvature models for concentric-tube robots [23, 25] and for multibackbone robots [34].
- *Classical elasticity theories* [11, 12]. Various elasticity theories for long slender bodies have been adopted to describe continuum robots. The widely used approach which involves the classical Bernoulli-Euler beam theory and thus, makes the assumption of small deflections, has been adopted to model flexible manipulators in the past years. The planar, large-deflection, Bernoulli-Euler elastica theory and its analytical solution in terms of elliptic functions have been used to describe the exact mechanics of planar robotic arms [35]. Timoshenko beam models have also been investigated to include shear effects [36]. A promising approach for modeling soft continuum robots comes from the Cosserat rod theory, which allows to include both shear and torsion effects. The theory makes no small-deflection geometric approximations and can accommodate any non-linear stress-strain relationship. The first attempt to use Cosserat rod theory to model continuum robots was by Trivedi et. al in [37]. Starting from the pioneer works of Simo and Vu-Quoc [11, 12], Sonnevile et. al recently developed a beam model formulated on a Lie group [38]. Similar governing equations have been derived by Chirikjian to model structures in robotics and molecular biology [39].

In the current work, we develop a novel *piecewise constant deformation* model for soft continuum manipulators in the context of the Cosserat rod theory which accounts for bending, shear and torsion effects, using the *screw theory* and the *finite element method*.

Objectives

The objective of the thesis is to propose a general theoretical framework for soft robots formulated on a Lie group. The framework leads to the following main derivations

1. Product-of-exponential formulae for soft manipulator arms.
2. Soft geometric Jacobians for soft manipulator arms.
3. Dynamic models for soft manipulator arms.
4. Dynamic models for robotic mechanisms including rigid and/or soft arms connected by rigid and/or flexible joints.

Outline

The manuscript is outlined as follows

- Chapter 1: The descriptions of the geometric formalism and the geometric methods used in the thesis are presented.
- Chapter 2: The kinematic formulation for a continuum arm and a piecewise constant deformation arm is derived on a Lie group.
- Chapter 3: The differential kinematics is formulated in terms of the Jacobians, and the kineto-statics duality is emphasized.
- Chapter 4: The dynamics is formulated using the Hamilton's principle, which leads to the well-known structure of the dynamic model for rigid manipulators.
- Chapter 5: The dynamics of hybrid mechanisms is formulated thanks to the definition of the joints.
- Chapter 6: The description, benchmarks and applications of SimSOFT, a physics engine for soft robots, are addressed.

Chapter 1

The geometric formalism

The use of *geometric techniques* in robotics is motivated by the fact that often the most salient physical features of a robot are best captured by a geometric description. The advantages of the geometric approach have been recognized for a while by practitioners of classical *screw theory*, which dates back to the early 1900s to the pioneering work of Ball [40]. However, the mostly algebraic alternatives to screw theory and the requirement of a new language of concepts and constructs (screws, twists, wrenches, ...) have slowed down a wide development in the robotics community. The breakthrough that made the techniques of classical screw theory accessible to a more general audience arrived in the early 1980s, when Roger Brockett showed how to describe kinematic chains mathematically in terms of the Lie group structure of rigid-body motions [41]. This discovery allowed re-inventing screw theory by appealing to the basic linear algebra and linear differential equations. With this *modern screw theory*, the powerful tools of *modern differential geometry* can be used to describe, in a simple and elegant manner, a wide range of robotics problems. This has led to the development of excellent textbooks by Murray et al. [42], Selig [43], and, recently, Lynch and Park [44] in the context of *rigid robotics*.

In general, bodies, as robotics arms, deform under applied loads and external conditions. Their physical behavior can be caught by using the classical tools of *continuum mechanics* for three-dimensional solids [45–47]. However, in robotics applications, we usually deal with solids in which one dimension is predominant over the two others. Thus, we can refer to the classical *beam theory*, such that the configuration of a body can be defined by considering two-dimensional cross-sections upon a three-dimensional reference curve. When the cross-section dimensions are small compared to the length of the reference curve, the following kinematic assumption can be formulated to describe the behaviour of the original three-dimensional solid: "*the cross-sections behave as rigid bodies and, as a consequence, remain flat.*" This assumption allows the use of the powerful geometric techniques in the context of **soft robotics**: *a soft robotic arm is the result of the different rigid body transformations of the cross-sections.*

As result, the main concept of rigid body transformations might be applied both to rigid and soft robotics arms. Using a *finite element* terminology, a rigid body is described by one node, while a soft body is described by a finite set of nodes which moves upon a shape function which approximate the original three-dimensional reference curve.

In this chapter we start by introducing the concept of rigid body transformations, represented by frame transformations which belongs to the special Euclidean group $SE(3)$. For the mathematical description of frames, the reader can refer to the abstract concepts of Lie group and Lie algebra, explained in Appendix A. In this chapter we only point out the geometrical interpretations. After, the notion of local frame representation of derivatives is introduced. This concept will be relevant in all the thesis, since it will allow the derivation of equations of motion based on intrinsic quantities. As illustrative example of the geometric techniques, the equations of motion of a rigid body are derived using the Hamilton's principle of mechanics. Further, geometric time integration methods are presented to obtain global parametrization-free equations of motion.

1.1 Rigid body transformations

The motion of a particle moving in Euclidean space is described by giving the location of the particle at each instant of time, relative to an inertial Cartesian reference frame. Let us consider a vector $\mathbf{p} \in \mathbb{R}^3$ to denote the position vector of a material point p of a body with respect to the fixed reference frame $(\mathcal{I}, \mathbf{o})$. The configuration of the body in the fixed reference frame is called the *reference configuration*. Let us observe the motion of the point towards another configuration, in which the position of the point p with respect to the fixed reference frame is denoted $\mathbf{q} = g(\mathbf{p})$. We call the novel configuration of the body the *current configuration*. The position vectors of the current configuration can be expressed as *transformations* of the same position vectors of the reference configuration. A *rigid-body transformation* is a special class of transformation which is defined as follows.

Definition 1 *A mapping $g : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ is a rigid body transformation if it satisfies the following two properties:*

- *Distance is preserved, i.e., $\|g(\mathbf{p}_j) - g(\mathbf{p}_i)\| = \|\mathbf{p}_j - \mathbf{p}_i\| \forall \mathbf{p}_i, \mathbf{p}_j \in \mathbb{R}^3$*
- *Cross product is preserved, i.e., $u(\mathbf{a} \times \mathbf{b}) = g(\mathbf{a}) \times g(\mathbf{b}) \in \mathbb{R}^3$*

A rigid motion preserves the distance between points and the angle between vectors. This means that rigid-body transformations map right-handed, orthonormal frames to right-handed, orthonormal frames.

Definition 1 leads to the following proposition, which is illustrated in Fig. 1.1

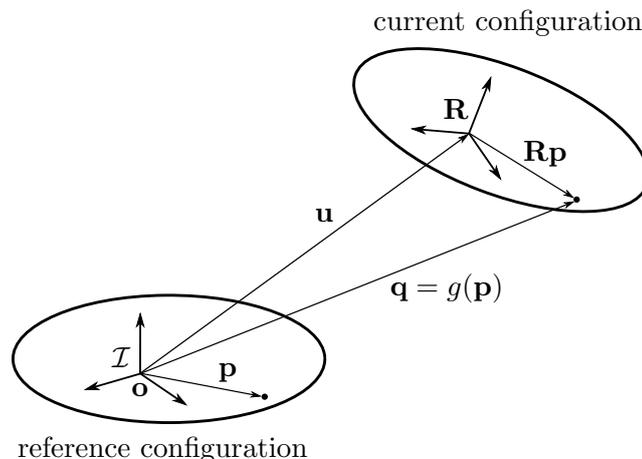


Figure 1.1: Rigid body transformation

Proposition 1 *Rigid body transformations are such that the position vector of any material point p in the current configuration is given by the affine mapping*

$$\mathbf{q} = \mathbf{u} + \mathbf{R}\mathbf{p} \quad (1.1)$$

where $\mathbf{u} \in \mathbb{R}^3$ is the current position vector of the reference point and $\mathbf{R} \in SO(3)$, which characterizes the orientation of the body in the current configuration with respect to the reference frame \mathcal{I} , is a rotation matrix, i.e., a 3×3 matrix which satisfies

$$\mathbf{R}^T \mathbf{R} = \mathbf{I}_{3 \times 3} \quad (1.2)$$

$$\det(\mathbf{R}) = +1 \quad (1.3)$$

The rigid body transformation from the reference configuration to the current configuration is fully represented by a rotation matrix \mathbf{R} and a position vector \mathbf{u} . This transformation is represented by six parameters: three are related to \mathbf{u} , while three are related to \mathbf{R} . Indeed, the nine components of a rotation matrix are subject to six independent nonlinear orthonormality constraints in the form $\mathbf{R}^T \mathbf{R} = \mathbf{I}_{3 \times 3}$.

The special Orthogonal group $SO(3)$ is the matrix Lie group which describes the rotational motion in \mathbb{R}^3 . This group is fully described in [A.2](#).

1.2 Homogeneous transformation matrices

The rigid body transformation in Eq. 1.1 can be represented in matrix form by using the homogeneous representation of vector as

$$\begin{bmatrix} \mathbf{q} \\ 1 \end{bmatrix} = \mathcal{H}(\mathbf{R}, \mathbf{u}) \begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix} \quad (1.4)$$

where the 4×4 matrix

$$\mathbf{H} = \mathcal{H}(\mathbf{R}, \mathbf{u}) = \begin{bmatrix} \mathbf{R} & \mathbf{u} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \quad (1.5)$$

contains the whole information about the current configuration. The general rigid body motion is a combination of rotation and translation. The homogeneous transformation matrix $\mathbf{H} = \mathcal{H}(\mathbf{R}, \mathbf{u}) \in SE(3)$ allows representing a general rigid body motion in a simple and elegant manner.

The special Euclidean group $SE(3)$ is the matrix Lie group which describes the rigid body motion in \mathbb{R}^3 . This group is fully described in [A.3](#).

1.3 Twists

Let us consider the derivative of the homogeneous transformation matrix in Eq. 1.5 as

$$\dot{\mathbf{H}} = \mathbf{H}\tilde{\boldsymbol{\eta}}_L \quad (1.6)$$

$$= \tilde{\boldsymbol{\eta}}_R \mathbf{H} \quad (1.7)$$

where $\tilde{\boldsymbol{\eta}}_L \in \mathfrak{se}(3)$ and $\tilde{\boldsymbol{\eta}}_R \in \mathfrak{se}(3)$ are respectively called the left and the right invariant vector field. The element $\tilde{\boldsymbol{\eta}} \in \mathfrak{se}(3)$ is the Lie algebra associated to the Lie group $\mathbf{H} \in SE(3)$. In the screw theory, the Lie algebra $\tilde{\boldsymbol{\eta}} \in \mathfrak{se}(3)$ is called *twist*. It has the following expression

$$\tilde{\boldsymbol{\eta}} = \begin{bmatrix} \tilde{\boldsymbol{\omega}} & \mathbf{v} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \quad (1.8)$$

where the Lie algebra $\tilde{\boldsymbol{\omega}} \in \mathfrak{so}(3)$ is the skew-symmetric matrix

$$\tilde{\boldsymbol{\omega}} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \quad (1.9)$$

which allows the description of the angular velocity associated to a rigid body transformation.

According to the isomorphism $\mathfrak{g} \simeq \mathbb{R}^k$, $\mathfrak{so}(3)$ is isomorphic to \mathbb{R}^3 with $\boldsymbol{\omega} = [\omega_1 \ \omega_2 \ \omega_3]^T$, while $\mathfrak{se}(3)$ is isomorphic to \mathbb{R}^6 with

$$\boldsymbol{\eta} = \begin{bmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{bmatrix} \quad (1.10)$$

If $\boldsymbol{\eta}_a$ is the expression of the velocity vector in the current configuration $\{a\}$, the expression of the same velocity vector in another frame $\{b\}$, whose transformation matrix with respect to the current configuration is given by the homogeneous transformation matrix $\mathbf{H}_{ba} = \mathcal{H}(\mathbf{R}_{ba}, \mathbf{u}_{ba}) \in SE(3)$, is given by

$$\boldsymbol{\eta}_b = \begin{bmatrix} \mathbf{R}_{ba} & \tilde{\mathbf{u}}_{ba} \mathbf{R}_{ba} \\ \mathbf{0}_{3 \times 3} & \mathbf{R}_{ba} \end{bmatrix} \boldsymbol{\eta} = \text{Ad}_{\mathbf{H}_{ba}}(\boldsymbol{\eta}_a) \quad (1.11)$$

$$\tilde{\boldsymbol{\eta}}_b = \mathbf{H}_{ba} \tilde{\boldsymbol{\eta}} \mathbf{H}_{ba}^{-1} = \text{Ad}_{\mathbf{H}_{ba}}(\tilde{\boldsymbol{\eta}}_a) \quad (1.12)$$

where Ad is the Adjoint representation defined in Eq. A.4. For a deeper understanding of the invariant vector fields, the reader can refer to Appendix A.

1.4 Screws: a geometrical description of twists

A twist $\tilde{\boldsymbol{\eta}}$ can be interpreted in terms of a *screw axis* \mathcal{S} and a velocity $\dot{\theta}$ about the screw axis. A screw axis represents the familiar motion of a screw: rotating about the axis while also translating along the axis. We can represent a screw axis \mathcal{S} as the collection $\{\mathbf{q}, \mathbf{s}, p\}$, where $\mathbf{q} \in \mathbb{R}^3$ is any point on the axis, \mathbf{s} is a unit vector in the direction of the axis, and p is the *screw pitch*, which defines the ratio of the linear velocity along the screw axis to the angular velocity $\dot{\theta}$ about the screw axis.

For a physical interpretation of the screw axis, see Fig. 1.2. Using this Figure, we can write the twist $\boldsymbol{\eta}$ corresponding to an angular velocity $\dot{\theta}$ about \mathcal{S} as

$$\boldsymbol{\eta} = \begin{bmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} -\mathbf{s}\dot{\theta} \times \mathbf{q} + p\mathbf{s}\dot{\theta} \\ \mathbf{s}\dot{\theta} \end{bmatrix} \quad (1.13)$$

Notice that the linear velocity \mathbf{v} is the sum of two terms: one related to the translation along the screw axis, $p\mathbf{s}\dot{\theta}$, and one related to the linear motion at the origin induced by the rotation about the axis, $-\mathbf{s}\dot{\theta} \times \mathbf{q}$. The first term is in the direction of \mathbf{s} , while the second term is in the plane orthogonal to \mathbf{s} .

One of the possibilities to define a screw axis \mathcal{S} is to define it as the normalized version of the twist corresponding to the motion along the screw: in this case we have $\boldsymbol{\eta} = \mathcal{S}\dot{\theta}$ and $\tilde{\boldsymbol{\eta}} = \tilde{\mathcal{S}}\dot{\theta}$, where \mathcal{S} and $\tilde{\mathcal{S}}$ are defined by

$$\mathcal{S} = \begin{bmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{bmatrix} \in \mathbb{R}^6 \quad (1.14)$$

$$\tilde{\mathcal{S}} = \begin{bmatrix} \tilde{\boldsymbol{\omega}} & \mathbf{v} \\ \mathbf{0}_{1 \times 3} & 0 \end{bmatrix} \in \mathfrak{se}(3) \quad (1.15)$$

As for the twists, it holds $\mathcal{S}_b = \text{Ad}_{\mathbf{H}_{ba}}(\mathcal{S}_a)$ and $\tilde{\mathcal{S}}_b = \text{Ad}_{\mathbf{H}_{ba}}(\tilde{\mathcal{S}}_a)$.

1.5 Exponential coordinate representation

The *Chasles-Mozzi theorem* states that every rigid body motion can be realized by a rotation about an axis combined with a translation parallel to that axis.

In this Section, the objective is to find the *exponential coordinates of a homogeneous transformation matrix*, i.e., the exponential coordinates associated with a

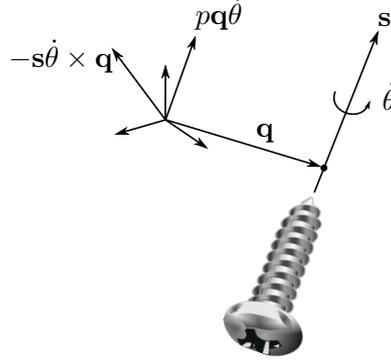


Figure 1.2: A screw axis \mathcal{S} represented by a point \mathbf{q} , a unit direction vector \mathbf{s} and a pitch p .

rigid body motion. Let us start by considering Eq. 1.6 using the left invariant vector field as

$$\dot{\mathbf{H}} = \mathbf{H}\tilde{\boldsymbol{\eta}} \quad (1.16)$$

Equation 1.16 can be seen as a linear differential equation on a Lie group. If $\tilde{\boldsymbol{\eta}}$ does not depend on t , the solution is given by

$$\mathbf{H}(t) = \mathbf{H}_0 \exp_{SE(3)}(\tilde{\boldsymbol{\eta}}t) \quad (1.17)$$

where \mathbf{H}_0 is a constant of integration and $\exp_{SE(3)}(\cdot)$ is the exponential map on $SE(3)$, which maps an element of the Lie algebra $\tilde{\boldsymbol{\eta}} \in \mathfrak{se}(3)$ into an element of the Lie group $\mathbf{H} \in SE(3)$. The exponential operator is defined in Eq. B.3. The exponential map on $SE(3)$ is given by

$$\exp_{SE(3)}(\boldsymbol{\eta}) = \begin{bmatrix} \exp_{SO(3)}(\boldsymbol{\omega}) & \mathbf{T}_{SO(3)}^T(\boldsymbol{\omega})\mathbf{v} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \quad (1.18)$$

where \mathbf{T} is the tangent operator defined in Eq. B.8 and $\exp_{SO(3)}(\cdot)$ is the exponential map on $SO(3)$, which is given by

$$\exp_{SO(3)}(\boldsymbol{\omega}) = \mathbf{I}_{3 \times 3} + \alpha(\boldsymbol{\omega})\tilde{\boldsymbol{\omega}} + \frac{\beta(\boldsymbol{\omega})}{2}\tilde{\boldsymbol{\omega}}^2 \quad (1.19)$$

with

$$\alpha(\boldsymbol{\omega}) = \frac{\sin(\|\boldsymbol{\omega}\|)}{\|\boldsymbol{\omega}\|} \quad \beta(\boldsymbol{\omega}) = 2\frac{1 - \cos(\|\boldsymbol{\omega}\|)}{\|\boldsymbol{\omega}\|^2} \quad (1.20)$$

Equation 1.19 is known as Rodriguez' formula for rotations.

The exponential map on $SE(3)$ provides a constructive proof of the *Chasles-Mozzi theorem*. That is, given an arbitrary $\mathbf{H} = \mathcal{H}(\mathbf{R}, \mathbf{u})$, one can always find a twist vector $\tilde{\boldsymbol{\eta}}$ such that

$$\exp_{SE(3)}(\tilde{\boldsymbol{\eta}}) = \begin{bmatrix} \mathbf{R} & \mathbf{u} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \quad (1.21)$$

where

$$\tilde{\boldsymbol{\eta}} = \log_{SE(3)}(\mathcal{H}(\mathbf{R}, \mathbf{u})) \quad (1.22)$$

being $\log_{SE(3)}(\cdot)$ the logarithmic map on $SE(3)$, which is defined as

$$\log_{SE(3)}(\mathcal{H}(\mathbf{R}, \mathbf{u})) = \begin{bmatrix} \tilde{\boldsymbol{\omega}} & \mathbf{T}_{SO(3)}^{-T}(\boldsymbol{\omega})\mathbf{u} \\ \mathbf{0}_{1 \times 3} & 0 \end{bmatrix} \quad (1.23)$$

where \mathbf{T}^{-1} is the inverse of the tangent operator defined in Eq. B.10 and $\tilde{\boldsymbol{\omega}} = \log_{SO(3)}(\mathbf{R})$, with

$$\log_{SO(3)}(\mathbf{R}) = \frac{\theta}{2\sin\theta}(\mathbf{R} - \mathbf{R}^T) \quad (1.24)$$

being

$$\theta = \arccos\left(\frac{1}{2}(\text{trace}(\mathbf{R}) - 1)\right), \quad \theta < \pi \quad (1.25)$$

The geometric interpretation of this result is that the rotation matrix \mathbf{R} is equivalent to a rotation of the angle θ about the axis given by $\boldsymbol{\omega}$. This is known as the *angle and axis* nonminimal representation of the orientation.

1.6 Wrenches

As twists merge the linear and angular components of velocities, we can merge also forces and moments into a single six-dimensional vector called *wrench* as

$$\boldsymbol{\tau} = \begin{bmatrix} \mathbf{f} \\ \mathbf{m} \end{bmatrix} \in \mathbb{R}^6 \quad (1.26)$$

Let us recall that the dot product of a force and a velocity acting on a rigid body is a power, and power is a coordinate-independent quantity, we have

$$\boldsymbol{\eta}_b^T \boldsymbol{\tau}_b = \boldsymbol{\eta}_a^T \boldsymbol{\tau}_a \quad (1.27)$$

By inserting (1.11) in Eq. 1.27, we obtain

$$(\text{Ad}_{\mathbf{H}_{ba}} \boldsymbol{\eta}_a)^T \boldsymbol{\tau}_b = \boldsymbol{\eta}_a^T \boldsymbol{\tau}_a \quad (1.28)$$

Since (1.28) must hold for all $\boldsymbol{\eta}_a$, we achieve the result

$$\boldsymbol{\tau}_b = \text{Ad}_{\mathbf{H}_{ba}}^{-T} \boldsymbol{\tau}_a \quad (1.29)$$

stating that the relationship between two wrenches expressed in two coordinate frames is given by the inverse-transpose of the Adjoint representation.

1.7 Equations of motion of a rigid body

As illustrative example, in this section we derive the equations of motion of a rigid body using the left invariant vector field for the derivatives. This approach, as pointed in [48], leads to intrinsic equations, i.e., equations which do not depend on the position or the orientation of the bodies with respect to the inertial reference frame.

1.7.1 Variations

The *variations* are used extensively in the calculus of variations [49] to derive the equations of mechanics. The variation of a rotation matrix is given by

$$\delta(\mathbf{R}) = \mathbf{R}\widetilde{\delta\boldsymbol{\theta}} \quad (1.30)$$

where $\widetilde{\delta\boldsymbol{\theta}} \in \mathfrak{so}(3)$ is an arbitrary infinitesimal rotation associated with the axial vectors $\delta\boldsymbol{\theta} \in \mathbb{R}^3$. By using Eq. A.8, the commutativity of the cross derivatives reads

$$\delta(\widetilde{\boldsymbol{\omega}}) - (\widetilde{\delta\boldsymbol{\theta}})^\cdot = [\widetilde{\boldsymbol{\omega}}, \widetilde{\delta\boldsymbol{\theta}}] = \widetilde{\boldsymbol{\omega}}\widetilde{\delta\boldsymbol{\theta}} - \widetilde{\delta\boldsymbol{\theta}}\widetilde{\boldsymbol{\omega}} \quad (1.31)$$

which can be reformulated, according to (A.9), as

$$\delta(\boldsymbol{\omega}) - (\delta\boldsymbol{\theta})^\cdot = \widehat{\boldsymbol{\omega}}\delta\boldsymbol{\theta} - \widehat{\delta\boldsymbol{\theta}}\boldsymbol{\omega} \quad (1.32)$$

The variation of a frame, following (A.23), is given by

$$\delta(\mathbf{H}) = \mathbf{H}\widetilde{\delta\mathbf{h}} = \mathbf{H}\mathcal{V}(\widetilde{\delta\boldsymbol{\theta}}, \delta\mathbf{h}_u) \quad (1.33)$$

where $\delta\mathbf{h}_u = \mathbf{R}^T\delta\mathbf{u} \in \mathbb{R}^3$ is an arbitrary infinitesimal displacement expressed in the local frame; the six-dimensional vector associated to $\widetilde{\delta\mathbf{h}}$ is given by

$$\delta\mathbf{h} = \begin{bmatrix} \delta\mathbf{h}_u \\ \delta\boldsymbol{\theta} \end{bmatrix} \quad (1.34)$$

As for the rotation, the variation of a twist element is given by

$$\delta(\widetilde{\boldsymbol{\eta}}) - (\widetilde{\delta\mathbf{h}})^\cdot = [\widetilde{\boldsymbol{\eta}}, \widetilde{\delta\mathbf{h}}] = \widetilde{\boldsymbol{\eta}}\widetilde{\delta\mathbf{h}} - \widetilde{\delta\mathbf{h}}\widetilde{\boldsymbol{\eta}} \quad (1.35)$$

and it can be reformulated as

$$\delta(\boldsymbol{\eta}) - (\delta\mathbf{h})^\cdot = \widehat{\boldsymbol{\eta}}\delta\mathbf{h} - \widehat{\delta\mathbf{h}}\boldsymbol{\eta} \quad (1.36)$$

where $\widehat{\boldsymbol{\eta}}$ and $\widehat{\delta\mathbf{h}}$ are the 6×6 matrices

$$\widehat{\boldsymbol{\eta}} = \begin{bmatrix} \widetilde{\boldsymbol{\omega}} & \widetilde{\mathbf{v}} \\ \mathbf{0}_{3 \times 3} & \widetilde{\boldsymbol{\omega}} \end{bmatrix}; \quad \widehat{\delta\mathbf{h}} = \begin{bmatrix} \widetilde{\delta\boldsymbol{\theta}} & \widetilde{\delta\mathbf{h}_u} \\ \mathbf{0}_{3 \times 3} & \widetilde{\delta\boldsymbol{\theta}} \end{bmatrix} \quad (1.37)$$

1.7.2 Kinetic energy

The kinetic energy \mathcal{K} of a rigid body is the quadratic form

$$\mathcal{K} = \frac{1}{2} \int_V \rho \dot{\mathbf{q}}^T \mathbf{q} dV \quad (1.38)$$

where V and ρ are respectively the volume and the mass density of the body. Considering that the velocity of a material point can be written as

$$\dot{\mathbf{q}} = \mathbf{R}(\mathbf{u} + \tilde{\boldsymbol{\omega}}\mathbf{p}) = \mathbf{R} [\mathbf{I}_{3 \times 3} - \tilde{\mathbf{p}}] \boldsymbol{\eta} \quad (1.39)$$

The kinetic energy can be expressed as

$$\mathcal{K} = \frac{1}{2} \boldsymbol{\eta}^T \mathbf{M} \boldsymbol{\eta} \quad (1.40)$$

where, $\forall \mathbf{a} \in \mathbb{R}^3$,

$$m = \int_V \rho dV; \quad \mathbf{J}_1(\mathbf{a}) = \int_V \rho \tilde{\mathbf{a}} dV; \quad \mathbf{J}_2(\mathbf{a}) = \int_V \rho \tilde{\mathbf{a}}^T \tilde{\mathbf{a}} dV \quad (1.41)$$

the mass matrix is given by

$$\mathbf{M} = \begin{bmatrix} m \mathbf{I}_{3 \times 3} & \mathbf{J}_1^T(\mathbf{p}) \\ \mathbf{J}_1(\mathbf{p}) & \mathbf{J}_2(\mathbf{p}) \end{bmatrix} \quad (1.42)$$

and it does not depend on the position and the orientation of the body. This is a direct consequence of the local frame interpretation of $\boldsymbol{\eta}$. This will also results in the invariance property of the inertia forces.

Classically, the reference configuration is defined such that the reference point and the reference axes imply that \mathbf{J}_1 vanishes and \mathbf{J}_2 is diagonal. In the following, we denote $\mathbf{J}_2(\mathbf{p})$ as \mathbf{J} . This must not be confused with \mathbf{J} or \mathbb{J} , the soft geometric Jacobian that will be derived in Chapter 3.

1.7.3 Potential energy

The potential energy given by the external forces \mathcal{V}_{ext} of a rigid body is defined as

$$\mathcal{V}_{ext} = \int_V \mathbf{q}^T \mathbf{g}_e dV \quad (1.43)$$

where \mathbf{g}_e is a 3×1 vector of applied external forces expressed in the fixed reference frame.

1.7.4 Hamiltonian formulation

The dynamic equilibrium equations of the continuum arm can be obtained from Hamilton's principle, which states that the action integral over the time interval

$[t_0, t_1]$ is stationary provided that the initial and final configurations are fixed, *i.e.*,

$$\int_{t_0}^{t_1} (\delta(\mathcal{K}) + \delta(\mathcal{V}_{ext})) dt = 0. \quad (1.44)$$

where the variations are fixed at t_0 and t_1 . The variation of the kinetic energy results

$$\delta(\mathcal{K}) = ((\delta\mathbf{h}^T)^\cdot + \delta\mathbf{h}^T \hat{\boldsymbol{\eta}}^T) \mathbf{M}\boldsymbol{\eta} \quad (1.45)$$

The variation of the external potential energy can be written as

$$\delta(\mathcal{V}_{ext}) = \int_V \delta(\mathbf{q})^T \mathbf{g}_e dV \quad (1.46)$$

where

$$\delta(\mathbf{q}) = \mathbf{R} [\mathbf{I}_{3 \times 3} - \tilde{\mathbf{q}}] \delta\mathbf{h} \quad (1.47)$$

such that

$$\delta(\mathcal{V}_{ext}) = \delta\mathbf{h}^T \mathbf{g}_{ext} \quad (1.48)$$

where

$$\mathbf{g}_{ext} = \begin{bmatrix} \mathbf{g}_{ext,u} \\ \mathbf{g}_{ext,\omega} \end{bmatrix} = \int_V \begin{bmatrix} \mathbf{I}_{3 \times 3} \\ \tilde{\mathbf{p}} \end{bmatrix} \dot{\mathbf{R}}^T \mathbf{g}_e dV \quad (1.49)$$

According to the interpretation of $\delta\mathbf{h}$, \mathbf{g}_{ext} takes the meaning of the external forces expressed in the local frame. Inserting (1.45) and (1.49) in Eq. (1.44) and integrating by parts lead to

$$[\delta\mathbf{h}^T (\mathbf{M}\boldsymbol{\eta})]_{t_0}^{t_1} - \int_{t_0}^{t_1} \delta\mathbf{h}^T (\mathbf{M}\dot{\boldsymbol{\eta}} - \hat{\boldsymbol{\eta}}^T \mathbf{M}\boldsymbol{\eta} - \mathbf{g}_{ext}) dt = 0 \quad (1.50)$$

Since the equations are valid $\forall \delta\mathbf{h}$, the equations of motion are given by

$$\dot{\mathbf{H}} = \mathbf{H}\tilde{\boldsymbol{\eta}} \quad (1.51)$$

$$\mathbf{M}\dot{\boldsymbol{\eta}} - \hat{\boldsymbol{\eta}}^T \mathbf{M}\boldsymbol{\eta} = \mathbf{g}_{ext} \quad (1.52)$$

where Eq. 1.51 are kinematic equations which relate the velocity representation and the time derivative of the frame representing the rigid body, and Eq. 1.52 are six dynamic equilibrium equations, three for the translational part and three for the rotational part. According to the nature of $\boldsymbol{\eta}$ and $\delta\mathbf{h}$, these equilibrium equations are written in the local frame attached to the body.

In case of no external loads, Eq. 1.52 depends only on the left representation of the velocities and are therefore independent of the configuration of the rigid body. For this reason, they are coined as intrinsic equations. In other words, (1.51) and (1.52) are decoupled and Eq. 1.52 can be solved for $\boldsymbol{\eta}$. The configuration can be obtained afterwards by solving Eq. 1.51.

In case of external loads, let us consider two representative practical load cases.

- a follower moment as

$$\mathbf{g}_{ext} = \begin{bmatrix} \mathbf{0}_{3 \times 1} \\ \mathbf{m} \end{bmatrix} \quad (1.53)$$

where \mathbf{m} is the 3×1 vector of the resulting moments of the distributed forces. Under such loadingm Eq. 1.51 and Eq. 1.52 are still uncoupled.

- the gravity field, a force with a constant direction with respect to the fixed reference frame given by

$$\mathbf{g}_e = \rho g \mathbf{e} \quad (1.54)$$

where g is the gravitational acceleration and \mathbf{e} is a constant unit vector. Accordingly, we have

$$\mathbf{g}_{ext} = \begin{bmatrix} mg \mathbf{R}^T \mathbf{e} \\ \mathbf{0}_{3 \times 1} \end{bmatrix} \quad (1.55)$$

where the choice of the reference frame as $\mathbf{J}_1(\mathbf{p}) = \mathbf{0}_{3 \times 3}$ leads to a vanishing resulting moment. Notice that, due to the expression in the local frame, $\mathbf{g}_{ext,u}$ depends on the rotation of the rigid body. This case leads to a coupling of Eq. 1.51-1.52 and to a nonlinear external force.

1.8 Geometric time integration

This section introduces the problem of time integration for the numerical solution of the equations of motion using a geometric approach. This is not a trivial problem since frames, and in particular rotation variables, belong to a nonlinear and non commutative space.

The *classical approach* to solve numerically the equations of motion of a dynamical system formulated with rotation variables is to introduce a *global parametrization* of the motion, and in particular of the rotation, i.e., expressing the motion variables in terms of specific coordinates with respect to the reference frame. Although this procedure allows the convenient use of *classical vector space solvers*, it suffers from some drawbacks. This approach is discussed in Section 1.8.1.

A *modern approach* to solve numerically the equations of motion of a dynamical system involves the use of *geometric methods* which do not require a *global parametrization* of the motion. For using such methods, the dynamics of the system must be formulated on a Lie group. The Lie group structure of the configuration space allows preserving the structure of the original equations of motion, which do not suffer from any of the drawbacks inherent to the global parametrization process.

In the context of geometric time integration, two modern approaches exist to solve differential equations directly on the Lie groups, without any *a priori* definition of generalized coordinates: the method of Crouch and Grossman [50] and the method of Munthe-Kaas [51]. They are both based on the use of the exponential map. The Crouch and Grossman idea is to use the exponential map as the solution of the differential equations of a frozen velocity field on the

Lie group. The Munthe-Kaas method uses the exponential map to construct a local coordinate chart at each time step. In this work, we use the implicit generalized- α -scheme proposed in [52,53], which follows the Crouch and Grossman ideas. This method has a proven second-order convergence rate and some numerical damping can be used to lessen high frequency content. Furthermore, it exhibits a reduced complexity because it involves purely geometric quantities. The implicit generalized- α -scheme is reported in Appendix D. Obviously, other schemes from the literature can be used, as [54,55]

1.8.1 Global parametrization of rotation

In this Section we illustrate the main drawbacks of a classical approach, through an illustrative example. The example is of a rotating rigid body without translational motion, such that a single 3D rotation variable is involved, and we use the left representation of the derivative only, i.e., $\boldsymbol{\omega}_L$.

The equations of motion of a single rotating body reduce to the Euler's equations

$$\dot{\mathbf{R}} = \mathbf{R}\tilde{\boldsymbol{\omega}} \quad (1.56)$$

$$\mathbf{J}\dot{\boldsymbol{\omega}} + \tilde{\boldsymbol{\omega}}^T \mathbf{J}\boldsymbol{\omega} = \mathbf{0}_{3 \times 1} \quad (1.57)$$

Equations 1.57 are first order nonlinear ordinary differential equations for the vectorial quantity $\boldsymbol{\omega}$, while Eq. 1.56 are first order differential equations for the rotation matrix $\mathbf{R} \in SO(3)$, which cannot be solved using classical vector space methods since $SO(3)$ is not a vector space.

A first classical family of methods to solve (1.56)–(1.57) is to introduce a *global minimal-parametrization* of the rotation, i.e., expressing the 3×3 matrix \mathbf{R} in terms of a certain set of 3 parameters only, such that for any value of the parameters, it is guaranteed that \mathbf{R} is a rotation matrix. Formally, we have

$$\mathbf{R} = \mathbf{R}(\alpha_1, \alpha_2, \alpha_3) \quad (1.58)$$

where $\alpha_i, i = 1, 2, 3$ are three rotation parameters, which can be collected in a 3×1 vector $\boldsymbol{\alpha}$. Many different parametrizations are possible, as Euler angles or the conformal rotation vector; see e.g. [4,13,56] for an extensive description of the possible parametrizations. Deriving Eq. 1.58 with respect to time leads to the expression of the rotational velocities in terms of the time derivative of the parameters. Formally, it reads

$$\boldsymbol{\omega} = \mathbf{T}(\boldsymbol{\alpha})\dot{\boldsymbol{\alpha}} \quad (1.59)$$

where $\mathbf{T}(\boldsymbol{\alpha})$ is a 3×3 matrix whose expression depends on the particular parametrization. Introducing (1.58) and (1.59) in (1.56)–(1.57), we obtain the parametrized equations of motion as

$$\mathbf{R} = \mathbf{R}(\boldsymbol{\alpha}) \quad (1.60)$$

$$\mathbf{T}(\boldsymbol{\alpha})^T \left(\mathbf{J}_2(\mathbf{T}(\boldsymbol{\alpha})\ddot{\boldsymbol{\alpha}} + \dot{\mathbf{T}}(\boldsymbol{\alpha}, \dot{\boldsymbol{\alpha}})\dot{\boldsymbol{\alpha}}) + (\mathbf{T}(\boldsymbol{\alpha})\dot{\boldsymbol{\alpha}})^\sim \mathbf{J}_2 \mathbf{T}(\boldsymbol{\alpha})\dot{\boldsymbol{\alpha}} \right) = \mathbf{0}_{3 \times 1} \quad (1.61)$$

Equations 1.56 are now replaced by the trivial equations in (1.60), while Eq. 1.57 is now replaced by the three second-order ordinary differential parametrized equilibrium equations in (1.61), for the three unknowns parameters in $\boldsymbol{\alpha}$. These equations can now be solved using classical integration methods for vector space since $\boldsymbol{\alpha} \in \mathbb{R}^3$.

Unfortunately, the global minimal-parametrization of the rotation suffers from the following drawbacks:

- Singularity. There exists no set of 3 parameters which does not lead to a singularity, i.e., there is always a configuration in which changing one of the parameters does not change the rotation matrix.
- High nonlinearities. The expression of the rotation matrix (1.58) and the rotational velocities (1.59) in terms of the parameters are always nonlinear and often involve transcendental functions. As a consequence, the parametrized equilibrium equations are highly nonlinear due to the presence of \mathbf{T} and $\dot{\mathbf{T}}$, as opposed to the original equilibrium equations in Eq. 1.57 which were only quadratic.
- Dependency on the orientation of the body. The parametrized equilibrium equations depend on the orientation of the body, since \mathbf{T} and $\dot{\mathbf{T}}$ depend on $\boldsymbol{\alpha}$, while the original equilibrium equations do not, namely, they are *intrinsic*.

A second classical family of methods is to introduce a *global non minimal-parametrization* of the rotation, based on a redundant set of parameters, e.g., the quaternions or the 9 components of the rotation matrix [4, 13, 56]. In this case, the redundant set of parameters are related to each other through additional nonlinear algebraic constraints. Classical solvers for differential-algebraic equations on vector spaces can then be used. With this approach, there is no singularity issue and the nonlinearities are usually reduced compared to a global-minimal parametrization strategy. The cost in this case is an increased number of parameters and equations.

In summary, the global parametrization process of the rotations affects meaningfully the equations of motion and destroys many essential properties of the original equations of motion.

The geometric methods, relying on the Lie group structure of the configuration space, have been developed to handle the kinematic equation in Eq. 1.56 without the need of introducing a global parametrization. Such methods are able to preserve the structure of the original equations of motion and do not suffer from any of the drawbacks listed above. The core of such methods is represented by the exponential map on a Lie group, which is described in Appendix B.

1.8.2 Global parametrization-free equations of motion

Let us apply the geometric time integration scheme in Appendix D to the Euler's equations in (1.56)–(1.57). We obtain

$$\mathbf{R}_{n+1} = \mathbf{R}_n \exp_{SO(3)}(\tilde{\mathbf{n}}_{n+1}) \quad (1.62)$$

$$\mathbf{J}\dot{\boldsymbol{\omega}}_{n+1}^T + \tilde{\boldsymbol{\omega}}_{n+1}\mathbf{J}\boldsymbol{\omega}_{n+1} = \mathbf{0}_{3 \times 1} \quad (1.63)$$

where $\mathbf{n}_{n+1} \in \mathfrak{g}$ is a vectorial quantity interpreted as an increment about $\mathbf{R}_n \in G$ and it is projected onto the Lie group thanks to the exponential map, which introduces a *local parametrization* about \mathbf{R}_n . The time integration formulae in (D.7)–(D.9), for a free rotating rigid body become

$$\mathbf{n}_{n+1} = h\boldsymbol{\omega}_n + (0.5 - \beta)h^2\mathbf{a}_n + \beta h^2\mathbf{a}_{n+1} \quad (1.64)$$

$$\boldsymbol{\omega}_{n+1} = \boldsymbol{\omega}_n + (1 - \gamma)h\mathbf{a}_n + \gamma h\mathbf{a}_{n+1} \quad (1.65)$$

$$\mathbf{a}_{n+1} = \frac{1}{1 - \alpha_m} ((1 - \alpha_f)\dot{\boldsymbol{\omega}}_{n+1} + \alpha_f\dot{\boldsymbol{\omega}}_n - \alpha_m\mathbf{a}_n) \quad (1.66)$$

where n refers to the time step, h is the time step size and \mathbf{a} is a vector of pseudo-acceleration. The numerical parameters of the method, namely $\alpha_m, \alpha_f, \gamma, \beta$ are computed according to Eq. D.10. The integration formulae are provided in terms of geometric quantities, namely the rotational velocities $\boldsymbol{\omega}$ and the rotation accelerations $\dot{\boldsymbol{\omega}}$, without expressing them in terms of the local parametrization. Thus, Eq. 1.63 is not affected by the local parametrization process. In particular, it is still quadratic and does not depend on the orientation of the body. All the drawbacks in Sec. 1.8.1 are not presented here. The implicit time integration scheme involves the computation, at each time step, of the iteration matrix in Eq. D.22, which, in case of a rotating body takes the simple form

$$\mathbf{S}_T = \beta'\mathbf{J} + \gamma'(\tilde{\boldsymbol{\omega}}\mathbf{J} - \tilde{\mathbf{J}}\boldsymbol{\omega}) \quad (1.67)$$

Finally, the geometric time integration method applied to the equations of motion of a rigid body (1.51)–(1.52), in case of no external forces, leads to

$$\mathbf{H}_{n+1} = \mathbf{H}_n \exp_{SE(3)}(\tilde{\mathbf{n}}_{n+1}) \quad (1.68)$$

$$\mathbf{M}\dot{\boldsymbol{\eta}}_{n+1} - \hat{\boldsymbol{\eta}}_{n+1}^T \mathbf{M}\boldsymbol{\eta}_{n+1} = \mathbf{0}_{6 \times 1} \quad (1.69)$$

where $\mathbf{n}_{n+1} \in \mathfrak{g}$ is again the vectorial quantity of the increment about $\mathbf{H}_n \in G$ and it is projected onto the Lie group thanks to the exponential map, which introduces a *local parametrization* about \mathbf{H}_n . Its expression and the integration formulae are given by (D.7)–(D.9). Notice that Eq. 1.69 is again not affected by the local parametrization process. The iteration matrix in Eq. D.22, in case of a rigid body, is given by

$$\mathbf{S}_T = \begin{bmatrix} \beta'm\mathbf{I}_{3 \times 3} + \gamma'm\tilde{\boldsymbol{\omega}} & -\gamma'm\tilde{\mathbf{u}} \\ \mathbf{0}_{3 \times 3} & \beta'\mathbf{J} + \gamma'(\tilde{\boldsymbol{\omega}}\mathbf{J} - \tilde{\mathbf{J}}\boldsymbol{\omega}) \end{bmatrix} \quad (1.70)$$

1.9 Summary

- Rigid body transformations are represented by Euclidean transformations which, in turn, can be represented by homogeneous transformation matrices \mathbf{H} , which belongs to the Lie group $SE(3)$.
- The derivatives are naturally expressed by intrinsic quantities, called *twists*, in local frames attached to the bodies. Their geometric description is given by the *screws*. Their mathematical description is given by the Lie algebra $\mathfrak{se}(3)$ associated to the Lie group $SE(3)$.
- The exponential coordinate representation for the motion constitutes a constructive proof of the *Chasles-Mozzi theorem*. The exponential map can be seen as a local parametrization which provides the solution of a linear differential equation on a Lie group.
- The forces and the moments acting on a rigid body can be collected in the *wrench* vector, which plays for the statics the same role that the *twist* plays for the velocity kinematics.
- The representation of *twists* and *wrenches* in different frames is given by the Adjoint representation.
- The equations of motion of a free rigid body are given by

$$\dot{\mathbf{H}} = \mathbf{H}\tilde{\boldsymbol{\eta}}_L \quad (1.71)$$

$$\mathbf{M}\dot{\boldsymbol{\eta}}_L - \hat{\boldsymbol{\eta}}_L^T \mathbf{M}_L \boldsymbol{\eta}_L = \mathbf{0}_{6 \times 1} \quad (1.72)$$

and they are coined as *intrinsic*.

- The geometric time integration method applied to the equations of motion of a free rigid body leads to

$$\mathbf{H}_{n+1} = \mathbf{H}_n \exp_{SE(3)}(\tilde{\mathbf{n}}_{n+1}) \quad (1.73)$$

$$\mathbf{M}\dot{\boldsymbol{\eta}}_{n+1} - \hat{\boldsymbol{\eta}}_{n+1}^T \mathbf{M}\boldsymbol{\eta}_{n+1} = \mathbf{0}_{6 \times 1} \quad (1.74)$$

thus, it allows the development of a *global-parametrization free* framework for the motion description.

Chapter 2

Kinematics

Kinematics pertains to the motion of bodies in a robotic mechanism without regarding to the forces/torques that cause the motion. In case of rigid bodies, the kinematics aims at finding a suitable representation for the position field, namely the three components of position and the three components of orientation, of a body in space. In case of deformable bodies, kinematics also regards the definition of a deformation field, namely a suitable representation for the three components of linear deformations and the three components of angular deformations.

In this Chapter we first derive the kinematics for a continuum arm, then for the piecewise constant deformation arm. In the latter case, we present the *product of exponentials (PoE)* for describing the *forward kinematics* of open chains made of soft bodies. Furthermore, the *inverse kinematics* using classical concepts of differential geometry is presented. The geometric interpretation of the results are discussed recalling the geometry of the reference curve.

2.1 Continuum arm

The initial configuration of a soft continuum arm is defined by considering two-dimensional cross-sections upon a three-dimensional reference curve. The mechanical behaviour of the continuum arm is thus the result of the different rigid body transformations of the cross-sections along the reference curve.

2.1.1 Geometry of the reference curve

The three-dimensional reference curve is parametrized by the material abscissa $\alpha \in \mathbb{R}$. Figure 2.1 illustrates the geometric description of the curve in the initial and current configuration. We indicate the position vector of a point of the curve with $\mathbf{u}(\alpha)$. The unit tangent vector $\mathbf{t}(\alpha)$, the unit normal $\mathbf{n}(\alpha)$ and the unit bi-normal $\mathbf{b}(\alpha)$ to the curve are defined as

$$\mathbf{t}(\alpha) = \frac{\mathbf{u}'(\alpha)}{\|\mathbf{u}'(\alpha)\|} \quad (2.1)$$

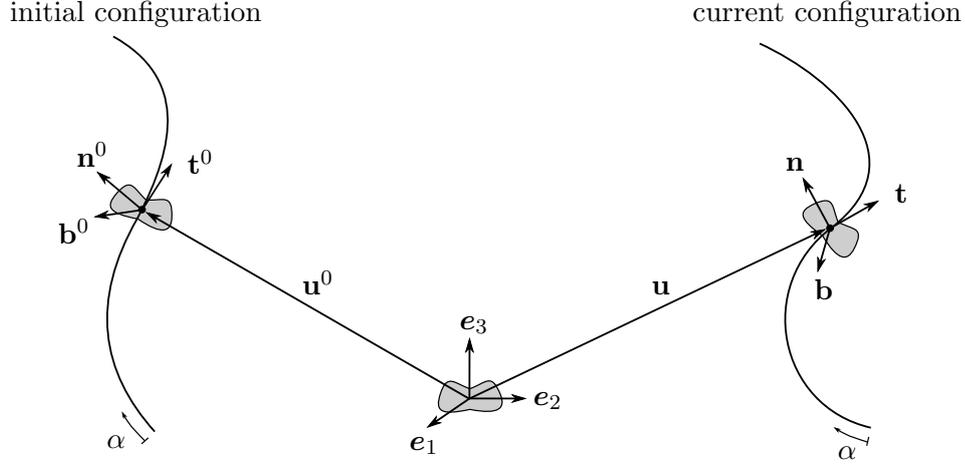


Figure 2.1: Geometric description of the reference curve for the continuum arm.

$$\mathbf{n}(\alpha) = \frac{1}{\|\mathbf{u}'(\alpha)\|\kappa(\alpha)} \mathbf{t}'(\alpha) = \frac{-1}{\|\mathbf{u}'(\alpha)\|\tau(\alpha)} \mathbf{b}'(\alpha) \quad (2.2)$$

$$\mathbf{b}(\alpha) = \tilde{\mathbf{t}}(\alpha) \mathbf{n}(\alpha) \quad (2.3)$$

where $(\cdot)'$ denotes the derivative with respect to α , while $\kappa(\alpha)$ and $\tau(\alpha)$ indicate the curvature and the torsion of the curve defined as

$$\kappa(\alpha) = \frac{\|\mathbf{t}'(\alpha)\|}{\|\mathbf{u}'(\alpha)\|} \quad (2.4)$$

$$\tau(\alpha) = \frac{\|\mathbf{b}'(\alpha)\|}{\|\mathbf{u}'(\alpha)\|} \quad (2.5)$$

The three unit vectors constitute a local triad on the reference curve, and thus, a rotation matrix $\mathbf{R} \in SO(3)$ can be constructed as

$$\mathbf{R}(\alpha) = [\mathbf{t}(\alpha) \ \mathbf{n}(\alpha) \ \mathbf{b}(\alpha)] \quad (2.6)$$

The derivative of (2.6) is expressed by the Frenet formulas, which take the form

$$\mathbf{R}'(\alpha) = \mathbf{R}(\alpha) \tilde{\mathbf{f}}_\omega(\alpha) \quad (2.7)$$

where $\tilde{\mathbf{f}}_\omega(\alpha) \in \mathfrak{so}(3)$, while the associated axial vector $\mathbf{f}_\omega(\alpha)$ reads

$$\mathbf{f}_\omega(\alpha) = \|\mathbf{u}'(\alpha)\| [\tau(\alpha) \ 0 \ \kappa(\alpha)]^T \quad (2.8)$$

This introduces the concept of deformation measures along the reference curve, which will be explained in 2.1.3

2.1.2 Position field

In the $SE(3)$ formalism, the position and orientation fields, which describe the configuration of the continuum arm in space, is represented by the mapping

$$\alpha \in \mathbb{R} \mapsto \mathbf{H}(\alpha) = \mathcal{H}(\mathbf{R}(\alpha), \mathbf{u}(\alpha)) \in SE(3) \quad (2.9)$$

where $\alpha \in [0, L]$, and L is the length of the arm.

2.1.3 Deformation field

The deformation field is obtained by taking the space derivatives of the position field. By using the left invariant representation of the derivatives on $SE(3)$ introduced in Eq. A.2, an element $\tilde{\mathbf{f}}(\alpha) \in \mathfrak{se}(3)$ representing the deformation measures can be introduced as

$$\mathbf{H}'(\alpha) = \mathbf{H}(\alpha)\tilde{\mathbf{f}}(\alpha). \quad (2.10)$$

The deformation measures are identified from the initial configuration as

$$\mathbf{f}(\alpha) = \mathbf{f}^0(\alpha) + \boldsymbol{\epsilon}(\alpha) \quad (2.11)$$

where \mathbf{f}^0 is the initial deformation vector and $\boldsymbol{\epsilon}(\alpha)$ is the 6×1 strain vector which includes the classical position part and orientation part of the deformations as

$$\boldsymbol{\epsilon}(\alpha) = \begin{bmatrix} \boldsymbol{\gamma}(\alpha) \\ \boldsymbol{\kappa}(\alpha) \end{bmatrix} \quad (2.12)$$

Notice that Eq. 2.10 accounts for continuum arms of generic initial configurations (not necessarily straight).

2.2 Piecewise constant deformation soft arm

Equation 2.10 is as a differential equation on a Lie group. Unfortunately, when \mathbf{f} depends on α , i.e., in the general case of three-dimensional curves with variable deformations, there exists no closed form expression for the solution of (2.10).

The intuition for developing a closed form solution of (2.10) is to make a *piecewise constant deformation assumption*, i.e., forcing both the linear and the angular strain to be constant along each segment in which we discretize the original reference curve.

At any instant t , considering the deformation field $\mathbf{f}(\alpha)$ (and thus, the strain field $\boldsymbol{\epsilon}(\alpha)$) constant along each of the N segments of the continuum arm, we can replace the continuous field with a finite set of N six-dimensional twist vectors $\boldsymbol{\epsilon}$, which eventually play the role of the joint vector of traditional rigid robotics.

Thus, by considering the strains as the state of the robotic arm, the ∞ -degrees-of-freedom of the continuum arm are replaced with $N \cdot 6$ variables.

This assumption will lead to the development of a dynamic model for open chains of soft bodies which has the same structure of the serial rigid manipulators, as we will see in Chapter 4.

2.2.1 Forward kinematics

The forward kinematics gives the mapping from the strain field to the $SE(3)$ field of the soft robotic arm.

Let us discretize the reference curve with n curves which present a constant deformation. We call *piecewise constant deformation (PWCD) soft arm* the continuum arm which has been discretized using this approach.

In the most general case, the PWCD is composed by n -elements. Thus, the abscissa $\alpha \in [0, L_n]$ is divided into n parts in the form $[0, L_1), (L_1, L_2), (L_2, L_3), \dots, (L_{n-1}, L_n]$, where L_n is the total length of the arm. By construction, in each element, \mathbf{f} does not depend on α . Hence, the solution of Eq. 2.10 can be obtained by recalling the exponential map as

$$\mathbf{H}(\alpha) = \mathbf{H}_0 \prod_{i=1}^n \exp_{SE(3)} \left((\min(L_i, \alpha) - L_{i-1}) \tilde{\mathbf{f}}_i \right) \quad (2.13)$$

where \mathbf{H}_0 , the frame at $\alpha = 0$, is a constant of integration and \mathbf{f}_i are the deformations in each element. The finite element method discretizes a continuum arm using a finite set of nodes which are connected through a spatial interpolation formula. By using a finite element terminology, we call jA and jB the two end nodes of the j -th element, which are located in the nodal frames $\mathbf{H}_{jA} = \mathcal{H}(\mathbf{R}_{jA}, \mathbf{u}_{jA})$ and $\mathbf{H}_{jB} = \mathcal{H}(\mathbf{R}_{jB}, \mathbf{u}_{jB})$. Since \mathbf{H}_0 is the frame at $\alpha = 0$, we have $\mathbf{H}_0 = \mathbf{H}_{1A} \mathbf{H}_{1A0}$, where \mathbf{H}_{1A0} is a constant frame which accounts for the initial configuration, while \mathbf{H}_{1A} is the frame from the initial configuration to the current configuration. In the following, if not differently indicated, we simply use \mathbf{H}_{1A} for the product $\mathbf{H}_{1A} \mathbf{H}_{1A0}$. The length of the j -th element is $l_j = L_j - L_{j-1}$ and the corresponding abscissa $\alpha_j \in (L_{j-1}, L_j)$. For each element, we introduce the relative configuration vector $\mathbf{d}_i = [\mathbf{d}_{u,i}^T, \mathbf{d}_{\omega,i}^T]^T$ defined from the deformation vector as

$$\mathbf{f}_i = \frac{\mathbf{d}_i}{L_n} \quad (2.14)$$

Hence, by using (2.14), Eq. 2.13 becomes

$$\mathbf{H}(\alpha) = \mathbf{H}_{1A} \prod_{i=1}^n \exp_{SE(3)} \left(\frac{\min(L_i, \alpha) - L_{i-1}}{L_n} \tilde{\mathbf{d}}_i \right) \quad (2.15)$$

which represents the *product of exponentials* (PoE) formula for the forward kinematics mapping of robotic manipulators structured in open chains made of soft bodies. This is the generalization of the PoE representation for rigid manipulators: in the case of soft manipulators, the joint values as well as the joint twists are replaced with the twists representing the strains along the kinematic structure. The process is illustrated in Fig. 2.2.

The PoE formula for the j -th element of the PWCD soft arm reads

$$\mathbf{H}_j(\alpha_j) = \mathbf{H}_{jA} \exp_{SE(3)} \left(\frac{\alpha_j - L_{j-1}}{L_j} \tilde{\mathbf{d}}_j \right) \quad (2.16)$$

where the initial condition is given by

$$\mathbf{H}_{jA} = \mathbf{H}_{1A} \prod_{i=1}^{j-1} \exp_{SE(3)} \left(\frac{L_i - L_{i-1}}{L_n} \tilde{\mathbf{d}}_i \right) \quad (2.17)$$

Equation 2.16 represents a formula for the spatial interpolation of consecutive frames, elements belonging to $SE(3)$. This formula is the shape function involved in the finite element discretization. This equation can be interpreted as, starting from the nodal frame $\mathbf{H}_{jA} = \mathbf{H}_j(L_{j-1})$, the end frame $\mathbf{H}_{jB} = \mathbf{H}_j(L_j)$ of the j -th element is approached by moving along the frame transformation implied by the projection on the group of the relative configuration vector \mathbf{d}_j . The exponential map introduces a local parametrization which allows the description of the reference curve between the two nodes jA and jB with an element belonging to a linear space, namely the Lie algebra $\mathfrak{se}(3)$.

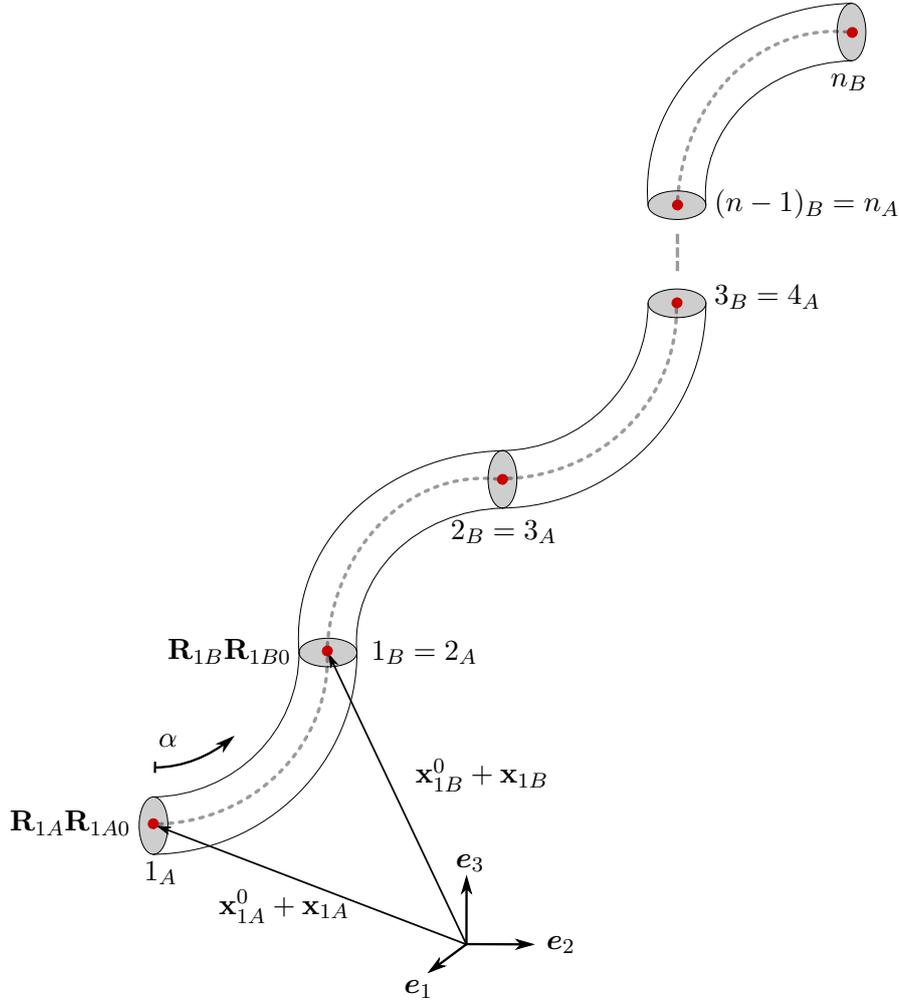


Figure 2.2: Kinematics of the piecewise constant deformation model

2.2.2 Inverse kinematics

The inverse kinematics gives the mapping from the position and orientation fields to the strain field of the soft robotic arm.

Let us consider a PWCD soft manipulator with n elements. By using (2.15), the configuration of the last node is given by

$$\mathbf{H}_{nB} = \mathbf{H}_{1A} \prod_{i=1}^n \exp_{SE(3)} \left(\frac{L_i - L_{i-1}}{L_n} \tilde{\mathbf{d}}_i \right) \quad (2.18)$$

In differential geometry, the inverse of the exponential map is represented by the logarithmic map. If we consider a first order approximation, $\prod \exp(\tilde{\mathbf{d}}_i) = (\sum \tilde{\mathbf{d}}_i)$, i.e., we truncate the Lie bracket expressions of the Campbell-Backer-Hausdorff relations [43], the inversion of 2.18 takes the recursive form

$$\sum_{j=1}^n \left(\frac{L_j - L_{j-1}}{L_n} \mathbf{d}_j \right) = \log_{SE(3)}(\mathbf{H}_{1A}^{-1} \mathbf{H}_{nB}) \quad (2.19)$$

Equation 2.19 relates the configurations of the first and the last node of the manipulator with the sum of the relative configurations vectors for each element. In order to solve the inverse kinematics problem for the $6n$ unknown quantities, namely the strains for all the elements, we need to solve the inverse kinematics from the tip to the base. By applying (2.19) for all the elements, we obtain a set of $6n$ equations in $6n$ unknowns

$$\begin{aligned} \frac{L_n - L_{n-1}}{L_n} \mathbf{d}_n + \dots + \frac{L_1 - L_0}{L_n} \mathbf{d}_1 &= \log_{SE(3)}(\mathbf{H}_{1A}^{-1} \mathbf{H}_{nB}) & (2.20) \\ \frac{L_{n-1} - L_{n-2}}{L_n} \mathbf{d}_{n-1} + \dots + \frac{L_1 - L_0}{L_n} \mathbf{d}_1 &= \log_{SE(3)}(\mathbf{H}_{1A}^{-1} \mathbf{H}_{(n-1)B}) & (2.21) \\ & \cdot \\ & \cdot \\ & \cdot \end{aligned}$$

$$\frac{L_2 - L_1}{L_n} \mathbf{d}_2 + \frac{L_1 - L_0}{L_n} \mathbf{d}_1 = \log_{SE(3)}(\mathbf{H}_{1A}^{-1} \mathbf{H}_{2B}) \quad (2.22)$$

$$\frac{L_1 - L_0}{L_n} \mathbf{d}_1 = \log_{SE(3)}(\mathbf{H}_{1A}^{-1} \mathbf{H}_{1B}) \quad (2.23)$$

Starting from the base, from Eq. 2.23 we compute \mathbf{d}_1 . Thus, using (2.22) we obtain \mathbf{d}_2 . By applying this process recursively till the last element of the soft manipulator, we obtain the six-dimensional vectors of relative configurations for all the elements. The only required data are: the length of each section, the configurations of the $n + 1$ nodes of the manipulator.

Starting from (2.16), let us now consider the configuration of the final node of the j -th element as

$$\mathbf{H}_{jB} = \mathbf{H}_{jA} \exp_{SE(3)} \left(\frac{L_i - L_{i-1}}{L_n} \tilde{\mathbf{d}}_i \right) \quad (2.24)$$

The inversion of (2.24) reads

$$\frac{L_j - L_{j-1}}{L_n} \mathbf{d}_j = \log_{SE(3)}(\mathbf{H}_{jA}^{-1} \mathbf{H}_{jB}) \quad (2.25)$$

Equation 2.25 gives the six deformations of the j -th element starting from the initial and final frames of the j -th element.

Using (2.11) and (2.14), the strain field for the generic j -th section reads

$$\epsilon_j = \frac{\mathbf{d}_j - \mathbf{d}_j^0}{l_j} \quad (2.26)$$

where \mathbf{d}_j^0 is the relative configuration vector in the initial configuration. It is given by

$$\mathbf{d}_j^0 = \begin{bmatrix} \mathbf{d}_{u,j}^0 \\ \mathbf{d}_{\omega,j}^0 \end{bmatrix} = \begin{bmatrix} \mathbf{T}_{SO(3)}^{-T}(\mathbf{d}_{\omega,j}^0) \mathbf{R}_{jA0}^T (\mathbf{u}_{jB}^0 - \mathbf{u}_{jA}^0) \\ \log_{SO(3)}(\mathbf{R}_{jA0}^T \mathbf{R}_{jB0}) \end{bmatrix} \quad (2.27)$$

where $\mathbf{T}_{SO(3)}^{-T}$ is the transpose of the tangent operator given in (B.17), the vectors \mathbf{u}_{jA}^0 and \mathbf{u}_{jB}^0 define the initial position of the two nodes of the j -th element, while the matrices \mathbf{R}_{jA0} and \mathbf{R}_{jB0} define the initial orientation of the cross sections. By recalling (2.6), these matrices constitute the Frenet triads at the element extremities, i.e., their first column is the tangent to the reference curve and the second and third columns are the cross-section axes. Notice that $\|\mathbf{d}_{u,j}^0\| = l_j$, i.e., the relative configuration vector has been constructed such that the norm of its positional part in the initial configuration is equal to the length of the j -th element. Indeed, the vector $\mathbf{d}_{\omega,j}^0$ accounts for the initial curvature of the element. In case of an initially straight element, we have $\mathbf{R}_{jA0} = \mathbf{R}_{jB0}$, so that $\mathbf{d}_{\omega,j}^0 = \mathbf{0}_{3 \times 1}$ and $\mathbf{d}_{u,j}^0 = \mathbf{R}_{jA0}^T (\mathbf{u}_{jB}^0 - \mathbf{u}_{jA}^0)$.

The combined use of (2.19) and (2.26) solves the inverse kinematics problem. This problem has a big practical impact. The application scenario could be the design of a custom soft manipulator which has to adapt its shape to a nonlinear path given by the environment. This circumstance could be, for instance, the one that occurs in robotic surgery for intravascular operations or in maintenance and inspection procedures in hazardous spaces which are difficult to reach with standard robotic systems. In such cases, we can plan offline the configurations of two consecutive nodes such that the corresponding deformations of the element result constant. Thus, based on the constrained environment in which the manipulator has to operate, we can select the number of nodes of the manipulator, evaluate their configuration in three-dimensional space, computing the deformations of the elements using (2.19) and (2.26), and selecting an appropriate actuation system.

2.2.3 Special case: constant deformation soft arm

In this section we present the special case of the constant deformation soft arm, i.e., the PWCD model with $n = 1$. In this case, the dimensionality of the problem reduces from ∞ to 6. The circumstance in which a solid can be approximate with a body of constant deformations occurs in a lot of practical robotics problems.

Forward kinematics

The geometric description of a constant deformation soft arm is illustrated in Fig. 2.3. In this case, the abscissa $\alpha \in [0, L]$, being L the total length of the arm. The forward kinematics mapping in (2.15) reads

$$\mathbf{H}(\alpha) = \mathbf{H}_A \exp_{SE(3)} \left(\frac{\alpha}{L} \tilde{\mathbf{d}} \right) \quad (2.28)$$

where in \mathbf{H}_A is also present $\mathbf{H}_{A0} = \mathcal{H}(\mathbf{R}_{A0}, \mathbf{0})$, the constant frame which accounts for the initial configuration of the cross section. Equation 2.28 is the PoE formula for a constant deformation soft arm.

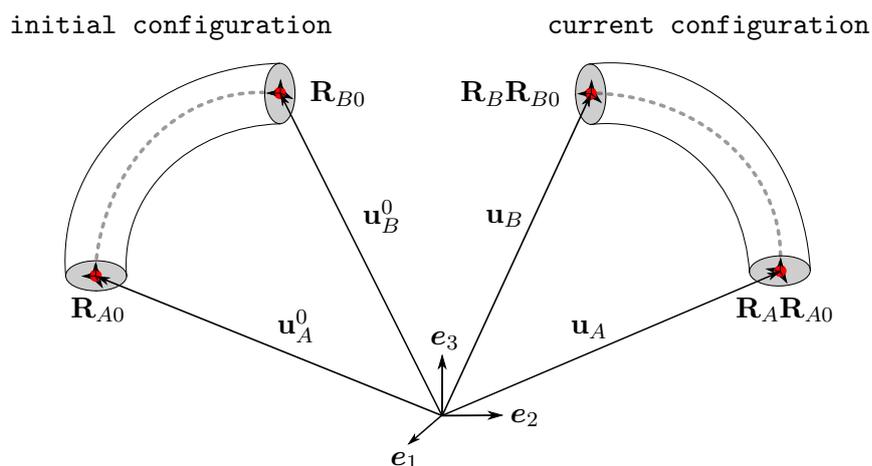


Figure 2.3: Constant deformation soft arm in its initial and current configuration.

Inverse kinematics

The inversion of (2.28) is exactly given by

$$\mathbf{d} = \log_{SE(3)}(\mathbf{H}_A^{-1} \mathbf{H}_B). \quad (2.29)$$

where, in the same fashion as \mathbf{H}_A also \mathbf{H}_B contains $\mathbf{H}_{B0} = \mathcal{H}(\mathbf{R}_{B0}, \mathbf{0})$, the constant frame which account for the initial configuration of the cross section of the end node.

Using (2.11) and (2.14), the strain field reads

$$\boldsymbol{\epsilon} = \frac{\mathbf{d} - \mathbf{d}^0}{L} \quad (2.30)$$

where \mathbf{d}^0 is the relative configuration vector in the initial configuration, and is given by

$$\mathbf{d}^0 = \begin{bmatrix} \mathbf{d}_u^0 \\ \mathbf{d}_\omega^0 \end{bmatrix} = \begin{bmatrix} \mathbf{T}_{SO(3)}^{-T}(\mathbf{d}_\omega^0) \mathbf{R}_{A0}^T (\mathbf{u}_B^0 - \mathbf{u}_A^0) \\ \log_{SO(3)}(\mathbf{R}_{A0}^T \mathbf{R}_{B0}) \end{bmatrix} \quad (2.31)$$

The combined use of (2.29) and (2.30) solves the inverse kinematics problem for the constant deformation soft arm.

Geometrical interpretation

Let us construct the local Frenet triad along the interpolated reference curve. From the space derivative of (2.28), we obtain

$$\mathbf{u}'(\alpha) = \mathbf{R}(\alpha) \frac{\tilde{\mathbf{d}}_u}{L} \quad \text{and} \quad \mathbf{R}'(\alpha) = \mathbf{R}(\alpha) \frac{\tilde{\mathbf{d}}_\omega}{L} \quad (2.32)$$

The unit tangent to the reference curve is given by

$$\mathbf{t}(\alpha) = \frac{\mathbf{u}'(\alpha)}{\|\mathbf{u}'(\alpha)\|} = \mathbf{R}(\alpha) \frac{\mathbf{d}_u}{L} \quad (2.33)$$

where we used $\|\mathbf{d}_u\| = L$. For computing the unit normal vector, let us derive (2.33) with respect to α as

$$\mathbf{t}'(\alpha) = \mathbf{R}(\alpha) \frac{\tilde{\mathbf{d}}_\omega}{L} \frac{\mathbf{d}_u}{L} \quad (2.34)$$

so that, using (2.2), the unit normal vector is given by

$$\mathbf{n}(\alpha) = \frac{1}{\kappa} \mathbf{t}'(\alpha) = \mathbf{R}(\alpha) \frac{\tilde{\mathbf{d}}_\omega \mathbf{d}_u}{\|\tilde{\mathbf{d}}_\omega \mathbf{d}_u\|} \quad (2.35)$$

where the curvature is computed according to (2.4) as

$$\kappa = \frac{\|\tilde{\mathbf{d}}_\omega \mathbf{d}_u\|}{L^2} \quad (2.36)$$

It results that, since \mathbf{d} is assumed to be constant, the interpolated reference curve is a curve with constant curvature. Finally, using Eq. 2.3, the unit bi-normal is calculated as

$$\mathbf{b}(\alpha) = \frac{1}{\|\mathbf{d}_u\| \|\tilde{\mathbf{d}}_\omega \mathbf{d}_u\|} \widetilde{\mathbf{R}(\alpha) \mathbf{d}_u \mathbf{R}(\alpha) \mathbf{d}_\omega \mathbf{d}_u} = \mathbf{R}(\alpha) \frac{\tilde{\mathbf{d}}_u \tilde{\mathbf{d}}_\omega \mathbf{d}_u}{\|\mathbf{d}_u\| \|\tilde{\mathbf{d}}_\omega \mathbf{d}_u\|} \quad (2.37)$$

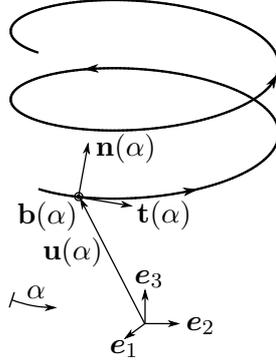


Figure 2.4: Geometry of the interpolated reference curve.

In order to compute the torsion τ , we need to compute $\mathbf{b}'(\alpha)$, since Eq. 2.2 holds. It results

$$\mathbf{b}'(\alpha) = -\frac{(\mathbf{d}_\omega^T \mathbf{d}_u)}{L \|\mathbf{d}_u\|} \mathbf{R}(\alpha) \frac{\tilde{\mathbf{d}}_\omega \mathbf{d}_u}{\|\tilde{\mathbf{d}}_\omega \mathbf{d}_u\|} \quad (2.38)$$

Thus, τ is given by

$$\tau = \frac{(\mathbf{d}_\omega^T \mathbf{d}_u)}{L^2} \quad (2.39)$$

It results that the interpolated reference curve presents a constant torsion along its axis. Developing (2.36) and (2.39), we obtain

$$\kappa = \frac{\|\mathbf{d}_\omega\| \sin(\mathbf{d}_\omega, \mathbf{d}_u)}{L} \quad \text{and} \quad \tau = \frac{\|\mathbf{d}_\omega\| \cos(\mathbf{d}_\omega, \mathbf{d}_u)}{L} \quad (2.40)$$

and the Gaussian curvature κ_g computed as

$$\kappa = \sqrt{\kappa^2 + \tau^2} = \frac{\|\mathbf{d}_\omega\|}{L} = \frac{\|\mathbf{d}_\omega\|}{\|\mathbf{d}_u\|} \quad (2.41)$$

is also constant along the constant deformation soft arm. Thus, the interpolated reference curve, from a geometric point of view, is represented by an *helix*, i.e., a spatial curve with constant curvature and torsion. Its geometric interpretation is given in Fig. 2.4. Notice that no approximation on the magnitude of the initial curvature or torsion of the arm has been made in the development of the formulation. This means that the presented formalism is able to account for finite curvature and torsion of the initial configuration of the soft arm. Since the exact solution for the position and orientation field of the arm in a pure bending/torsion configuration is a curve of constant curvature and torsion, and since the interpolation formula in (2.28) contains this exact solution, the presented discrete model is exact in pure bending/torsion configurations. Thus, the somewhat abstract mathematical concept of exponential map on $SE(3)$ translates in the intuitive geometric concept of an helix, a spatial differentiable curve.

2.3 Summary

- The *forward kinematics* of a piecewise constant deformation soft arm is described by the generalization of the product of exponentials formula as

$$\mathbf{H}(\alpha) = \mathbf{H}_{1A} \prod_{i=1}^n \exp_{SE(3)} \left(\frac{\min(L_i, \alpha_i) - L_{i-1}}{L_n} \tilde{\mathbf{d}}_i \right) \quad (2.42)$$

- The *inverse kinematics* of a piecewise constant deformation soft arm is described by using the logarithmic map as

$$\sum_{j=1}^n \left(\frac{L_j - L_{j-1}}{L_n} \mathbf{d}_j \right) = \log_{SE(3)}(\mathbf{H}_{1A}^{-1} \mathbf{H}_{nB}) \quad (2.43)$$

and the strain field for the generic j -th element is given by

$$\boldsymbol{\epsilon}_j = \frac{\mathbf{d}_j - \mathbf{d}_j^0}{l_j} \quad (2.44)$$

Chapter 3

Differential kinematics and statics

Differential kinematics aims at deriving a velocity field and an acceleration field for robotic mechanisms. The derivation of the strain energy, together with the application of the principle of virtual work, allows to determine the static equilibrium equations, i.e., the configuration of the soft manipulator in static equilibrium settings.

The chapter is built on top of the derivation of the mapping between the velocities along the arm and the derivatives of the states of the manipulator, namely the internal strains. Using a terminology beloved to the robotics community, we refer to this mapping as the *soft geometric Jacobian*. The Jacobian constitutes an essential tool to describe the differential kinematics, the statics and, as we will see in the next Chapter, the dynamics of the piecewise constant deformation model.

3.1 Continuum arm

Some mathematical tools from the differential geometry, namely the left invariant vector field of the derivatives in (A.2) and the Lie bracket operator in (A.7) are used here to define in a mechanics context the velocity field and the acceleration field, the latter through the derivation of the compatibility equations. The selection of a constitutive law for the material allows the definition of the internal energy of the continuum arm, used for the derivation of the statics.

3.1.1 Velocity field

The velocity field is obtained by taking the time derivatives of the position field. By using the left invariant representation of the derivatives on $SE(3)$, we can introduce the velocity variables as an element $\tilde{\boldsymbol{\eta}}(\alpha) \in \mathfrak{se}(3)$, which is associated to the 6×1 axial vector $\boldsymbol{\eta}(\alpha) = [\mathbf{v}(\alpha)^T \ \boldsymbol{\omega}(\alpha)^T]^T$, where \mathbf{v} and $\boldsymbol{\omega}$ are respectively the linear and angular velocities. Hence, the derivative of (2.9) with respect to

time yields

$$\dot{\mathbf{H}}(\alpha) = \mathbf{H}(\alpha)\tilde{\boldsymbol{\eta}}(\alpha) \quad (3.1)$$

which constitutes the velocity field of a continuum arm.

3.1.2 Compatibility equations

Compatibility conditions for finite strains in continuum mechanics are formulated such that the body is left without unphysical gaps or overlaps after a deformation. This translates in formulating compatibility conditions between the strain and the velocity of a continuum body.

Considering that two different derivatives are involved to define the strain and the velocity, namely respectively the space and time derivatives, the commutativity of the cross derivatives must hold, according to (A.9). Hence, this condition is used to formulate the compatibility equations as

$$\boldsymbol{\eta}' - \dot{\boldsymbol{\epsilon}} = \hat{\boldsymbol{\eta}}\boldsymbol{\epsilon} \quad (3.2)$$

Notice that similar compatibility equations can be formulated as $\tilde{\boldsymbol{\eta}}' - \dot{\tilde{\boldsymbol{\epsilon}}} = [\tilde{\boldsymbol{\eta}}, \tilde{\boldsymbol{\epsilon}}]$, where $[\cdot, \cdot]$ indicates the Lie bracket operator defined in (A.7).

3.1.3 Acceleration field

By taking the time derivative of (3.2), we obtain the continuous model of acceleration as

$$\dot{\boldsymbol{\eta}}' - \ddot{\boldsymbol{\epsilon}} = \hat{\dot{\boldsymbol{\eta}}}\boldsymbol{\epsilon} + \hat{\boldsymbol{\eta}}\dot{\boldsymbol{\epsilon}} \quad (3.3)$$

3.1.4 Strain energy

The internal strain energy of the continuum arm is defined as

$$\mathcal{V}_{int} = \frac{1}{2} \int_L \boldsymbol{\epsilon}^T \mathbf{r} \, d\alpha \quad (3.4)$$

where $\mathbf{r} = [\mathbf{n}(\alpha)^T \ \mathbf{m}(\alpha)^T]^T$ is the vector of the stress resultants over the cross-section of the arm, and \mathbf{n} and \mathbf{m} are the 3×1 resulting force and resulting moment vectors. In particular, n_1 is the force along the reference curve, while n_2 and n_3 are the shear forces along the cross-section axes. Indeed, m_1 is the torsion moment about the reference curve, while m_2 and m_3 are the bending moments about the cross-section axes.

The internal force \mathbf{r} and the mechanical strains $\boldsymbol{\epsilon}$ can be related through the material constitutive law. Linear constitutive equations for an isotropic hyperelastic material are chosen both for the elastic and the viscous members, and no bulge effects are considered. This simplified approach suits for robotics applications, where the objective is to describe the global dynamics and the geometric properties of the system, rather than details on material behaviour.

The simplest visco-elastic constitutive model is the Kelvin-Voigt model [57], which simply adds a viscous linear contribution, proportional to the rate of the strains, to the elastic contribution, as

$$\mathbf{r} = \mathbf{K}\boldsymbol{\epsilon} + \boldsymbol{\Upsilon}\dot{\boldsymbol{\epsilon}} \quad (3.5)$$

where \mathbf{K} and $\boldsymbol{\Upsilon}$ are respectively the 6×6 stiffness matrix and the 6×6 viscosity matrix equal to

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{uu} & \mathbf{K}_{u\omega} \\ \text{SYM} & \mathbf{K}_{\omega\omega} \end{bmatrix} \quad \text{and} \quad \boldsymbol{\Upsilon} = \begin{bmatrix} \boldsymbol{\Upsilon}_{uu} & \boldsymbol{\Upsilon}_{u\omega} \\ \text{SYM} & \boldsymbol{\Upsilon}_{\omega\omega} \end{bmatrix} \quad (3.6)$$

In general, these matrices are not diagonal. But, in case of an initially straight configuration of the soft arm, they become diagonal when the reference curve is chosen to be the neutral axis of the arm, and \mathbf{n} and \mathbf{b} are chosen to be the principal axes of the cross-sections. In such case, $\mathbf{K}_{uu} = \text{diag}(EA, GA_2, GA_3)$ contains the axial and shear stiffnesses and $\mathbf{K}_{\omega\omega} = \text{diag}(GJ, EI_2, EI_3)$ contains the torsional and bending stiffnesses, being $E \in \mathbb{R}$ and $G \in \mathbb{R}$ the Young modulus and the shear modulus. For an isotropic material, it holds $G = E/2(1+\nu)$, where $\nu \in \mathbb{R}$ is the Poisson ration. Similarly, $\boldsymbol{\Upsilon}_{uu} = \text{diag}(3A, A, A)\mu$ is the linear part of the viscosity tensor and $\boldsymbol{\Upsilon}_{\omega\omega} = \text{diag}(J_1, J_2, J_3)$ is the rotational part, for an incompressible material, where $\mu \in \mathbb{R}$ is the shear viscosity.

Using (3.5), Eq. 3.4 becomes

$$\mathcal{V}_{int} = \frac{1}{2} \int_L \boldsymbol{\epsilon}^T \mathbf{K} \boldsymbol{\epsilon} \, d\alpha + \frac{1}{2} \int_L \boldsymbol{\epsilon}^T \boldsymbol{\Upsilon} \dot{\boldsymbol{\epsilon}} \, d\alpha \quad (3.7)$$

where the first term at the right hand side recalls the well known structure of the internal energy for a linear elastic material expressed as a quadratic form in $\boldsymbol{\epsilon}$.

3.1.5 Statics

According to the principle of virtual work, the manipulator is in static equilibrium if and only if

$$\delta(\mathcal{V}_{int}) = \delta(\mathcal{V}_{ext}) \quad (3.8)$$

where $\delta(\mathcal{V}_{ext})$ is the virtual work done by the external forces. The variation of the expression of the internal energy (3.4) reads

$$\delta(\mathcal{V}_{int}) = \int_L \delta(\boldsymbol{\epsilon})^T \mathbf{r} \, d\alpha \quad (3.9)$$

where, recalling the commutativity of the Lie derivatives in (A.9), the variation of the strains is expressed as

$$\delta(\boldsymbol{\epsilon}) = \delta(\mathbf{f}) = (\delta\mathbf{h})' + \widehat{\mathbf{f}}\delta\mathbf{h} \quad (3.10)$$

in which we used $\delta(\mathbf{H}(\alpha)) = \mathbf{H}(\alpha)\widetilde{\delta\mathbf{h}}(\alpha)$. Inserting (3.10) into Eq. 3.9 and integrating by parts yield

$$\delta(\mathcal{V}_{int}) = [\delta\mathbf{h}^T \mathbf{r}] \Big|_0^L - \int_L \delta\mathbf{h}^T (\mathbf{r}' - \hat{\mathbf{f}}^T \mathbf{r}) \, d\alpha \quad (3.11)$$

where the first term at the right hand side is interpreted as a boundary condition. In general, the virtual work done by the external forces can be expressed as

$$\delta(\mathcal{V}_{ext}) = +\delta\mathbf{h}(0)^T \mathbf{g}_{ext}(0) - \delta\mathbf{h}(L)^T \mathbf{g}_{ext}(L) - \int_L \delta\mathbf{h}^T \mathbf{g}_{ext} \, d\alpha \quad (3.12)$$

where $\mathbf{g}_{ext}(\alpha) = [\mathbf{g}_{ext,u}^T \ \mathbf{g}_{ext,\omega}^T]^T$ contains the resulting forces and moments over the cross-sections due to the external loading, expressed in the local frames attached to the cross-sections.

Finally, the weak form of the static equilibrium equations is obtained by inserting (3.12) and (3.11) into Eq. 3.8, which yields

$$[\delta\mathbf{h}^T (\mathbf{r} - \mathbf{g}_{ext})] \Big|_0^L - \int_L \delta\mathbf{h}^T (\mathbf{r}' - \hat{\mathbf{f}}^T \mathbf{r} - \mathbf{g}_{ext}) \, d\alpha = 0 \quad (3.13)$$

Indeed, the strong form reads

$$\mathbf{r}' - \hat{\mathbf{f}}^T \mathbf{r} = \mathbf{g}_{ext} \quad (3.14)$$

3.1.6 Example

Let us consider the soft arm in Fig. 3.1, clamped in its origin, and submitted to a tip load at its free end. This means that $\delta\mathbf{h}(0) = \mathbf{0}_{6 \times 1}$, $\alpha \in [0, L[$ and

$$\mathbf{r}(L) = \mathbf{K}(L)\boldsymbol{\epsilon}(L) = \mathbf{g}_{ext} \quad (3.15)$$

where we consider the soft arm made of linear elastic material. Let also consider that \mathbf{f}^0 and \mathbf{K} are constant over the length of the arm. This hypothesis fits for this example since it corresponds to a soft arm of constant initial torsion and curvature, and constant cross-section geometry.

Deformation field

The equilibrium equations in the static configuration expressed by Eq. 3.14 become

$$\mathbf{K}\boldsymbol{\epsilon}' - \hat{\mathbf{f}}^{0T} \mathbf{K}\boldsymbol{\epsilon} = \mathbf{0}_{6 \times 1} \quad (3.16)$$

where we used the fact that the stiffness matrix is constant over the arm. In this special case, the analytical solution of the deformation field can be expressed in closed form as

$$\boldsymbol{\epsilon}(\alpha) = \mathbf{K}^{-1} \mathbf{F}(\alpha) \mathbf{K} \boldsymbol{\epsilon}_0 \quad (3.17)$$

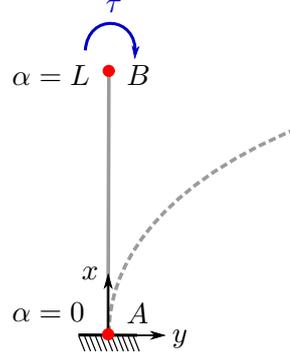


Figure 3.1: Continuum arm subject to a torque τ at its free end. Solid line: initial configuration. Dashed line: deformed configuration.

where ϵ_0 , the deformation at $\alpha = 0$, is a constant of integration and

$$\mathbf{F}(\alpha) = \begin{bmatrix} \exp_{SO(3)}(\alpha \mathbf{f}_\omega^0)^T & \mathbf{0}_{3 \times 3} \\ (\mathbf{T}_{SO(3)}(\alpha \mathbf{f}_\omega^0) \alpha \mathbf{f}_u^0)^T \exp_{SO(3)}(\alpha \mathbf{f}_\omega^0)^T & \exp_{SO(3)}(\alpha \mathbf{f}_\omega^0)^T \end{bmatrix} \quad (3.18)$$

Inserting Eq. 3.17, computed at $\alpha = L$, in the boundary condition given by (3.15), we have

$$\mathbf{K} \mathbf{K}^{-1} \mathbf{F}(L) \mathbf{K} \epsilon_0 = \mathbf{g}_{ext}(L) \quad (3.19)$$

Thus, the constant of integration ϵ_0 is given by

$$\epsilon_0 = \mathbf{K}^{-1} \mathbf{F}(-L) \mathbf{g}_{ext}(L) \quad (3.20)$$

where $(\mathbf{F}(L))^{-1} = \mathbf{F}(-L)$ is a property of the matrix exponential. Hence, the solution of the deformation field is given by

$$\epsilon(\alpha) = \mathbf{K}^{-1} \mathbf{F}(\alpha) \mathbf{F}(-L) \mathbf{g}_{ext}(L) \quad (3.21)$$

In the case of pure bending/torsion solicitation, we have $\mathbf{g}_{ext,u}(L) = \mathbf{0}_{3 \times 1}$ and $\mathbf{g}_{ext,\omega}(L) = \tau \mathbf{m}$, where $M \in \mathbb{R}$ and $\mathbf{m} = \mathbf{f}_\omega^0$ if $\|\mathbf{f}_\omega^0\| \neq 0$ and is an arbitrary vector $\in \mathbb{R}^3$ if $\|\mathbf{f}_\omega^0\| = 0$. In this case, we have

$$\mathbf{F}(\alpha) \mathbf{F}(-L) \begin{bmatrix} \mathbf{0}_{3 \times 1} \\ \tau \mathbf{m} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{3 \times 1} \\ \tau \mathbf{m} \end{bmatrix} \quad (3.22)$$

and the solution for the deformation field in (3.21) becomes

$$\epsilon(\alpha) = \mathbf{K}^{-1} \begin{bmatrix} \mathbf{0}_{3 \times 1} \\ \tau \mathbf{m} \end{bmatrix} \quad (3.23)$$

where ϵ is constant along the continuum arm. For an initially straight arm, the solution reads

$$\begin{bmatrix} \gamma \\ \boldsymbol{\kappa} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{3 \times 1} \\ \mathbf{K}_{\omega\omega}^{-1}(\tau \mathbf{m}) \end{bmatrix} \quad (3.24)$$

where $\mathbf{K}_{\omega\omega} = \text{diag}(GJ, EI_2, EI_3)$.

$SE(3)$ field

In order to obtain the position and rotation fields, the kinematic equations in (2.10) must be solved, i.e.,

$$\mathbf{H}'(\alpha) = \mathbf{H}(\alpha)(\tilde{\mathbf{f}}^0(\alpha) + \tilde{\boldsymbol{\epsilon}}(\alpha)) \quad (3.25)$$

Since there exists no closed form expression of the general solution of this equation, we need to compute it in a particular case. In the case of pure bending/torsion solicitation, $\boldsymbol{\epsilon}$ and \mathbf{f}^0 do not depend on α , and the solution of the $SE(3)$ field is given by

$$\mathbf{H}(\alpha) = \mathbf{H}_0 \exp_{SE(3)}(\alpha(\tilde{\mathbf{f}}^0 + \tilde{\boldsymbol{\epsilon}})) \quad (3.26)$$

where $\mathbf{H}_0 = \mathcal{H}(\mathbf{R}_0, \mathbf{u}_0)$, the frame at $\alpha = 0$, is a constant of integration. Explicitly, Eq. 3.26 means

$$\mathbf{u}(\alpha) = \mathbf{u}_0 + \mathbf{R}_0 \mathbf{T}_{SO(3)}^T(\alpha(\mathbf{f}_\omega^0 + \boldsymbol{\kappa})\alpha(\mathbf{f}_u^0 + \boldsymbol{\gamma})) \quad (3.27)$$

$$\mathbf{H}(\alpha) = \mathbf{R}_0 \exp_{SO(3)}(\alpha(\mathbf{f}_\omega^0 + \boldsymbol{\kappa})) \quad (3.28)$$

In our example, the soft arm is initially straight ($\mathbf{u}_0 = \mathbf{0}_{3 \times 1}$, $\mathbf{R}_0 = \mathbf{I}_{3 \times 3}$) and subject to a tip bending moment $\mathbf{g}_{ext}(L) = \tau[0 \ 0 \ 1]^T$. It results that $\mathbf{f}_u^0 = [1 \ 0 \ 0]^T$, $\mathbf{f}_\omega^0 = \mathbf{0}_{3 \times 1}$, $\boldsymbol{\kappa} = [0 \ 0 \ \kappa]^T$, $\boldsymbol{\gamma} = \mathbf{0}_{3 \times 1}$, where $\kappa = \tau/(EI_2)$, such that the position and rotations fields are given by

$$\mathbf{u}(\alpha) = \begin{bmatrix} \frac{1}{\kappa} \sin(\alpha\kappa) \\ \frac{1}{\kappa} (1 - \cos(\alpha\kappa)) \\ 0 \end{bmatrix} \quad (3.29)$$

$$\mathbf{R}(\alpha) = \begin{bmatrix} \cos(\alpha\kappa) & -\sin(\alpha\kappa) & 0 \\ \sin(\alpha\kappa) & \cos(\alpha\kappa) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.30)$$

which is the exact solution with a circular shape expected from the classical Cosserat rod theory.

3.2 Piecewise constant deformation soft arm

Differential kinematics and statics for the piecewise constant deformation model reduce to the derivation of the Soft Geometric Jacobian, a $6n \times 6n$ matrix which maps either the velocity along the arm with the strain's rate or the external forces with the internal forces.

3.2.1 Soft geometric Jacobian

In this section we will derive formally the soft geometric Jacobian, as the relationship between the velocity along the arm $\boldsymbol{\eta}(\alpha)$ and the time derivative of the states $\dot{\boldsymbol{\epsilon}}$.

Velocity

$\mathbf{n} = 1$

For simplicity of computation, we first compute the velocity kinematics for a constant deformation soft arm, and after we generalize the treatment for a n -elements piecewise constant deformation manipulator.

We will find this relationship by first computing the velocity as a function of the nodal velocities and after the variation of the strain as a function of the nodal velocities. Let us introduce a 12×1 velocity vector $\boldsymbol{\eta}_{AB} = [\boldsymbol{\eta}_A^T \ \boldsymbol{\eta}_B^T]$ from the nodal velocities $\dot{\mathbf{H}}_A = \mathbf{H}_A \tilde{\boldsymbol{\eta}}_A$ and $\dot{\mathbf{H}}_B = \mathbf{H}_B \tilde{\boldsymbol{\eta}}_B$. Notice that, being a one element manipulator, we do not indicate the subscripts. In this case, the forward kinematics in (2.15) reads

$$\mathbf{H}(\alpha) = \mathbf{H}_A \mathbf{H}_{A0} \exp_{SE(3)} \left(\frac{\alpha}{L} \tilde{\mathbf{d}} \right) \quad (3.31)$$

where we used $L_0 = 0$. The time derivatives of (3.31) can be computed using the following two expressions, respectively according to the definition of the left invariant vector field and the definition of the derivatives in general

$$\dot{\mathbf{H}}(\alpha) = \mathbf{H}(\alpha) \tilde{\boldsymbol{\eta}} \quad (3.32)$$

$$\dot{\mathbf{H}}(\alpha) = \dot{\mathbf{H}}_A \mathbf{H}_{A0} \exp_{SE(3)} \left(\frac{\alpha}{L} \tilde{\mathbf{d}} \right) + \mathbf{H}_A \mathbf{H}_{A0} \frac{\alpha}{L} \mathbf{D} \left(\exp_{SE(3)} \left(\frac{\alpha}{L} \tilde{\mathbf{d}} \right) \right) \dot{\mathbf{d}} \quad (3.33)$$

Using the properties of the Adjoint representation in (A.6) and simple mathematical simplifications, we obtain

$$\boldsymbol{\eta}(\alpha) = \text{Ad}_{\exp_{SE(3)} \left(-\frac{\alpha}{L} \tilde{\mathbf{d}} \right) \mathbf{H}_{A0}^{-1}} (\boldsymbol{\eta}_A) + \frac{\alpha}{L} \mathbf{T}_{SE(3)} \left(\frac{\alpha}{L} \tilde{\mathbf{d}} \right) \dot{\mathbf{d}} \quad (3.34)$$

Equation 3.34 relates the velocity along the arm with the initial velocity and the time derivative of the deformations, since it holds $\dot{\boldsymbol{\epsilon}} = \dot{\mathbf{d}}$. In order to relate the velocity along the arm with the nodal velocities, let us compute the term $\dot{\mathbf{d}}$ as a function of the nodal velocities $\boldsymbol{\eta}_A$ and $\boldsymbol{\eta}_B$. From (3.31), the relationships between the initial and final frames read

$$\begin{aligned} \mathbf{H}_B \mathbf{H}_{B0} &= \mathbf{H}_A \mathbf{H}_{A0} \exp_{SE(3)} (\mathbf{d}) \\ \Leftrightarrow \text{Ad}_{\mathbf{H}_{B0}^{-1}} (\boldsymbol{\eta}_B) &= \text{Ad}_{\exp_{SE(3)} (-\mathbf{d}) \mathbf{H}_{A0}^{-1}} (\boldsymbol{\eta}_A) + \mathbf{T}_{SE(3)} (\mathbf{d}) \dot{\mathbf{d}} \end{aligned} \quad (3.35)$$

and

$$\begin{aligned} \mathbf{H}_A \mathbf{H}_{A0} &= \mathbf{H}_B \mathbf{H}_{B0} \exp_{SE(3)} (-\mathbf{d}) \\ \Leftrightarrow \text{Ad}_{\mathbf{H}_{A0}^{-1}} (\boldsymbol{\eta}_A) &= \text{Ad}_{\exp_{SE(3)} (\mathbf{d}) \mathbf{H}_{B0}^{-1}} (\boldsymbol{\eta}_B) - \mathbf{T}_{SE(3)} (-\mathbf{d}) \dot{\mathbf{d}} \end{aligned} \quad (3.36)$$

Since $\text{Ad}_{\exp_{SE(3)} (\mp \mathbf{d})} = \mathbf{T}_{SE(3)} (\pm \mathbf{d})$ we obtain

$$\dot{\mathbf{d}} = \mathbf{P}(\mathbf{d}) \boldsymbol{\eta}_{AB} \quad (3.37)$$

where

$$\mathbf{P}(\mathbf{d}) = [-\mathbf{T}_{SE(3)}^{-1}(-\mathbf{d})\text{Ad}_{\mathbf{H}_{A_0}^{-1}} \quad \mathbf{T}_{SE(3)}^{-1}(\mathbf{d})\text{Ad}_{\mathbf{H}_{B_0}^{-1}}] \quad (3.38)$$

Introducing (3.37) in Eq. 3.34 and considering the properties of the adjoint representation, we obtain

$$\begin{aligned} \boldsymbol{\eta}(\alpha) &= (\text{Ad}_{\exp_{SE(3)}(-\frac{\alpha}{L}\mathbf{d})} - \frac{\alpha}{L}\mathbf{T}_{SE(3)}(\frac{\alpha}{L}\mathbf{d})\mathbf{T}_{SE(3)}^{-1}(-\mathbf{d}))\text{Ad}_{\mathbf{H}_{A_0}^{-1}}\boldsymbol{\eta}_A \dots \quad (3.39) \\ &\dots + \frac{\alpha}{L}\mathbf{T}_{SE(3)}(\frac{\alpha}{L}\mathbf{d})\mathbf{T}_{SE(3)}^{-1}(\mathbf{d})\text{Ad}_{\mathbf{H}_{B_0}^{-1}}\boldsymbol{\eta}_B \end{aligned}$$

Let us sum and subtract the quantity $\frac{\alpha}{L}\mathbf{T}_{SE(3)}(\frac{\alpha}{L}\mathbf{d})\mathbf{T}_{SE(3)}^{-1}(\mathbf{d})$ in the first term at the right hand side of (3.39), such that we can rewrite it as

$$\begin{aligned} \text{Ad}_{\exp_{SE(3)}(-\frac{\alpha}{L}\mathbf{d})} + \frac{\alpha}{L}\mathbf{T}_{SE(3)}(\frac{\alpha}{L}\mathbf{d})(\mathbf{T}_{SE(3)}^{-1}(\mathbf{d}) - \mathbf{T}_{SE(3)}^{-1}(-\mathbf{d})) \dots \quad (3.40) \\ \dots - \frac{\alpha}{L}\mathbf{T}_{SE(3)}(\frac{\alpha}{L}\mathbf{d})\mathbf{T}_{SE(3)}^{-1}(\mathbf{d}) \end{aligned}$$

Now, we consider the series development of $\mathbf{T}_{SE(3)}^{-1}$, such that

$$\mathbf{T}_{SE(3)}^{-1}(\mathbf{d}) - \mathbf{T}_{SE(3)}^{-1}(-\mathbf{d}) = \sum_{i=0}^{\infty} (-1)^i B_i \frac{\widehat{\mathbf{d}}^i - \widehat{-\mathbf{d}}^i}{i!} = \sum_{i=0}^{\infty} (-1)^i B_i \frac{\widehat{\mathbf{d}}^i}{i!} (1 - (-1)^i) \quad (3.41)$$

Since $B_i = 0 \forall$ odd $i > 1$ and $(1 - (-1)^i) \neq 0$ only for odd i , the only non-vanishing term in (3.41) is for $i = 1$, that is, using $B_1 = -1/2$,

$$\mathbf{T}_{SE(3)}^{-1}(\mathbf{d}) - \mathbf{T}_{SE(3)}^{-1}(-\mathbf{d}) = \widehat{\mathbf{d}} \quad (3.42)$$

Using Eq. 3.42 and considering that $\text{Ad}_{\exp_{SE(3)}(-\frac{\alpha}{L}\mathbf{d})} = \exp_{SE(3)}(-\frac{\alpha}{L}\widehat{\mathbf{d}})$, Eq. 3.40 turns to be

$$\exp_{SE(3)}(-\frac{\alpha}{L}\widehat{\mathbf{d}}) + \frac{\alpha}{L}\mathbf{T}_{SE(3)}(\frac{\alpha}{L}\mathbf{d})\widehat{\mathbf{d}} - \frac{\alpha}{L}\mathbf{T}_{SE(3)}(\frac{\alpha}{L}\mathbf{d})\mathbf{T}_{SE(3)}^{-1}(\mathbf{d}) \quad (3.43)$$

The first term of (3.43) can be expressed using the series development of the exponential map as

$$\begin{aligned} \exp_{SE(3)}(-\frac{\alpha}{L}\widehat{\mathbf{d}}) &= \sum_{i=0}^{\infty} (-1)^i \left(\frac{\alpha}{L}\right)^i \frac{\widehat{\mathbf{d}}^i}{(i+1)!} = \mathbf{I} + \sum_{i=0}^{\infty} (-1)^i \left(\frac{\alpha}{L}\right)^i \frac{\widehat{\mathbf{d}}^i}{i!} \quad (3.44) \\ &= \mathbf{I} + \sum_{i=0}^{\infty} (-1)^{i+1} \left(\frac{\alpha}{L}\right)^{i+1} \frac{\widehat{\mathbf{d}}^{i+1}}{(i+1)!} \end{aligned}$$

The second term of (3.43) can be expressed using the series development of the tangent application as

$$\begin{aligned} \frac{\alpha}{L}\mathbf{T}_{SE(3)}(\frac{\alpha}{L}\mathbf{d})\widehat{\mathbf{d}} &= \frac{\alpha}{L} \left(\sum_{i=0}^{\infty} (-1)^i \left(\frac{\alpha}{L}\right)^i \frac{\widehat{\mathbf{d}}^i}{(i+1)!} \right) \widehat{\mathbf{d}} \quad (3.45) \\ &= \sum_{i=0}^{\infty} (-1)^i \left(\frac{\alpha}{L}\right)^{i+1} \frac{\widehat{\mathbf{d}}^{i+1}}{(i+1)!} \end{aligned}$$

Summing the two expressions in (3.44) and (3.45) leads to

$$\mathbf{I} + \sum_{i=0}^{\infty} ((-1)^{i+1} + (-1)^i) \left(\frac{\alpha}{L}\right)^{i+1} \frac{\widehat{\mathbf{d}}^{i+1}}{(i+1)!} = \mathbf{I} \quad (3.46)$$

since $((-1)^{i+1} + (-1)^i) = 0 \forall i$. Thus, Eq. 3.39 turns to be

$$\boldsymbol{\eta}(\alpha) = \mathbf{Q}(\alpha, \mathbf{d})\boldsymbol{\eta}_{AB} \quad (3.47)$$

where

$$\mathbf{Q}(\alpha, \mathbf{d}) = [(\mathbf{I}_{6 \times 6} - \mathbf{T}^*(\alpha, \mathbf{d}))\text{Ad}_{\mathbf{H}_{A0}^{-1}} \quad \mathbf{T}^*(\alpha, \mathbf{d})\text{Ad}_{\mathbf{H}_{B0}^{-1}}] \quad (3.48)$$

and

$$\mathbf{T}^*(\alpha, \mathbf{d}) = \frac{\alpha}{L} \mathbf{T}_{SE(3)} \left(\frac{\alpha}{L} \mathbf{d}\right) \mathbf{T}_{SE(3)}^{-1}(\mathbf{d}). \quad (3.49)$$

Using Eq. 3.37 and Eq. 3.47, and considering that $\dot{\boldsymbol{\epsilon}} = \dot{\mathbf{d}}$, we obtain

$$\boldsymbol{\eta}(\alpha) = \mathbf{J}(\alpha, \mathbf{d})\dot{\boldsymbol{\epsilon}} \quad (3.50)$$

where

$$\begin{aligned} \mathbf{J} &= L\mathbf{Q}\mathbf{P}^{-1} \\ &= L \left((\mathbf{T}^*(\alpha, \mathbf{d}) - \mathbf{I}_{6 \times 6}) \mathbf{T}_{SE(3)}(-\mathbf{d}) + \mathbf{T}^*(\alpha, \mathbf{d}) \mathbf{T}_{SE(3)}(\mathbf{d}) \right) \end{aligned} \quad (3.51)$$

generic n

Let us consider a generic manipulator composed by n elements. In order to achieve the general form of the soft geometric Jacobian, we need to introduce a $12n \times 1$ velocity vector $\boldsymbol{\eta}_{AB} = [\boldsymbol{\eta}_{1A}^T \quad \boldsymbol{\eta}_{2A}^T \cdots \boldsymbol{\eta}_{nA}^T \quad \boldsymbol{\eta}_{1B}^T \quad \boldsymbol{\eta}_{2B}^T \cdots \boldsymbol{\eta}_{nB}^T]$ from the nodal velocities $\dot{\mathbf{H}}_{1A} = \mathbf{H}_{1A} \tilde{\boldsymbol{\eta}}_{1A}$, \dots , $\dot{\mathbf{H}}_{nB} = \mathbf{H}_{nB} \tilde{\boldsymbol{\eta}}_{nB}$. The PoE formula in (2.15) for the n -th element reads

$$\mathbf{H}_n(\alpha_n) = \mathbf{H}_{1A} \mathbf{H}_{1A0} \left(\prod_{i=1}^{n-1} \exp_{SE(3)} \left(\frac{L_i - L_{i-1}}{L_n} \tilde{\mathbf{d}}_i \right) \right) \exp_{SE(3)} \left(\frac{\alpha_n - L_{n-1}}{L_n} \tilde{\mathbf{d}}_n \right) \quad (3.52)$$

By taking the derivative of (3.52) and using the properties of the Adjoint representation in A.6, after some mathematical computations we obtain

$$\begin{aligned} \boldsymbol{\eta}_n(\alpha_n) &= \text{Ad}_{\exp_{SE(3)}(-\sum_{i=1}^{n-1} \frac{L_j - L_{j-1}}{L_n} \mathbf{d}_i) \exp_{SE(3)}(-\frac{\alpha_n - L_{n-1}}{L_n} \mathbf{d}_n) \mathbf{H}_{1A0}^{-1}} (\boldsymbol{\eta}_{1A}) \cdots \\ &\cdots + \text{Ad}_{\exp_{SE(3)}(-\frac{\alpha_n - L_{n-1}}{L_n} \mathbf{d}_n)} \sum_{i=1}^{n-1} \frac{L_i - L_{i-1}}{L_n} \left(\mathbf{T}_{SE(3)} \left(\frac{L_i - L_{i-1}}{L_n} \mathbf{d}_i \right) \dot{\mathbf{d}}_i \right) \cdots \\ &\cdots + \frac{\alpha_n - L_{n-1}}{L_n} \left(\mathbf{T}_{SE(3)} \left(\frac{\alpha_n - L_{n-1}}{L_n} \mathbf{d}_n \right) \dot{\mathbf{d}}_n \right) \end{aligned} \quad (3.53)$$

Equation 3.53 relates the velocity of the n -th element of the arm with the initial velocity of the first element, with all the set of deformations \mathbf{d}_i and the deformations rate $\dot{\mathbf{d}}_i$. Using the same procedure that we used above for the one element manipulator, we can obtain the relationship between the deformations rate and the nodal velocities for a n - elements manipulator as

$$\begin{bmatrix} \dot{\mathbf{d}}_1 \\ \dot{\mathbf{d}}_2 \\ \vdots \\ \dot{\mathbf{d}}_n \end{bmatrix} = \begin{bmatrix} \mathbf{P}_1^-(\mathbf{d}_1) & \mathbf{0}_{6 \times 6} & \dots & \mathbf{0}_{6 \times 6} & \mathbf{P}_1^+(\mathbf{d}_1) & \mathbf{0}_{6 \times 6} & \dots & \mathbf{0}_{6 \times 6} \\ \mathbf{0}_{6 \times 6} & \mathbf{P}_2^-(\mathbf{d}_2) & \dots & \mathbf{0}_{6 \times 6} & \mathbf{0}_{6 \times 6} & \mathbf{P}_2^+(\mathbf{d}_2) & \dots & \mathbf{0}_{6 \times 6} \\ \vdots & \vdots \\ \mathbf{0}_{6 \times 6} & \mathbf{0}_{6 \times 6} & \dots & \mathbf{P}_n^-(\mathbf{d}_n) & \mathbf{0}_{6 \times 6} & \mathbf{0}_{6 \times 6} & \dots & \mathbf{P}_n^+(\mathbf{d}_n) \end{bmatrix} \begin{bmatrix} \eta_{1A} \\ \eta_{2A} \\ \vdots \\ \eta_{nA} \\ \eta_{1B} \\ \eta_{2B} \\ \vdots \\ \eta_{nB} \end{bmatrix} \quad (3.54)$$

where

$$\mathbf{P}_j^-(\mathbf{d}_j) = -\frac{L_j - L_{j-1}}{L_n} \mathbf{T}_{SE(3)}^{-1} \left(-\frac{L_j - L_{j-1}}{L_n} \mathbf{d}_j \right) \text{Ad}_{\mathbf{H}_{jA0}^{-1}} \quad (3.55)$$

$$\mathbf{P}_j^+(\mathbf{d}_j) = +\frac{L_j - L_{j-1}}{L_n} \mathbf{T}_{SE(3)}^{-1} \left(\frac{L_j - L_{j-1}}{L_n} \mathbf{d}_j \right) \text{Ad}_{\mathbf{H}_{jB0}^{-1}} \quad (3.56)$$

are the 6×6 block matrices that appears in the diagonals of (3.54). Introducing (3.54) in Eq. 3.53, we obtain $\boldsymbol{\eta}_n(\alpha_n)$ as a function of the nodal velocities. The relationship between the velocity of the end element and the set of deformations rate, after some mathematical computations, results

$$\begin{aligned} \boldsymbol{\eta}_n(\alpha_n) &= \text{Ad}_{\exp_{SE(3)}\left(-\frac{\alpha_n - L_{n-1}}{L_n} \mathbf{d}_n\right)} \text{Ad}_{\exp_{SE(3)}\left(-\sum_{i=1}^{n-1} \frac{L_i - L_{i-1}}{L_n} \mathbf{d}_i\right)} \mathbf{J}_1(L_1, \mathbf{d}_1)(\dot{\boldsymbol{\epsilon}}_1) \dots \\ &\dots + \text{Ad}_{\exp_{SE(3)}\left(-\frac{\alpha_n - L_{n-1}}{L_n} \mathbf{d}_n\right)} \text{Ad}_{\exp_{SE(3)}\left(-\sum_{i=2}^{n-1} \frac{L_i - L_{i-1}}{L_n} \mathbf{d}_i\right)} \mathbf{J}_2(L_2, \mathbf{d}_2)(\dot{\boldsymbol{\epsilon}}_2) \dots \\ &\dots + \dots + \text{Ad}_{\exp_{SE(3)}\left(-\frac{\alpha_n - L_{n-1}}{L_n} \mathbf{d}_n\right)} \mathbf{J}_{n-1}(L_{n-1}, \mathbf{d}_{n-1})(\dot{\boldsymbol{\epsilon}}_{n-1}) \dots \\ &\dots + \mathbf{J}_n(\alpha_n, \mathbf{d}_n)(\dot{\boldsymbol{\epsilon}}_n) \end{aligned} \quad (3.57)$$

where the generic element \mathbf{J}_j is given by

$$\begin{aligned} \mathbf{J}_j(\min(\alpha_j, L_j), \mathbf{d}_j) &= (\mathbf{T}_j^*(\min(\alpha_j, L_j), \mathbf{d}_j) - \mathbf{I}_{6 \times 6}) \mathbf{T}_{SE(3)} \left(-\frac{L_j - L_{j-1}}{L_n} \mathbf{d}_j \right) \dots \\ &\dots + \mathbf{T}_j^*(\min(\alpha_j, L_j), \mathbf{d}_j) \mathbf{T}_{SE(3)} \left(\frac{L_j - L_{j-1}}{L_n} \mathbf{d}_j \right) \end{aligned} \quad (3.58)$$

and

$$\begin{aligned} \mathbf{T}_j^*(\min(\alpha_j, L_j), \mathbf{d}_j) &= \frac{\min(\alpha_j, L_j) - L_{j-1}}{L_n} \mathbf{T}_{SE(3)} \left(\frac{\min(\alpha_j, L_j) - L_{j-1}}{L_n} \mathbf{d}_j \right) \dots \\ &\dots \mathbf{T}_{SE(3)}^{-1} \left(\frac{L_j - L_{j-1}}{L_n} \mathbf{d}_j \right) \end{aligned} \quad (3.59)$$

The differential kinematics for the piecewise constant deformation soft manipulator reads

$$\boldsymbol{\eta}(\alpha) = \mathbb{J}(\alpha, \mathbf{d}) \dot{\boldsymbol{\epsilon}} \quad (3.60)$$

where $\mathbb{J}(\alpha, \mathbf{d})$ is the $6n \times 6n$ matrix representing the relationship between the velocity along the arm and the time derivative of the discretized strains. The action of \mathbb{J} on the state vector $\dot{\boldsymbol{\epsilon}}$ returns the arm velocity $\boldsymbol{\eta}(\alpha)$, which is expressed in the local coordinate frame. Hence, we refer to this matrix as the *soft geometric Jacobian* of the piecewise constant deformation arm. It is given in Table 3.1.

$$\mathbb{J}(\boldsymbol{\alpha}, \mathbf{d}) = \begin{bmatrix} \mathbf{J}_1(\boldsymbol{\alpha}_1, \mathbf{d}_1) & \mathbf{0}_{6 \times 6} & \dots & \mathbf{0}_{6 \times 6} \\ \text{Ad}_{\exp_{SE(3)}\left(-\frac{\alpha_2 - L_1}{L_n} \mathbf{d}_2\right)} \mathbf{J}_1(L_1, \mathbf{d}_1) & \mathbf{J}_2(\boldsymbol{\alpha}_2, \mathbf{d}_2) & \dots & \mathbf{0}_{6 \times 6} \\ \text{Ad}_{\exp_{SE(3)}\left(-\frac{\alpha_3 - L_2}{L_n} \mathbf{d}_3 - \frac{L_2 - L_1}{L_n} \mathbf{d}_2\right)} \mathbf{J}_1(L_1, \mathbf{d}_1) & \text{Ad}_{\exp_{SE(3)}\left(-\frac{\alpha_3 - L_2}{L_n} \mathbf{d}_3\right)} \mathbf{J}_2(L_2, \mathbf{d}_2) & \mathbf{J}_3(\boldsymbol{\alpha}_3, \mathbf{d}_3) & \mathbf{0}_{6 \times 6} \\ \vdots & \vdots & \ddots & \vdots \\ \text{Ad}_{\exp_{SE(3)}\left(-\frac{\alpha_n - L_{n-1}}{L_n} \mathbf{d}_n - \sum_{i=1}^{L_i - L_{i-1}} \frac{L_i - L_{i-1}}{L_n} \mathbf{d}_i\right)} \mathbf{J}_1(L_1, \mathbf{d}_1) & \dots & \dots & \mathbf{J}_n(\boldsymbol{\alpha}_n, \mathbf{d}_n) \end{bmatrix}$$

Table 3.1: Geometric Jacobian of the piecewise constant deformation soft manipulator.

Acceleration

By taking the time derivative of (3.60), we obtain the discrete model of acceleration:

$$\dot{\boldsymbol{\eta}}(\alpha) = \mathbb{J}(\alpha, \mathbf{d})\ddot{\boldsymbol{\epsilon}} + \dot{\mathbb{J}}(\alpha, \mathbf{d})\dot{\boldsymbol{\epsilon}} \quad (3.61)$$

3.2.2 Inverse differential kinematics

Since the soft geometric Jacobian matrix of a piecewise constant deformation arm is a *squared* and *full rank* $6n \times 6n$ matrix, we can invert Eq. 3.60 such that

$$\dot{\boldsymbol{\epsilon}} = \mathbb{J}(\alpha, \mathbf{d})^{-1}\dot{\boldsymbol{\eta}}(\alpha) \quad (3.62)$$

and the strain field $\boldsymbol{\epsilon}$ can be calculated by integrating (3.62) over time. In the same manner, we can invert Eq. 3.61 such that

$$\ddot{\boldsymbol{\epsilon}} = \mathbb{J}(\alpha, \mathbf{d})^{-1}(\dot{\boldsymbol{\eta}}(\alpha) - \dot{\mathbb{J}}(\alpha, \mathbf{d})\dot{\boldsymbol{\epsilon}}) \quad (3.63)$$

and the strain field $\boldsymbol{\epsilon}$ can be calculated by double integration of (3.63) over time.

3.2.3 Statics

The static equilibrium equations are obtained by recalling the principle of virtual work (3.8). In order to apply that principle, we need to compute the discretized variation of the expression of the internal energy in Eq. 3.9 and the discretized variation of the expression of the external energy in Eq. 3.12.

Considering a linear elastic material, the discretized variation of the internal energy for an n - elements piecewise constant deformation arm reads

$$\delta(\mathcal{V}_{int}) = \delta(\boldsymbol{\epsilon}_1)^T \mathbf{K}_1^{l_1} \boldsymbol{\epsilon}_1 + \delta(\boldsymbol{\epsilon}_2)^T \mathbf{K}_2^{l_2} \boldsymbol{\epsilon}_2 + \cdots + \delta(\boldsymbol{\epsilon}_n)^T \mathbf{K}_n^{l_n} \boldsymbol{\epsilon}_n \quad (3.64)$$

where we used the fact the the strains, and thus, its variations are piecewise constant. For each element, $\mathbf{K}_i^{l_i}$ is the discretized stiffness matrix defined as

$$\mathbf{K}_i^{l_i} = \int_{l_{i-1}}^{l_i} \mathbf{K}(\alpha_i) d\alpha_i \quad (3.65)$$

Introducing the $6n \times 6n$ diagonal discretized stiffness matrix for all the elements as

$$\mathbf{K}_L = \text{diag}(\mathbf{K}_i^{l_i}), \quad i = 1, \dots, n \quad (3.66)$$

Equation 3.64 becomes

$$\delta(\mathcal{V}_{int}) = \delta(\boldsymbol{\epsilon})^T \mathbf{K}_L \boldsymbol{\epsilon} \quad (3.67)$$

For the discretized variation of the external energy, we need to compute the discretized variation of \mathbf{h} . To that purpose, the differential kinematic equations in (3.60) at the variation level read

$$\delta \mathbf{h}(\alpha) = \mathbb{J}(\alpha, \mathbf{d}) \delta(\boldsymbol{\epsilon}) \quad (3.68)$$

Let us indicate with $\mathbb{J}_{ij}(\alpha_i, \mathbf{d}_i)$ the 6×6 block-element of block-row i and block-column j of the soft geometric Jacobian as

$$\mathbb{J}(\alpha, \mathbf{d}) = \begin{bmatrix} \mathbb{J}_{11}(\alpha_1, \mathbf{d}_1) & \mathbf{0}_{6 \times 6} & & \dots & \mathbf{0}_{6 \times 6} \\ \mathbb{J}_{21}(\alpha_2, \mathbf{d}_1) & \mathbb{J}_{22}(\alpha_2, \mathbf{d}_2) & \mathbf{0}_{6 \times 6} & \dots & \mathbf{0}_{6 \times 6} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbb{J}_{n1}(\alpha_n, \mathbf{d}_n) & \dots & \dots & \dots & \mathbb{J}_{nn}(\alpha_n, \mathbf{d}_n) \end{bmatrix} \quad (3.69)$$

Introducing (3.68) in Eq. 3.12, we obtain

$$\begin{aligned} \delta(\mathcal{V}_{ext}) &= -[\delta(\boldsymbol{\epsilon}_1)]^T \int_0^{L_1} \mathbb{J}_{11}^T(\alpha_1, \mathbf{d}_1) \mathbf{g}_{ext,1} d\alpha_1 \dots \\ &\dots + \delta(\boldsymbol{\epsilon}_2)^T \int_{L_1}^{L_2} \mathbb{J}_{21}^T(\alpha_2, \mathbf{d}_2) \mathbf{g}_{ext,1} + \mathbb{J}_{22}^T(\alpha_2, \mathbf{d}_2) \mathbf{g}_{ext,2} d\alpha_2 \dots \\ &\dots + \dots + \delta(\boldsymbol{\epsilon}_n)^T \int_{L_{n-1}}^{L_n} \mathbb{J}_{n1}^T(\alpha_n, \mathbf{d}_n) \mathbf{g}_{ext,1} + \dots + \mathbb{J}_{nn}^T(\alpha_n, \mathbf{d}_n) \mathbf{g}_{ext,n} d\alpha_n \end{aligned} \quad (3.70)$$

where we used the fact that the the strains, and thus, its variations are piecewise constant. Let us indicate with $\mathbb{J}_{ij}^{l_i}(\mathbf{d}_i)$ the solution of each integral in (3.70) as

$$\mathbb{J}_{ij}^{l_i}(\mathbf{d}_i) = \int_{l_{i-1}}^{l_i} \mathbb{J}_{ij}(\alpha_i, \mathbf{d}_i) d\alpha_i \quad (3.71)$$

such that Eq. 3.70 becomes

$$\begin{aligned} \delta(\mathcal{V}_{ext}) &= -[\delta(\boldsymbol{\epsilon}_1)]^T \mathbb{J}_{11}^{T,l_1}(\mathbf{d}_1) \mathbf{F}_1^{l_1} + \delta(\boldsymbol{\epsilon}_2)^T \sum_{j=1}^2 \mathbb{J}_{2j}^{T,l_2}(\mathbf{d}_2) \mathbf{F}_j^{l_2} \dots \\ &\dots + \dots + \delta(\boldsymbol{\epsilon}_n)^T \sum_{j=1}^n \mathbb{J}_{nj}^{T,l_n}(\mathbf{d}_n) \mathbf{F}_j^{l_n} \end{aligned} \quad (3.72)$$

where $\mathbf{F}_i^{l_j}$ is the integral of the external forces acting on the element of length l_j as

$$\mathbf{F}_i^{l_j} = \int_{l_{j-1}}^{l_j} \mathbf{g}_{ext,i} d\alpha_i \quad (3.73)$$

Let us introduce the soft geometric Jacobian computed over the length of the arm \mathbb{J}_L as

$$\mathbb{J}_L(\mathbf{d}) = \int_L \mathbb{J}(\alpha, \mathbf{d}) d\alpha \quad (3.74)$$

where $L = L_n$ is the total length of the piecewise constant deformation arm and \mathbb{J} is the soft geometric Jacobian defined in Table 3.1.

Finally, the discretized variation of the external energy for a n - elements piecewise constant deformation arm becomes

$$\delta(\mathcal{V}_{ext}) = \delta(\boldsymbol{\epsilon})^T \mathbb{J}_L^T \mathcal{F} \quad (3.75)$$

where \mathcal{F} indicates the vector of external forces as $\mathcal{F} = [\mathbf{F}^{T,l_1}, \mathbf{F}^{T,l_2}, \dots, \mathbf{F}^{T,l_n}]^T$. According to the principle of virtual work, the manipulator is in static equilibrium if and only if

$$\delta(\mathcal{V}_{int}) = \delta(\mathcal{V}_{ext}) \quad \forall \delta(\boldsymbol{\epsilon}) \quad (3.76)$$

Hence, substituting (3.67) and (3.75) into Eq. 3.88 leads to the notable result

$$\mathcal{T} = \mathbb{J}_L^T(\mathbf{d})\mathcal{F} \quad (3.77)$$

stating that the relationship between the external forces \mathcal{F} and the internal forces $\mathcal{T} = \mathbf{K}_L \boldsymbol{\epsilon}$, is established by the transpose of the soft geometric Jacobian computed over the length of the arm.

3.2.4 Special case: constant deformation soft arm

In this Section we particularize, for the constant deformation model, the results achieved for the piecewise constant deformation model.

Soft geometric Jacobian

The velocity kinematics in (3.60) for a constant deformation arm becomes

$$\boldsymbol{\eta}(\alpha) = \mathbf{J}(\alpha, \mathbf{d})\dot{\boldsymbol{\epsilon}} \quad (3.78)$$

where

$$\begin{aligned} \mathbf{J} &= L\mathbf{Q}\mathbf{P}^{-1} \\ &= L \left((\mathbf{T}^*(\alpha, \mathbf{d}) - \mathbf{I}_{6 \times 6}) \mathbf{T}_{SE(3)}(-\mathbf{d}) + \mathbf{T}^*(\alpha, \mathbf{d}) \mathbf{T}_{SE(3)}(\mathbf{d}) \right) \end{aligned} \quad (3.79)$$

is the 6×6 matrix representing the relationship between the velocity along the soft robotic arm and the time derivative of the discretized strains. The action of \mathbf{J} on the state vector $\dot{\boldsymbol{\epsilon}}$ returns the arm velocity $\boldsymbol{\eta}(\alpha)$, which is expressed in the local coordinate frame. Hence, we refer to this matrix as the *soft geometric Jacobian* of the constant deformation arm.

The discrete acceleration model in (3.61) for a constant deformation arm reads

$$\dot{\boldsymbol{\eta}}(\alpha) = \mathbf{J}(\alpha, \mathbf{d})\ddot{\boldsymbol{\epsilon}} + \dot{\mathbf{J}}(\alpha, \mathbf{d})\dot{\boldsymbol{\epsilon}} \quad (3.80)$$

Statics

The discretized variation of the internal energy in (3.64) for a constant deformation arm reads

$$\delta(\mathcal{V}_{int}) = \delta(\boldsymbol{\epsilon})^T \mathbf{K}_L \boldsymbol{\epsilon} \quad (3.81)$$

where \mathbf{K}_L is the discretized stiffness matrix defined as

$$\mathbf{K} = \int_L \mathbf{K}(\alpha) d\alpha \quad (3.82)$$

For the discretized variation of the external energy, we need to compute the discretized variation of \mathbf{h} . To that purpose, Eq. 3.78 at the variation level reads

$$\delta \mathbf{h}(\alpha) = \mathbf{J}(\alpha, \mathbf{d}) \delta(\boldsymbol{\epsilon}) \quad (3.83)$$

Introducing (3.83) in Eq. 3.12, we obtain

$$\delta(\mathcal{V}_{ext}) = -\delta(\boldsymbol{\epsilon})^T \mathbf{J}_L^T(\mathbf{d}) \mathbf{F} \quad (3.84)$$

where \mathbf{F} is defined as

$$\mathbf{F} = \int_L \mathbf{g}_{ext} d\alpha \quad (3.85)$$

and \mathbf{J}_L , the soft geometric Jacobian computed over the length of the arm, is defined as

$$\mathbf{J}_L(\mathbf{d}) = \int_L \mathbf{J}(\alpha, \mathbf{d}) d\alpha \quad (3.86)$$

According to the principle of virtual work, the manipulator is in static equilibrium if and only if

$$\delta(\mathcal{V}_{int}) = \delta(\mathcal{V}_{ext}) \quad \forall \delta(\boldsymbol{\epsilon}) \quad (3.87)$$

Hence, substituting (3.81) and (3.84) into Eq. 3.87 leads to the notable result

$$\mathbf{T} = \mathbf{J}_L^T(\mathbf{d}) \mathbf{F} \quad (3.88)$$

stating that the relationship between the external forces \mathbf{F} and the internal forces $\mathbf{T} = \mathbf{K}_L \boldsymbol{\epsilon}$ is established by the transpose of the soft geometric Jacobian computed over the length of the element.

3.2.5 Example

Let us consider the example in Sec. 3.1.6. The physical parameters used in the simulations are (from [58])

- $L = 0.5$ m
- $EI = 1$ Nm²
- $\tau = 1-10$ N m

Figure 3.2 illustrates the soft continuum arm in its initial and deformed configuration, after applying the torque at its free hand. The gray solid lines indicate the analytic solutions for the position field obtained in Sec. 3.1.6, while the solutions for the discrete position field are indicated by the dotted red lines. We notice that the current discretization matches the exact solution. This is valid for pure bending/torsion cases.

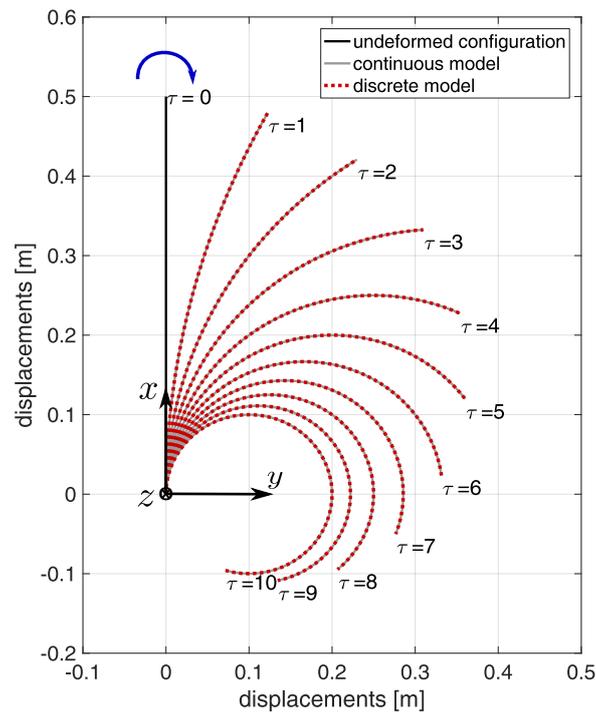


Figure 3.2: Cantilevered soft arm subject to a torque τ at its free end. The initial configuration is a straight vertical line, while the deformed configuration is a planar curve of constant curvature.

3.3 Summary

- The *differential kinematics* of a piecewise constant deformation soft arm is given by

$$\boldsymbol{\eta}(\alpha) = \mathbb{J}(\alpha, \mathbf{d})\dot{\epsilon} \quad (3.89)$$

- The *statics* of a piecewise constant deformation soft arm reads

$$\mathcal{T} = \mathbb{J}_L^T(\mathbf{d})\mathcal{F} \quad (3.90)$$

Chapter 4

Dynamics

Dynamics regards to the motion of a robotic mechanism by accounting to the forces and torques that cause it. The derivation of the dynamic model of a robotic manipulator is crucial for mechanical design, analysis of manipulator structures, design of control algorithms, simulation of motion.

In this chapter, we use the principles of the Hamiltonian mechanics [59] and the variational calculus [49] to derive the equations of motion of a continuum arm and a piecewise constant deformation arm.

4.1 Continuum arm

The dynamic equilibrium equations of a continuum arm take the form of nonlinear partial differential equations formulated on a Lie group.

4.1.1 Kinetic energy

The kinetic energy of the continuum arm is defined by

$$\mathcal{K} = \frac{1}{2} \int_L \boldsymbol{\eta}^T \mathbf{M} \boldsymbol{\eta} \, d\alpha \quad (4.1)$$

where \mathbf{M} is the constant 6×6 inertia matrix, which contains the mass and rotation inertia properties of the cross-sections as

$$\mathbf{M} = \begin{bmatrix} \rho A \mathbf{I}_{3 \times 3} & \mathbf{J}_I^T \\ \mathbf{J}_I & \mathbf{J}_{II} \end{bmatrix} \quad (4.2)$$

in which \mathbf{J}_I and \mathbf{J}_{II} are the first and the second moment of inertia of the cross-sections computed in the local axes of the arm, ρ is the density and A is the cross-section area. In the general case, \mathbf{M} is not diagonal. Indeed, if the reference curve is defined such that $\mathbf{J}_I = \mathbf{0}_{3 \times 3}$, and if the normal and the bi-normal to the reference curve are the principal axes of the cross sections such that \mathbf{J}_{II} is diagonal, then \mathbf{M} is a diagonal matrix. Since the cross-sections are assumed to be undeformable, \mathbf{M} is defined from the initial configuration of the continuum arm and it does not depend on its motion or deformation.

4.1.2 Hamiltonian formulation

The Hamilton's principle states that the *action integral* over the time interval $[t_0, t_1]$ is stationary provided that the initial and final configurations are fixed, i.e.,

$$\int_{t_0}^{t_1} (\delta(\mathcal{K}) - \delta(\mathcal{V}_{int}) + \delta(\mathcal{V}_{ext})) dt = 0. \quad (4.3)$$

where the variations are fixed at t_0 and t_1 . In (4.3), \mathcal{K} denotes the kinetic energy, while \mathcal{V}_{int} and \mathcal{V}_{ext} denote respectively the potential energy due to the internal and external forces. The variation of the strain energy has been already computed in Eq. 3.11, while the variation and the external forces in Eq. 3.12. The variation of the kinetic energy of Eq. 4.1 reads

$$\delta(\mathcal{K}) = \int_L \delta(\boldsymbol{\eta})^T \mathbf{M} \boldsymbol{\eta} d\alpha \quad (4.4)$$

By recalling the commutativity of the Lie derivatives in (A.9), the variation of the velocity in terms of the variation of the configuration variables is expressed as

$$\delta(\boldsymbol{\eta}) = (\delta \dot{\mathbf{h}}) + \hat{\boldsymbol{\eta}} \delta \mathbf{h} \quad (4.5)$$

Inserting (4.5) into Eq. 4.4 and integrating by parts yields

$$\int_{t_0}^{t_1} \delta(\mathcal{K}) dt = \left[\int_L \delta \mathbf{h}^T \mathbf{M} \boldsymbol{\eta} d\alpha \right]_{t_0}^{t_1} - \int_{t_0}^{t_1} \int_L \delta \mathbf{h}^T (\mathbf{M} \dot{\boldsymbol{\eta}} - \hat{\boldsymbol{\eta}}^T \mathbf{M} \boldsymbol{\eta}) d\alpha dt \quad (4.6)$$

Since the variations are fixed, the first term on the right hand side vanishes. Finally, by combining Eq. 4.6, Eq. 3.11 and 3.12, the Hamilton's principle in (4.3) yields the following weak form of the dynamic equilibrium equations

$$[\delta \mathbf{h}^T (\mathbf{r} - \mathbf{g}_{ext})] \Big|_0^L - \int_L \delta \mathbf{h}^T (-\mathbf{M} \dot{\boldsymbol{\eta}} + \hat{\boldsymbol{\eta}}^T \mathbf{M} \boldsymbol{\eta} + \mathbf{r}' - \hat{\mathbf{f}}^T \mathbf{r} + \mathbf{g}_{ext}) d\alpha = 0 \quad (4.7)$$

Indeed, the strong form of the dynamic equations of the continuum arm reads

$$\mathbf{M} \dot{\boldsymbol{\eta}} - \hat{\boldsymbol{\eta}}^T \mathbf{M} \boldsymbol{\eta} - \mathbf{r}' + \hat{\mathbf{f}}^T \mathbf{r} = \mathbf{g}_{ext} \quad (4.8)$$

4.1.3 Equations of motion

- Kinematic equations

$$\dot{\mathbf{H}} = \mathbf{H} \tilde{\boldsymbol{\eta}} \quad (4.9)$$

$$\mathbf{H}' = \mathbf{H} (\mathbf{f}^0 + \boldsymbol{\epsilon}) \tilde{\boldsymbol{\eta}} \quad (4.10)$$

- Boundary conditions

$$\delta \mathbf{h}(L) (\mathbf{K}(L) \boldsymbol{\epsilon}(L) - \mathbf{g}_{ext}(L)) - \delta \mathbf{h}(0) (\mathbf{K}(0) \boldsymbol{\epsilon}(0) - \mathbf{g}_{ext}(0)) = 0 \quad (4.11)$$

- Dynamical equilibrium equations

$$\mathbf{M}\dot{\boldsymbol{\eta}} - \hat{\boldsymbol{\eta}}^T \mathbf{M}\boldsymbol{\eta} - \mathbf{r}' + \mathbf{f}^T \mathbf{r} = \mathbf{g}_{ext} \quad (4.12)$$

- Compatibility equations

$$\boldsymbol{\eta}' - \dot{\boldsymbol{\epsilon}} = \hat{\boldsymbol{\eta}}\mathbf{f} \quad (4.13)$$

4.1.4 Example

Let us consider the soft arm in Fig. 4.1 which is free to rotate at a constant velocity ω_0 in its own plane. The arm is initially straight, i.e., $\mathbf{f}_u^0 = \mathbf{e}_1$ and $\mathbf{f}_\omega^0 = \mathbf{0}_{3 \times 1}$. The coordinate along the arm, α , ranges from $[0, L]$, being L the length of the arm. The mass and stiffness matrices of the cross-sections are given respectively by $\mathbf{M} = \rho \text{diag}(A, A, A, J, I_2, I_3)$ and $\mathbf{K} = \text{diag}(EA, GA_2, GA_3, GJ, EI_2, EI_3)$. The arm is not subjected to any external loads, i.e., $\mathbf{g}_{ext} = \mathbf{0}_{6 \times 1}$. The boundary conditions in Eq. 4.11 becomes

$$\boldsymbol{\epsilon}(L, t) = \boldsymbol{\epsilon}(0, t) = \mathbf{0}_{6 \times 1} \quad (4.14)$$

Due to the centrifugal forces, it is expected that the arm is stretched. Accordingly, we have $\boldsymbol{\gamma} = [\gamma_1 \ 0 \ 0]^T$, $\boldsymbol{\kappa} = \mathbf{0}_{3 \times 1}$, $\mathbf{v} = [0 \ 0 \ v_3]^T$, $\boldsymbol{\omega} = [0 \ \omega \ 0]^T$.

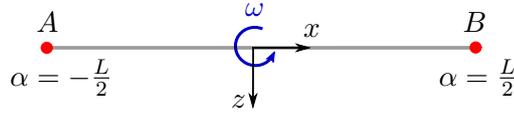


Figure 4.1: Schematic model of the free-rotating soft arm.

Deformation and velocity fields

The deformation and velocity fields are obtained by solving the dynamics and compatibility equations.

The dynamic equations in Eq. 4.12 becomes

$$\rho A v_3 \omega - EA \gamma_1' = 0 \quad (4.15)$$

$$\rho A \dot{v}_3 = 0 \quad (4.16)$$

$$J \dot{\omega} = 0 \quad (4.17)$$

The compatibility equations in Eq. 4.13 becomes

$$\dot{\gamma}_1 = 0 \quad (4.18)$$

$$v_3' + \omega(1 + \gamma_1) = 0 \quad (4.19)$$

$$\omega' = 0 \quad (4.20)$$

Eq. 4.17 and Eq. 4.20 lead to

$$\omega(\alpha, t) = \omega_0 \quad (4.21)$$

Deriving (4.15) with respect to α and replacing the expression of v'_3 from (4.19) yield

$$E\gamma_1'' + \rho\omega_0^2\gamma_1 = -\rho\omega_0^2 \quad (4.22)$$

The solution of (4.22), setting $c_0 = \sqrt{\rho/E}$, is

$$\gamma_1(\alpha) = a\cos(c_0\omega_0(\alpha + b)) - 1 \quad (4.23)$$

where a and b are two constants of integration. By using the boundary conditions $\gamma_1(0) = \gamma_1(L) = 0$, we obtain

$$a = \frac{1}{\cos(c_0\omega_0 L/2)}; \quad b = -L/2 \quad (4.24)$$

Inserting the resulting expression of γ_1 into Eq. 4.15, we obtain the value of v_3 as

$$v_3 = -\frac{a}{c_0}\sin(c_0\omega_0(\alpha + b)) \quad (4.25)$$

Finally, the expressions for v_3 and γ_1 are given by

$$v_3(\alpha, t) = \frac{\sin(c_0\omega_0(L/2 - \alpha))}{c_0\cos(c_0\omega_0 L/2)} \quad (4.26)$$

$$\gamma_1(\alpha, t) = \frac{\cos(c_0\omega_0(\alpha - L/2))}{\cos(c_0\omega_0 L/2)} - 1 \quad (4.27)$$

Notice that $v_3(L/2) = 0$: this means that the arm rotates about a fixed point, namely the material point at its mid-span.

$SE(3)$ field

The kinematic equations in Eq. 4.9 lead to

$$\mathbf{R}(t) = \mathbf{R}_0 \exp_{SO(3)}(\boldsymbol{\omega}t) = \begin{bmatrix} \cos(\omega_0 t) & 0 & \sin(\omega_0 t) \\ 0 & 1 & 0 \\ -\sin(\omega_0 t) & 0 & \cos(\omega_0 t) \end{bmatrix} \quad (4.28)$$

for the orientation part and

$$\mathbf{x}(\alpha, t) = \mathbf{x}_0 + \mathbf{R}_0 \begin{bmatrix} \frac{\cos(\omega_0 t)\sin(c_0\omega_0(\alpha - L/2))}{c_0\omega_0\cos(c_0\omega_0 L/2)} \\ 0 \\ -\frac{\sin(\omega_0 t)\sin(c_0\omega_0(\alpha - L/2))}{c_0\omega_0\cos(c_0\omega_0 L/2)} \end{bmatrix} \quad (4.29)$$

for the position part.

4.2 Piecewise constant deformation soft arm

The piecewise constant deformation dynamic model discretize the weak form of the dynamic equilibrium equations in (4.7) by using the Soft Geometric Jacobian obtained in the previous chapter in (3.1). This allows the derivation of a dynamic model for soft manipulators which has the identical structure of the dynamic model for rigid manipulators. The dynamic equations take the form of nonlinear ordinary differential equations. This dynamic model paves the way to the development of the same model-based control methods used in classical rigid manipulator theory [60].

4.2.1 Hamiltonian formulation

In order to obtain the discrete dynamic model, we need to insert

- the discretized variation of \mathbf{h} given in (3.68)
- the discrete model of velocity $\boldsymbol{\eta}$ given in (3.60)
- the discrete model of acceleration $\dot{\boldsymbol{\eta}}$ given in (3.61)

into the weak form of the dynamic equilibrium equations of the continuum model (4.7). By using the Soft Geometric Jacobian partition given in (3.69), the weak form of the discretized dynamic equilibrium equations reads

$$\begin{aligned}
& \delta(\boldsymbol{\epsilon}_1)^T \int_0^{L_1} \mathbb{J}_{11}^T \left[\mathbf{M}_1(\mathbb{J}_{11}\dot{\boldsymbol{\epsilon}}_1 + \dot{\mathbb{J}}_{11}\boldsymbol{\epsilon}_1) - \widehat{\mathbb{J}}_{11}\boldsymbol{\epsilon}_1^T \mathbf{M}_1\mathbb{J}_{11}\dot{\boldsymbol{\epsilon}}_1 + \widehat{\boldsymbol{\epsilon}}_1^T \mathbf{K}_1\boldsymbol{\epsilon}_1 - \mathbf{g}_{ext,1} \right] d\alpha_1 \dots \\
& \dots + \delta(\boldsymbol{\epsilon}_1)^T \int_{L_1}^{L_2} \mathbb{J}_{21}^T \left[\mathbf{M}_2(\mathbb{J}_{21}\dot{\boldsymbol{\epsilon}}_1 + \dot{\mathbb{J}}_{21}\boldsymbol{\epsilon}_1) - \widehat{\mathbb{J}}_{21}\boldsymbol{\epsilon}_1^T \mathbf{M}_2\mathbb{J}_{21}\dot{\boldsymbol{\epsilon}}_1 + \widehat{\boldsymbol{\epsilon}}_1^T \mathbf{K}_2\boldsymbol{\epsilon}_1 - \mathbf{g}_{ext,1} \right] d\alpha_2 \dots \\
& \dots + \delta(\boldsymbol{\epsilon}_2)^T \int_{L_1}^{L_2} \mathbb{J}_{22}^T \left[\mathbf{M}_2(\mathbb{J}_{22}\dot{\boldsymbol{\epsilon}}_2 + \dot{\mathbb{J}}_{22}\boldsymbol{\epsilon}_2) - \widehat{\mathbb{J}}_{22}\boldsymbol{\epsilon}_2^T \mathbf{M}_2\mathbb{J}_{22}\dot{\boldsymbol{\epsilon}}_2 + \widehat{\boldsymbol{\epsilon}}_2^T \mathbf{K}_2\boldsymbol{\epsilon}_2 - \mathbf{g}_{ext,2} \right] d\alpha_2 \dots \\
& \dots + \dots + \dots \\
& \dots + \delta(\boldsymbol{\epsilon}_1)^T \int_{L_{n-1}}^{L_n} \mathbb{J}_{n1}^T \left[\mathbf{M}_n(\mathbb{J}_{n1}\dot{\boldsymbol{\epsilon}}_1 + \dot{\mathbb{J}}_{n1}\boldsymbol{\epsilon}_1) - \widehat{\mathbb{J}}_{n1}\boldsymbol{\epsilon}_1^T \mathbf{M}_n\mathbb{J}_{n1}\dot{\boldsymbol{\epsilon}}_1 + \widehat{\boldsymbol{\epsilon}}_1^T \mathbf{K}_n\boldsymbol{\epsilon}_1 - \mathbf{g}_{ext,1} \right] d\alpha_n \dots \\
& \dots + \dots + \delta(\boldsymbol{\epsilon}_n)^T \int_{L_{n-1}}^{L_n} \mathbb{J}_{nn}^T \left[\mathbf{M}_n(\mathbb{J}_{nn}\dot{\boldsymbol{\epsilon}}_n + \dot{\mathbb{J}}_{nn}\boldsymbol{\epsilon}_n) - \widehat{\mathbb{J}}_{nn}\boldsymbol{\epsilon}_n^T \mathbf{M}_n\mathbb{J}_{nn}\dot{\boldsymbol{\epsilon}}_n + \widehat{\boldsymbol{\epsilon}}_n^T \mathbf{K}_n\boldsymbol{\epsilon}_n - \mathbf{g}_{ext,n} \right] d\alpha_n = 0
\end{aligned} \tag{4.30}$$

where we indicated with \mathbf{M}_j and \mathbf{K}_j the 6×6 mass matrix and the 6×6 stiffness matrix of the element of length l_j . This means that for each element, we have the possibility to select the cross-section properties. This is particularly important in manipulators whose cross-section, and thus geometric properties, change along the reference curve. By introducing the $6n \times 6n$ diagonal matrices $\mathbf{M}_N = \text{diag}(\mathbf{M}_j)$ and $\mathbf{K}_N = \text{diag}(\mathbf{K}_j)$, $j = 1, \dots, n$, and recalling to the expression of the Soft Geometric Jacobian given in Table 3.1, the weak form of the discretized dynamic equilibrium equations in (4.30) takes the compact form

$$\delta(\boldsymbol{\epsilon})^T \int_L \mathbb{J}^T \left[\mathbf{M}_N(\mathbb{J}\ddot{\boldsymbol{\epsilon}} + \dot{\mathbb{J}}\dot{\boldsymbol{\epsilon}}) - \widehat{\mathbb{J}}\dot{\boldsymbol{\epsilon}}^T \mathbf{M}_N\mathbb{J}\dot{\boldsymbol{\epsilon}} + \widehat{\boldsymbol{\epsilon}}^T \mathbf{K}_N\boldsymbol{\epsilon} - \mathbf{g}_{ext} \right] d\alpha = 0 \quad (4.31)$$

Since (4.31) holds $\forall \delta(\boldsymbol{\epsilon}) \neq 0$ and, we can rewrite this Equation as

$$\begin{aligned} & \left[\int_L \mathbb{J}^T \mathbf{M}_N \mathbb{J} d\alpha \right] \ddot{\boldsymbol{\epsilon}} + \left[\int_L \mathbb{J}^T \mathbf{M}_N \dot{\mathbb{J}} d\alpha \right] \dot{\boldsymbol{\epsilon}} - \left[\int_L \mathbb{J}^T \widehat{\mathbb{J}}\dot{\boldsymbol{\epsilon}}^T \mathbf{M}_N \mathbb{J} d\alpha \right] \dot{\boldsymbol{\epsilon}} \dots \\ & - \left[\int_L \mathbb{J}^T \widehat{\boldsymbol{\epsilon}}^T \mathbf{K}_N d\alpha \right] \boldsymbol{\epsilon} - \left[\int_L \mathbb{J}^T \mathbf{g}_{ext} d\alpha \right] = 0 \end{aligned} \quad (4.32)$$

Naming the coefficient matrices in the squared parenthesis of (4.32), we obtain the piecewise constant deformation dynamic equations as

$$\mathbb{M}(\alpha, \boldsymbol{\epsilon})\ddot{\boldsymbol{\epsilon}} + (\mathbb{C}_1(\alpha, \boldsymbol{\epsilon}, \dot{\boldsymbol{\epsilon}}) - \mathbb{C}_2(\alpha, \boldsymbol{\epsilon}, \dot{\boldsymbol{\epsilon}}))\dot{\boldsymbol{\epsilon}} - \mathbb{K}\boldsymbol{\epsilon} = \mathbb{F} \quad (4.33)$$

where we can recognize the classical structure of the Lagrangian model for rigid serial manipulators.

The coefficient matrices in the dynamic model (4.33) are:

- $\mathbb{M} = \int_L \mathbb{J}^T \mathbf{M}_N \mathbb{J} d\alpha$, the $6n \times 6n$ discretized mass matrix. The dependency on $\boldsymbol{\epsilon}$ of this matrix has an actual physical meaning. Indeed, if the soft arm is initially curved and/or twisted, its inertia, which is related to the shape of the arm, is affected and this is taken into account by \mathbf{d}_j^0 (see, e.g., Eq. 2.27).
- $\mathbb{C}_1 = \int_L \mathbb{J}^T \mathbf{M}_N \dot{\mathbb{J}} d\alpha$, the $6n \times 6n$ velocity matrix which contributes only if $\dot{\boldsymbol{\epsilon}}$ does not vanish, i.e., only when the deformation of the arm changes in time.
- $\mathbb{C}_2 = \int_L \mathbb{J}^T \widehat{\mathbb{J}}\dot{\boldsymbol{\epsilon}}^T \mathbf{M}_N \mathbb{J} d\alpha$, the $6n \times 6n$ velocity matrix related to gyroscopic effects, contributes also in the case of a rigid body motion of the soft arm.
- $\mathbb{K} = \int_L \mathbb{J}^T \widehat{\boldsymbol{\epsilon}}^T \mathbf{K}_N d\alpha$, the $6n \times 6n$ discretized stiffness matrix.
- $\mathbb{F} = \int_L \mathbb{J}^T \mathbf{g}_{ext} d\alpha$, the $6n \times 1$ vector of generalized applied forces. It also includes actuation loads and gravity field.

4.2.2 Special case: constant deformation soft arm

The dynamic equations for a constant deformation soft arm are obtained in the same fashion of the piecewise constant deformation model. We obtain the same structure, with matrices of dimensions 6×6 .

Hamiltonian formulation

Using (3.78), (3.80) and (3.83), the weak form of the discretized dynamic equilibrium equations in (4.7) for a constant deformation soft arm becomes

$$\delta(\boldsymbol{\epsilon})^T \int_L \mathbf{J}^T (\mathbf{M}(\mathbf{J}\ddot{\boldsymbol{\epsilon}} + \dot{\mathbf{J}}\dot{\boldsymbol{\epsilon}}) - \widehat{\mathbf{J}}\dot{\boldsymbol{\epsilon}}^T \mathbf{M}\mathbf{J}\dot{\boldsymbol{\epsilon}} + \widehat{\boldsymbol{\epsilon}}^T \mathbf{K}\boldsymbol{\epsilon} - \mathbf{g}_{ext}) d\alpha = 0 \quad (4.34)$$

Since (4.34) holds $\forall \delta(\boldsymbol{\epsilon}) \neq 0$ and, we can rewrite this Equation as

$$\begin{aligned} & \left[\int_L \mathbf{J}^T \mathbf{M}\mathbf{J} d\alpha \right] \ddot{\boldsymbol{\epsilon}} + \left[\int_L \mathbf{J}^T \mathbf{M}\dot{\mathbf{J}} d\alpha \right] \dot{\boldsymbol{\epsilon}} - \left[\int_L \mathbf{J}^T \widehat{\mathbf{J}}\dot{\boldsymbol{\epsilon}}^T \mathbf{M}\mathbf{J} d\alpha \right] \dot{\boldsymbol{\epsilon}} \dots (4.35) \\ & - \left[\int_L \mathbf{J}^T \widehat{\boldsymbol{\epsilon}}^T \mathbf{K} d\alpha \right] \boldsymbol{\epsilon} - \left[\int_L \mathbf{J}^T \mathbf{g}_{ext} d\alpha \right] = 0 \end{aligned}$$

Naming the coefficient matrices in the squared parenthesis of (4.35), we obtain the constant deformations dynamic equations as

$$\mathbb{M}(\alpha, \boldsymbol{\epsilon})\ddot{\boldsymbol{\epsilon}} + (\mathbb{C}_1(\alpha, \boldsymbol{\epsilon}, \dot{\boldsymbol{\epsilon}}) - \mathbb{C}_2(\alpha, \boldsymbol{\epsilon}, \dot{\boldsymbol{\epsilon}}))\dot{\boldsymbol{\epsilon}} - \mathbb{K}\boldsymbol{\epsilon} = \mathbb{F} \quad (4.36)$$

where we recognize the structure of the Lagrangian model of rigid serial manipulators. The coefficient matrices in the dynamic model (4.36) are:

- $\mathbb{M} = \int_L \mathbf{J}^T \mathbf{M}\mathbf{J} d\alpha$, the 6×6 discretized mass matrix.
- $\mathbb{C}_1 = \int_L \mathbf{J}^T \mathbf{M}\dot{\mathbf{J}} d\alpha$, the 6×6 velocity matrix which contributes only if $\dot{\boldsymbol{\epsilon}}$ does not vanish, i.e., only when the deformation of the arm changes in time.
- $\mathbb{C}_2 = \int_L \mathbf{J}^T \widehat{\mathbf{J}}\dot{\boldsymbol{\epsilon}}^T \mathbf{M}\mathbf{J} d\alpha$, the 6×6 velocity matrix related to gyroscopic effects, contributes also in the case of a rigid body motion of the soft arm.
- $\mathbb{K} = \int_L \mathbf{J}^T \widehat{\boldsymbol{\epsilon}}^T \mathbf{K} d\alpha$, the 6×6 discretized stiffness matrix.
- $\mathbb{F} = \int_L \mathbf{J}^T \mathbf{g}_{ext} d\alpha$, the 6×1 vector of generalized applied forces. It also includes actuation loads and gravity field.

4.2.3 Example

Let us consider a soft arm rotating at a constant velocity w in its own plane. The analytic solution for this problem is given in Sec. 4.1.4. Here, for we simulate the motion of the manipulator using the discretized model with the following physical parameters:

- $L = 1$ m
- $E = 69 \times 10^9$ Pa
- $\rho = 2.7 \times 10^3$ Kg/m³

Figure 4.2 illustrates the tip displacements of the soft arm during 1 s. The system is subject to three different velocities w in its own plane. The solid lines indicate the analytic solutions, while the numerical solutions are indicated by the dotted lines. Also in this case, we appreciate that the current nonlinear finite element formalism matches the exact solution.

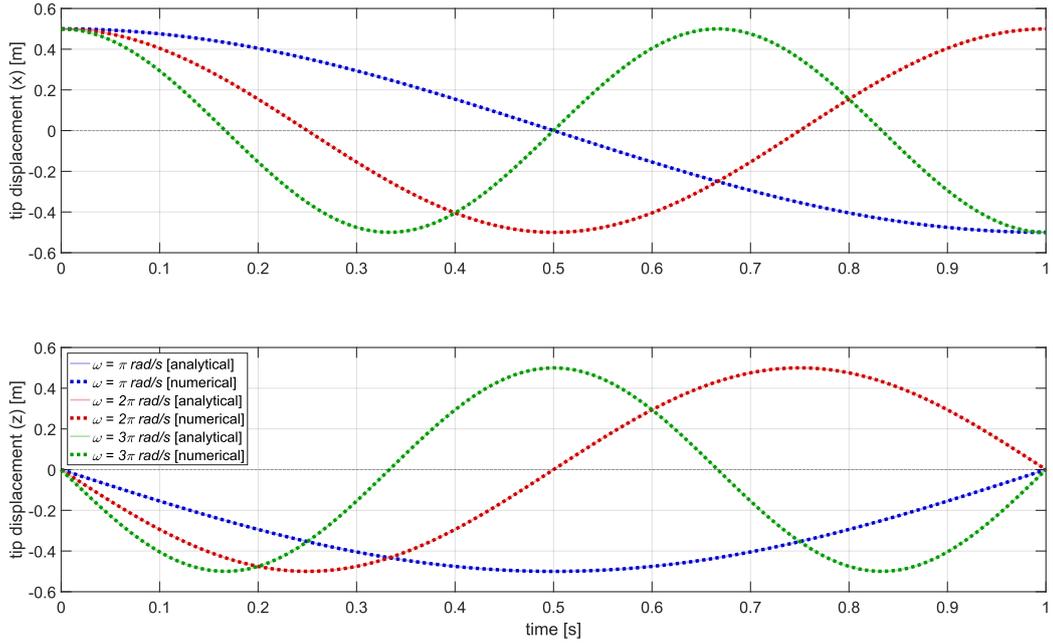


Figure 4.2: Tip displacements in x and z –directions for the free soft arm rotating at three different velocities w .

4.3 Summary

- The *dynamic model* of a piecewise constant deformation soft arm reads

$$\mathbf{M}(\alpha, \epsilon)\ddot{\epsilon} + (\mathbf{C}_1(\alpha, \epsilon, \dot{\epsilon}) - \mathbf{C}_2(\alpha, \epsilon, \dot{\epsilon}))\dot{\epsilon} - \mathbf{K}\epsilon = \mathbf{F} \quad (4.37)$$

Chapter 5

Constrained dynamics

In this thesis, with the term *constrained dynamics* we indicate the dynamics of articulated (or multibody) robotic mechanisms, constituted either by rigid or soft arms, and connected either by rigid or elastic joints. For the rigid arms, we refer to the equations of motion for a rigid body obtained in Chapter 1. For the soft arms, we refer to the equations of motion for the CD and PWCD models obtained in Chapter 4.

The mechanical coupling of the different arms is achieved by defining the concept of *relative transformations*. From this concept, we derive the kinematics equations for the rigid and the elastic joints. Finally, we derive the equations of motion for a generic robotic manipulator, which take the form of differential-algebraic equations formulated on a Lie group. Since the topology of the mechanical system is implicitly contained in the finite element mesh, the dynamics of trees, stars and loop-structured robotic mechanisms is naturally included in the equations of motion derived in this Chapter.

5.1 The coupling approach

The joints of a robotic mechanism restricts the relative motions between its arms. A convenient formulation can be achieved in the context of the Special Euclidean group $SE(3)$ formalism, thanks to the *relative transformation* description and the concept of restricted relative motion using the notion of *Lie subgroups*. This approach has been recently proposed in the contexts of rigid multibody systems [48] and flexible manipulators [61]. The method introduces, for each joint of the mechanism, six algebraic constraints, to which six Lagrange multipliers are associated. In this description, the relative variables of the joints appear as the unknown variables in the equations of motion. Accordingly, the resulting constrained dynamic formulation involves a mixed-variable formulation, where the *absolute variables* of the rigid and soft arms are related with the *relative variables* of the rigid and flexible joints through algebraic constraint equations.

The advantages for modelling joints via algebraic constraints are listed as follows

- the method allows the definition of all the low-pair joints;

- the method allows the simulation of passive joints;
- the method allows the straightaway inclusion of rigid and flexible joints.

5.2 Relative transformations

The *relative transformations* introduced in this Section play for the joints the same role that the *rigid body transformations* discussed in Section 1.1 play for the arms.

Since we associate frames with the nodes constituting both the rigid and the soft arms, to describe the relative motion between frames we need to define a concept of relative transformations of frames. The objective is to find a suitable representation of the relative transformation which is not affected by the overall motion of the two nodes, but that is sensitive *only* to their relative motion. To that purpose, we use again the special Euclidean group $SE(3)$ framework.

Let us consider two mappings $\mathbf{H}^a(t)$ and $\mathbf{H}^b(t)$, with $t \in \mathbb{R}$, which describe the trajectory of two nodes associated to the two frames. The two nodes exhibit a relative motion which can be conveniently expressed using the following proposition.

Proposition 2 *The relative transformation between two mappings a and b can be represented by a right multiplication on $SE(3)$, i.e.*

$$\mathbf{H}^b(t) = \mathbf{H}^a(t)\mathbf{H}^r(t) \quad (5.1)$$

where $\mathbf{H}^r(t)$ represents the relative transformation expressed with respect to the frame $\mathbf{H}^a(t)$.

Notice that Eq. 5.1 can be inverted as

$$\mathbf{H}^a(t) = \mathbf{H}^b(t)(\mathbf{H}^r(t))^{-1} \quad (5.2)$$

such that $(\mathbf{H}^r(t))^{-1}$ takes the meaning of the relative transformation between a and b expressed with respect to b .

In order to demonstrate that $\mathbf{H}^r(t)$ is the representation of the relative transformation that we are looking for, i.e., it is not affected by the overall motion of the two nodes, let us consider a superimposed Euclidean transformation \mathbf{H}^e to $\mathbf{H}^a(t)$ and $\mathbf{H}^b(t)$, which leads to $\mathbf{H}^{a'}(t)$ and $\mathbf{H}^{b'}(t)$ as

$$\mathbf{H}^{a'}(t) = \mathbf{H}^e\mathbf{H}^a(t) \quad (5.3)$$

$$\mathbf{H}^{b'}(t) = \mathbf{H}^e\mathbf{H}^b(t) \quad (5.4)$$

Inserting Eq. 5.1 in (5.4), we have

$$\mathbf{H}^{b'}(t) = \mathbf{H}^e\mathbf{H}^a(t)\mathbf{H}^r(t) \quad (5.5)$$

And, inserting Eq. 5.3 in (5.5), we obtain

$$\mathbf{H}^{b'}(t) = \mathbf{H}^{a'}(t)\mathbf{H}^r(t) \quad (5.6)$$

which means that the relative transformation between a' and b' is always given by $\mathbf{H}^r(t)$. Thus, this matrix is a measure of the relative transformation between the two mappings a and b which is *invariant under a superimposed Euclidean transformation*. This means that Eq. 5.1 is a convenient way to express the relative transformations between two mappings, since, in $\mathbf{H}^r(t)$, the overall large amplitude motion contributions contained in the mappings are filtered out and *only* the relative motion information is captured.

5.3 Rigid joint

A rigid joint is an element which implements restricted relative motion between two arms.

The representation of the relative transformation of mappings has been described in the previous section, while the restricted relative motions between the two nodes can be described using the concept of subgroups, and the respective sub-algebra (see, e.g., Appendix B).

Let us consider two generic arms which are connected by a kinematic joint, as illustrated in Fig. 5.1. The body-attached frames of the two nodes of the arms are denoted as $\mathbf{H}_A = \mathcal{H}(\mathbf{R}_A, \mathbf{u}_A)$ and $\mathbf{H}_B = \mathcal{H}(\mathbf{R}_B, \mathbf{u}_B)$. According to (5.1), the relative transformation allowed by the joint between the two nodes A and B is

$$\mathbf{H}_B = \mathbf{H}_A \mathbf{H}_{J,I} \quad (5.7)$$

where $\mathbf{H}_{J,I}$ is a frame element which represents the relative transformation describing the behavior of the joint I . Such frames have a restricted number of degrees of freedom $k_I \leq 6$ and belong to a subgroup of $SE(3)$.

According to (C.4), their derivatives takes the generic form $d_a(\mathbf{H}_{J,I}) = \mathbf{H}_{J,I} \widetilde{\mathbf{a}}_{J,I} = \mathbf{H}_{J,I}(\widetilde{\mathbf{A}}_I \mathbf{a}_{j,I})$. In particular, the variation and the time derivative of $\mathbf{H}_{J,I}$ can be expressed as

$$\delta(\mathbf{H}_{J,I}) = \mathbf{H}_{J,I} \widetilde{\delta \mathbf{h}}_{J,I} = \mathbf{H}_{J,I}(\widetilde{\mathbf{A}}_I \delta \mathbf{h}_{j,I}) \quad (5.8)$$

$$\dot{\mathbf{H}}_{J,I} = \mathbf{H}_{J,I} \widetilde{\boldsymbol{\eta}}_{J,I} = \mathbf{H}_{J,I}(\widetilde{\mathbf{A}}_I \boldsymbol{\eta}_{j,I}) \quad (5.9)$$

where $\boldsymbol{\eta}_{j,I}$ and $\delta \mathbf{h}_{j,I}$ are six-dimensional vectors of velocities and infinitesimal arbitrary motions, $\boldsymbol{\eta}_{j,I}$ and $\delta \mathbf{h}_{j,I}$ are k_I -dimensional vectors of velocities and infinitesimal arbitrary motions of the relative degrees of freedom, and \mathbf{A}_I is the $6 \times k_I$ full rank matrix which spans the subspace of allowed relative motions (see, e.g., Table 5.1).

Equation 5.7 is included in the equations of motion as the constraint equations which are computed in the next section.

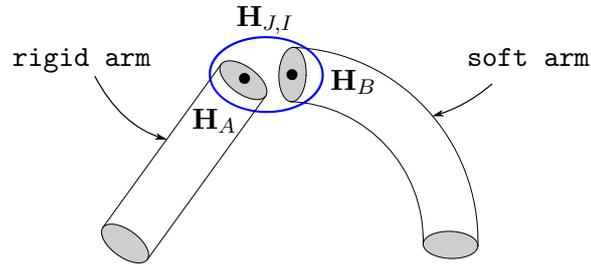


Figure 5.1: A kinematic joint which connects two generic arms of a robotic manipulator.

		Subgroup	Dimension of \mathfrak{h}	\mathbf{A}
Revolute		$SO(2)$	1	$\begin{bmatrix} \mathbf{0}_{3 \times 1} \\ \mathbf{n} \end{bmatrix}$
Prismatic		\mathbb{R}	1	$\begin{bmatrix} \mathbf{n} \\ \mathbf{0}_{3 \times 1} \end{bmatrix}$
Screw (pitch p)		H_p	1	$\begin{bmatrix} p\mathbf{n} \\ \mathbf{n} \end{bmatrix}$
Cylindrical		$SO(2) \times \mathbb{R}$	2	$\begin{bmatrix} \mathbf{0}_{3 \times 1} & \mathbf{n} \\ \mathbf{n} & \mathbf{0}_{3 \times 1} \end{bmatrix}$
Planar		\mathbb{R}^2	2	$\begin{bmatrix} \mathbf{n}_1 & \mathbf{n}_2 \\ \mathbf{0}_{3 \times 1} & \mathbf{0}_{3 \times 1} \end{bmatrix}$
Spherical		$SO(3)$	3	$\begin{bmatrix} \mathbf{0}_{3 \times 1} & \mathbf{0}_{3 \times 1} & \mathbf{0}_{3 \times 1} \\ \mathbf{n}_1 & \mathbf{n}_2 & \mathbf{n}_3 \end{bmatrix}$

Table 5.1: Joint definition.

5.3.1 Kinematic constraints

In this section we derive a suitable representation for the kinematic constraints. Since the constraints will be in the form $\boldsymbol{\varphi}(\mathbf{H}_\varphi) = \mathbf{0}$, they belong to the class of *holonomic* (or, *integrable*) constraints.

Constraint equations

Let us premultiply Eq. 5.7 with \mathbf{H}_B^{-1} such that we obtain

$$\mathbf{I}_{4 \times 4} = \mathbf{H}_B^{-1} \mathbf{H}_A \mathbf{H}_{J,I} \quad (5.10)$$

By introducing the residual matrix $\mathbf{H}_\varphi \in SE(3)$ defined as $\mathbf{H}_\varphi = \mathbf{H}_B^{-1} \mathbf{H}_A \mathbf{H}_{J,I}$, we can reformulate Eq. 5.7 as the following constraint equation

$$\mathbf{H}_\varphi(\mathbf{H}_A, \mathbf{H}_B, \mathbf{H}_{J,I}) = \mathbf{I}_{4 \times 4} \quad (5.11)$$

In order to include the kinematic joints in the dynamic formulation of the robotic manipulator using the Lagrange multiplier method in the Hamilton's principle, we need to define a constraint equation vector $\boldsymbol{\varphi}$. Since Eq. 5.11 is a matrix equation, we have to find a way to produce a six-dimensional vector from a $SE(3)$ element. One possible way is to use the vectorial map such that the constraint equations (5.11) for a kinematic joint take the form of the six-dimensional constraint vector

$$\boldsymbol{\varphi}(\mathbf{H}_\varphi) = \text{vect}_{SE(3)}(\mathbf{H}_\varphi) = \mathbf{0}_{6 \times 1} \quad (5.12)$$

where the vectorial map is the linear operator which can be seen as a linearized version of the logarithmic map and it is defined as

$$\text{vect}_{SE(3)}(\mathcal{H}(\mathbf{R}, \mathbf{u})) = \begin{bmatrix} \mathbf{u} \\ \boldsymbol{\varphi}(\mathbf{R}) \end{bmatrix} \quad (5.13)$$

such that $\tilde{\boldsymbol{\varphi}}(\mathbf{R}) = (\mathbf{R} - \mathbf{R}^T)/2$. The vectorial map introduces systematically six constraints for a kinematic joint. The vectorial map imposes that the relative displacements and the relative orientations contained in $\mathbf{H}_{J,I}$ are exactly the relative configuration between node A and B , by imposing three constraints at the position level and three constraints at the rotation level.

This formulation is a mixed-variable formulation, since it leads systematically to six constraint equations which involve the nodal variables \mathbf{H}_A and \mathbf{H}_B as well as the relative variable $\mathbf{H}_{J,I}$. Indeed, for the joint I with k_I degrees of freedom, the vectorial map introduces six equations which are linear combinations of the $6 - k_I$ equations preventing the non-allowed relative motion and the k_I equations describing the evolution of the allowed relative motions. For example, in a revolute joint, five equations deal with the non-allowed relative motions and one equation describes the evolution of the relative angle about the joint axis.

Constraint gradient

In this section we derive the constraint gradient, which will be used in the Hamilton's principle for formulating the constrained dynamics of rigid and soft robotic manipulators.

The constraint gradient φ_q is defined from the directional derivative of the constraints (5.12) as

$$\delta(\varphi) = D\varphi \cdot \widetilde{\mathbf{A}}\delta\mathbf{h} = \varphi_q \mathbf{A}\delta\mathbf{h} \quad (5.14)$$

where

$$\mathbf{A} = \text{diag}(\mathbf{I}_{6 \times 6}, \mathbf{I}_{6 \times 6}, \mathbf{A}_I) \quad (5.15)$$

and

$$\delta\mathbf{h} = \begin{bmatrix} \delta\mathbf{h}_A \\ \delta\mathbf{h}_B \\ \delta\mathbf{h}_{j,I} \end{bmatrix} \quad (5.16)$$

The variation of the constraint equations (5.12) leads to

$$\delta(\varphi) = \delta(\text{vect}_{SE(3)}(\mathbf{H}_\varphi)) = \text{vect}_{SE(3)}(\delta(\mathbf{H}_\varphi)) \quad (5.17)$$

where in the second equality we used the fact that the vectorial map is a linear operator.

The variation of \mathbf{H}_φ about the equilibrium $\mathbf{H}_\varphi = \mathbf{I}_{4 \times 4}$ is obtained as

$$\delta(\mathbf{H}_\varphi) = -\widetilde{\delta\mathbf{h}_B} \mathbf{H}_B^{-1} \mathbf{H}_A \mathbf{H}_{J,I} + \mathbf{H}_B^{-1} \mathbf{H}_A \widetilde{\delta\mathbf{h}_A} \mathbf{H}_{J,I} + \mathbf{H}_B^{-1} \mathbf{H}_A \mathbf{H}_{J,I} \widetilde{\delta\mathbf{h}_{J,I}} \quad (5.18)$$

$$\Leftrightarrow \delta\mathbf{h}_\varphi = -\delta\mathbf{h}_B + \text{Ad}_{\mathbf{H}_{J,I}^{-1}}(\delta\mathbf{h}_A) + \delta\mathbf{h}_{J,I} \quad (5.19)$$

where Ad is the adjoint representation defined in (A.4). By inserting (5.18)–(5.19) into Eq. 5.17, we obtain the constraint gradient evaluated at the equilibrium,

$$\varphi_q \mathbf{A} = \begin{bmatrix} \text{Ad}_{\mathbf{H}_{J,I}^{-1}} & -\mathbf{I}_{6 \times 6} & \mathbf{A}_I \end{bmatrix} \quad (5.20)$$

which only depends on the relative motion, and not on the overall motion of the nodes A and B . This means that the non-linearity of the formulation is only caused by the local motions.

Constant offset

Usually, the body attached frames of a soft arm are placed at the extremes of each constant deformation element, while the body-attached frame of a rigid body is usually placed at its center of mass. Since the kinematic joint connects the extremes of each arm, in the case of rigid bodies we might need to define an additional node on the body at its extreme and introduce a rigid constraint between this node and the node at the center of mass of the rigid body. Another possible solution is to account for this discrepancy by describing the offset between

the body-attached frame and the real point by which the arm is connected. In the $SE(3)$ framework, this strategy is appealing since it involves a constant offset. Let us consider the frame at a material point A' of a rigid arm whose body-attached frame variable is attached in A , its center of mass. We can express the frame at A' as

$$\mathbf{H}_{A'} = \mathbf{H}_A \mathbf{H}_{AA'} \quad (5.21)$$

where \mathbf{H}_A is the frame at A and $\mathbf{H}_{AA'}$ is a constant relative transformation. By applying the same transformation to the node B on the other arm, Eq. 5.11 becomes

$$\mathbf{H}_\varphi = \mathbf{H}_{BB'}^{-1} \mathbf{H}_B^{-1} \mathbf{H}_A \mathbf{H}_{AA'} \mathbf{H}_{J,I} = \mathbf{I}_{4 \times 4} \quad (5.22)$$

The variation of $\mathbf{H}_{A'}$ leads to

$$\delta(\mathbf{H}_{A'}) = \delta(\mathbf{H}_A) \mathbf{H}_{AA'} \quad (5.23)$$

$$\mathbf{H}_{A'} \widetilde{\delta \mathbf{h}_{A'}} = \mathbf{H}_A \widetilde{\delta \mathbf{h}_A} \mathbf{H}_{AA'} \quad (5.24)$$

such that we can express the variation of A' in terms of the body-attached frame at A as

$$\delta \mathbf{h}_{A'} = \text{Ad}_{\mathbf{H}_{AA'}^{-1}} (\delta \mathbf{h}_A) \quad (5.25)$$

Hence, the constraint gradient in (5.26) becomes

$$\varphi_q \mathbf{A} = \left[\text{Ad}_{\mathbf{H}_{J,I}^{-1} \mathbf{H}_{AA'}^{-1}} \quad - \text{Ad}_{\mathbf{H}_{BB'}^{-1}} \quad \mathbf{A}_I \right] \quad (5.26)$$

and it still depends only on the relative motion.

5.4 Flexible joint

Flexible joints are joints undergoing deflections. The presence of joint flexibility is common in current manipulators, where compliant motion transmission/reduction elements are used. Deflection of these flexible elements can be modeled as being concentrated at the joints of the robot, thus limiting the complexity of the associated equations of motion. In this work, we assume that joint deflections are small, so that flexibility effects are limited to the domain of linear elasticity; this hypothesis well suits for robotics applications. From now on, flexible-joints robots will be indicated as robots with *elastic* joints. The elasticity at joint I is modeled by a spring with a certain stiffness. Moreover, we can model friction effects inside the elastic joints by adding a damper. In this case, we refer to *damped-elastic* joints. Figure 5.2 illustrates a revolute elastic joint between two generic arms. In order to obtain a mathematical description for the contribution to the internal forces due to joint deflection, let us represent the relative motion in terms of relative coordinates, namely relative angles and relative displacements.

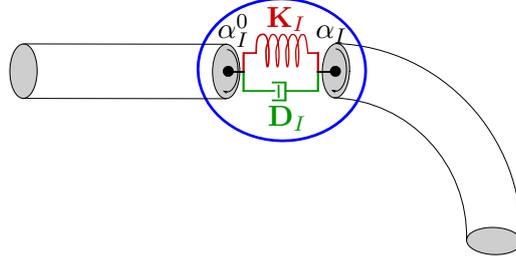


Figure 5.2: Schematic representation of a revolute elastic joint.

5.4.1 Joint deflection

The relative joint motion informations are contained in the matrix $\mathbf{H}_{j,I}$. Here, starting from this matrix, we want to obtain the representation of the relative motion in terms of relative angles and relative displacements.

The commutativity of the derivatives in (C.6) leads to

$$\mathbf{A}_I \delta(\boldsymbol{\eta}_{j,I}) = \mathbf{A}_I (\delta \dot{\mathbf{h}}_{j,I}) + (\widehat{\mathbf{A}_I \boldsymbol{\eta}_{j,I}}) \mathbf{A}_I \delta(\mathbf{h}_{j,I}) \quad (5.27)$$

For Abelian subgroups, namely all the lower pair joints except planar and spherical joints, the Lie bracket vanishes, i.e. $(\widehat{\mathbf{A}_I \boldsymbol{\eta}_{j,I}}) \mathbf{A}_I = \mathbf{0}$. In this case, we have that $\mathbf{A}_I \delta(\boldsymbol{\eta}_{j,I}) = \mathbf{A}_I (\delta \dot{\mathbf{h}}_{j,I})$, and, since \mathbf{A}_I has full rank, $\delta(\boldsymbol{\eta}_{j,I}) = (\delta \dot{\mathbf{h}}_{j,I})$. Introducing a $k_I \times 1$ vector $\boldsymbol{\alpha}_I$ as $\dot{\boldsymbol{\alpha}}_I = \boldsymbol{\eta}_{j,I}$, we have

$$\delta(\boldsymbol{\alpha}_I) = \delta \mathbf{h}_{j,I} \quad (5.28)$$

where $\boldsymbol{\alpha}_I$ is the relative coordinates of the joint. By introducing the initial coordinate of the joint as $\boldsymbol{\alpha}_I^0$, the deflection $\boldsymbol{\delta}_I$ at the I -th joint reads

$$\boldsymbol{\delta}_I = \boldsymbol{\alpha}_I - \boldsymbol{\alpha}_I^0 \quad (5.29)$$

and it results a scalar value if the joint has one degree of freedom. For joints with more than one degrees of freedom, the deflection $\boldsymbol{\delta}_I$ is a vectorial entity. Since we have that $\delta(\boldsymbol{\alpha}_I) = \delta \mathbf{h}_{j,I}$, the definition of the joint deflection $\boldsymbol{\delta}_I$ can be used for expressing the contribution of the flexible joints to the equations of motion of the robotic manipulator in a straightforward manner.

5.4.2 Internal forces of the joint

The internal forces of an elastic joints include the elastic and the dissipative terms, derived as follows.

Elastic forces

The potential energy of the elastic joint can be introduced by considering the stiffnesses related to the relative coordinates involved in the kinematic joints as

$$\mathcal{V}_{int,K} = \frac{1}{2} \boldsymbol{\delta}_I \mathbf{K}_I \boldsymbol{\delta}_I \quad (5.30)$$

where $\mathbf{K}_I = \text{diag}(K_{I,1}, \dots, K_{I,k_I})$ is the diagonal stiffness matrix of the kinematic joint I . As example, the elasticity of a single degree of freedom elastic joint I is modeled through a springer of stiffness $K_{I,1} > 0$. The contribution of a spring element in the internal forces is simply obtained as

$$\delta(\mathcal{V}_{int,K}) = \delta \mathbf{h}_{j,I}^T \mathbf{g}_{int,I,K}(\boldsymbol{\alpha}_I) \quad (5.31)$$

where the internal forces in the constrained equations of motion are given by

$$\mathbf{g}_{int,I,K}(\boldsymbol{\alpha}_I) = \mathbf{K}_I \boldsymbol{\delta}_I \quad (5.32)$$

Dissipation forces

The framework of relative coordinates allows also modelling dissipation effects inside the kinematic joints. The associated virtual work reads

$$\delta(\mathcal{V}_{int,D}) = \delta \mathbf{h}_{j,I}^T \mathbf{g}_{int,I,D}(\boldsymbol{\alpha}_I) \quad (5.33)$$

The internal forces due to a damper element in the constrained equations of motion are given by

$$\mathbf{g}_{int,I,D}(\boldsymbol{\alpha}_I) = \mathbf{D}_I \dot{\boldsymbol{\alpha}}_I = \mathbf{D}_I \boldsymbol{\eta}_{j,I} \quad (5.34)$$

where we used the fact that $\dot{\boldsymbol{\alpha}}_I = \boldsymbol{\eta}_{j,I}$ per definition and $\mathbf{D}_I = \text{diag}(D_{I,1}, \dots, D_{I,D_I})$ is the diagonal matrix of damping coefficients associated with the kinematic joint I .

5.5 The constrained dynamical system

A generic robotic manipulator is a collection of rigid and soft arms interconnected by rigid and/or flexible joints in a serial or parallel-like structure.

5.5.1 Kinematic configuration

In this thesis, we used the $SE(3)$ framework for obtaining the equations of motion of the arms and the kinematic constraint equations for the joints. As illustrated in Fig. 5.3, the elements of $SE(3)$, namely the frames \mathbf{H} , have two different meanings:

- Body-attached frames. They are referred to as *nodal frames*. For each rigid arm, we have one nodal frame attached to the node corresponding to the center of mass. For each constant deformation element of a soft arm, we have two nodal frames, attached to the initial and final node of the element. For node I , we use the notation \mathbf{H}_I . These frames have six degrees

of freedom, namely three translations and three rotations. According to the definition of derivative given by (A.2), the variation and the time derivative of nodal frames are denoted as

$$\delta(\mathbf{H}_I) = \mathbf{H}_I \widetilde{\delta \mathbf{h}}_I \quad (5.35)$$

$$\dot{\mathbf{H}}_I = \mathbf{H}_I \widetilde{\boldsymbol{\eta}}_I \quad (5.36)$$

- Relative transformations. They are referred to as *relative frames*. These $SE(3)$ elements describe the frame transformation between two nodal frames and are used to formulate the joints. For joint I , we use the notation $\mathbf{H}_{J,I}$. These frames have $k_I \leq 6$ degrees of freedom and belong to a Subgroup of $SE(3)$. According to the definition of derivative given by (C.4), the variation and the time derivative of relative frames are denoted as

$$\delta(\mathbf{H}_{J,I}) = \mathbf{H}_{J,I} \widetilde{\delta \mathbf{h}}_{J,I} = \mathbf{H}_{J,I} (\widetilde{\mathbf{A}}_I \delta \mathbf{h}_{j,I}) \quad (5.37)$$

$$\dot{\mathbf{H}}_{J,I} = \mathbf{H}_{J,I} \widetilde{\boldsymbol{\eta}}_{J,I} = \mathbf{H}_{J,I} (\widetilde{\mathbf{A}}_I \boldsymbol{\eta}_{j,I}) \quad (5.38)$$

Let us consider a generic robotic mechanism with n nodal frames and m joints. The kinematic joints constrain $k = k_1 + \dots + k_m$ degrees of freedom, such that the mechanism has $6n - k_m$ degrees of freedom. Thus, the configuration of the robotic mechanism can be described using the invertible block diagonal matrix

$$\mathbf{H} = \text{diag}(\mathbf{H}_1, \dots, \mathbf{H}_n, \mathbf{H}_{J,1}, \dots, \mathbf{H}_{J,m}) \quad (5.39)$$

which can be interpreted either as an element of a $6n + 6m$ dimensional Lie group or as an element of a $6n + k$ dimensional Lie subgroup. Notice that the variable in \mathbf{H} are not independent. Indeed, the description of nodal and relative frames relies on a redundant set of variables which are related by a set of algebraic constraints. According to (5.35)–(5.38), a general way to represent the variation and the time derivative of \mathbf{H} is

$$\delta(\mathbf{H}) = \mathbf{H} \widetilde{\mathbf{A}} \delta \mathbf{h} \quad (5.40)$$

$$\dot{\mathbf{H}} = \mathbf{H} \widetilde{\mathbf{A}} \boldsymbol{\eta} \quad (5.41)$$

where $\delta \mathbf{h}$ and $\boldsymbol{\eta}$ denote the $(6n + k)$ –dimensional vectors of arbitrary infinitesimal motions and velocities as

$$\delta \mathbf{h} = \begin{bmatrix} \delta \mathbf{h}_1 \\ \dots \\ \delta \mathbf{h}_n \\ \delta \mathbf{h}_{j,1} \\ \dots \\ \delta \mathbf{h}_{j,m} \end{bmatrix}; \quad \boldsymbol{\eta} = \begin{bmatrix} \boldsymbol{\eta}_1 \\ \dots \\ \boldsymbol{\eta}_n \\ \delta \boldsymbol{\eta}_{j,1} \\ \dots \\ \delta \boldsymbol{\eta}_{j,m} \end{bmatrix} \quad (5.42)$$

and \mathbf{A} is the constant $(6n + 6m) \times (6n + k)$ block-diagonal matrix

$$\mathbf{A} = \text{diag}(\mathbf{I}_{6 \times 6}, \dots, \mathbf{I}_{6 \times 6}, \mathbf{A}_1, \dots, \mathbf{A}_m) \quad (5.43)$$

5.5.2 Hamiltonian formulation

According to the novel configuration of the robotic mechanism which includes nodal and relative frames (5.39) and based on the derivatives of the configuration expressed as in (5.40) and (5.41), we can reformulate the Hamilton's principle using the Lagrange multiplier method as

$$\delta \left(\int_{t_0}^{t_1} (\mathcal{K}(\mathbf{H}, \boldsymbol{\eta}) - \mathcal{V}_{int}(\mathbf{H}) + \mathcal{V}_{ext}(\mathbf{H}) - \boldsymbol{\lambda}^T \boldsymbol{\varphi}(\mathbf{H})) dt \right) = 0. \quad (5.44)$$

where $\boldsymbol{\lambda}$ are the Lagrange multipliers associated with the kinematic constraints $\boldsymbol{\varphi}(\mathbf{H}) = \mathbf{0}$, $\mathcal{K}(\mathbf{H}, \boldsymbol{\eta})$ is the usual kinetic energy, while $\mathcal{V}_{int}(\mathbf{H})$ and $\mathcal{V}_{ext}(\mathbf{H})$ are the usual potential energies due to the internal and external forces. The variation of the single terms can be written as

$$\delta \left(\int_{t_0}^{t_1} \mathcal{K}(\mathbf{H}, \boldsymbol{\eta}) dt \right) = - \int_{t_0}^{t_1} \delta \mathbf{h}^T \mathbf{g}_{ine}(\mathbf{H}, \boldsymbol{\eta}, \dot{\boldsymbol{\eta}}) dt \quad (5.45)$$

$$\delta \left(\int_{t_0}^{t_1} \mathcal{V}_{int}(\mathbf{H}) dt \right) = \int_{t_0}^{t_1} \delta \mathbf{h}^T \mathbf{g}_{int}(\mathbf{H}) dt \quad (5.46)$$

$$\delta \left(\int_{t_0}^{t_1} \mathcal{V}_{ext}(\mathbf{H}) dt \right) = - \int_{t_0}^{t_1} \delta \mathbf{h}^T \mathbf{g}_{ext}(\mathbf{H}) dt \quad (5.47)$$

$$\delta \left(\int_{t_0}^{t_1} \boldsymbol{\lambda}^T \boldsymbol{\varphi}(\mathbf{H}) dt \right) = \int_{t_0}^{t_1} (\delta \mathbf{h}^T \mathbf{A}^T \boldsymbol{\varphi}_q^T(\mathbf{H}) \boldsymbol{\lambda} + \delta \boldsymbol{\lambda}^T \boldsymbol{\varphi}(\mathbf{H})) dt \quad (5.48)$$

where \mathbf{g}_{ine} are the inertia forces, \mathbf{g}_{int} are the internal forces, \mathbf{g}_{ext} are the external forces and $\boldsymbol{\varphi}_q$ is the constraint gradient.

5.5.3 Equations of motion

Owing to the arbitrariness of $\delta \mathbf{h}$ and $\delta \boldsymbol{\lambda}$, the equations of motion of a generic robotic mechanism take the form of differential-algebraic equations on a Lie group

$$\dot{\mathbf{H}} = \mathbf{H} \widetilde{\mathbf{A}} \boldsymbol{\eta} \quad (5.49)$$

$$\mathbf{g}_{ine}(\mathbf{H}, \boldsymbol{\eta}, \dot{\boldsymbol{\eta}}) + \mathbf{g}_{int}(\mathbf{H}) + \mathbf{A}^T \boldsymbol{\varphi}_q^T(\mathbf{H}) \boldsymbol{\lambda} - \mathbf{g}_{ext}(\mathbf{H}) = \mathbf{0}_{(6n+k) \times 1} \quad (5.50)$$

$$\boldsymbol{\varphi}(\mathbf{H}) = \mathbf{0}_{6m \times 1} \quad (5.51)$$

where Eq. 5.49 are the kinematic equations, Eq. 5.50 are the dynamic equations, and Eq. 5.51 are the kinematic constraint equations.

5.5.4 Time integration

The implicit generalized- α -scheme explained in Appendix D applied to a general constrained dynamical system leads to the discretized equations of motion

$$\mathbf{H}_{n+1} = \mathbf{H}_n \exp_{SE(3)}(\widetilde{\mathbf{A}} \boldsymbol{\eta}_{n+1}) \quad (5.52)$$

$$\mathbf{g}(\mathbf{H}_{n+1}, \boldsymbol{\eta}_{n+1}, \dot{\boldsymbol{\eta}}_{n+1}) + \mathbf{A}^T \boldsymbol{\varphi}_q^T(\mathbf{H}_{n+1}) \boldsymbol{\lambda}_{n+1} = \mathbf{0}_{(6M+k) \times 1} \quad (5.53)$$

$$\boldsymbol{\varphi}(\mathbf{H}_{n+1}) = \mathbf{0}_{6m \times 1} \quad (5.54)$$

where \mathbf{n}_{n+1} is a $(6M + k)$ -dimensional vector of incremental motions. The time integration formulae are readily identical to (D.7)–(D.9). Equation 5.52 is a formal expression which means that for each node and each joint, there is an expression as

$$\mathbf{H}_{I,n+1} = \mathbf{H}_{I,n} \exp_{SE(3)}(\tilde{\mathbf{n}}_{I,n+1}) \quad (5.55)$$

$$\mathbf{H}_{J,I,n+1} = \mathbf{H}_{J,I,n} \exp_{SE(3)}(\widetilde{\mathbf{A}}_I \mathbf{n}_{j,I,n+1}) \quad (5.56)$$

The tangent operator involved in Eq. D.13 is given by

$$\mathbf{T}(\mathbf{n}_{n+1}) = \text{diag}(\mathbf{T}_{SE(3)}(\mathbf{n}_{1,n+1}), \dots, \mathbf{T}_{SE(3)}(\mathbf{n}_{M,n+1}), \quad (5.57)$$

$$\mathbf{T}_{SE(3)}(\mathbf{A}_1 \mathbf{n}_{J,1,n+1}), \dots, T_{SE(3)}(\mathbf{A}_m \mathbf{n}_{J,m,n+1}))$$

and the iteration matrix is readily identical to Eq. D.22.

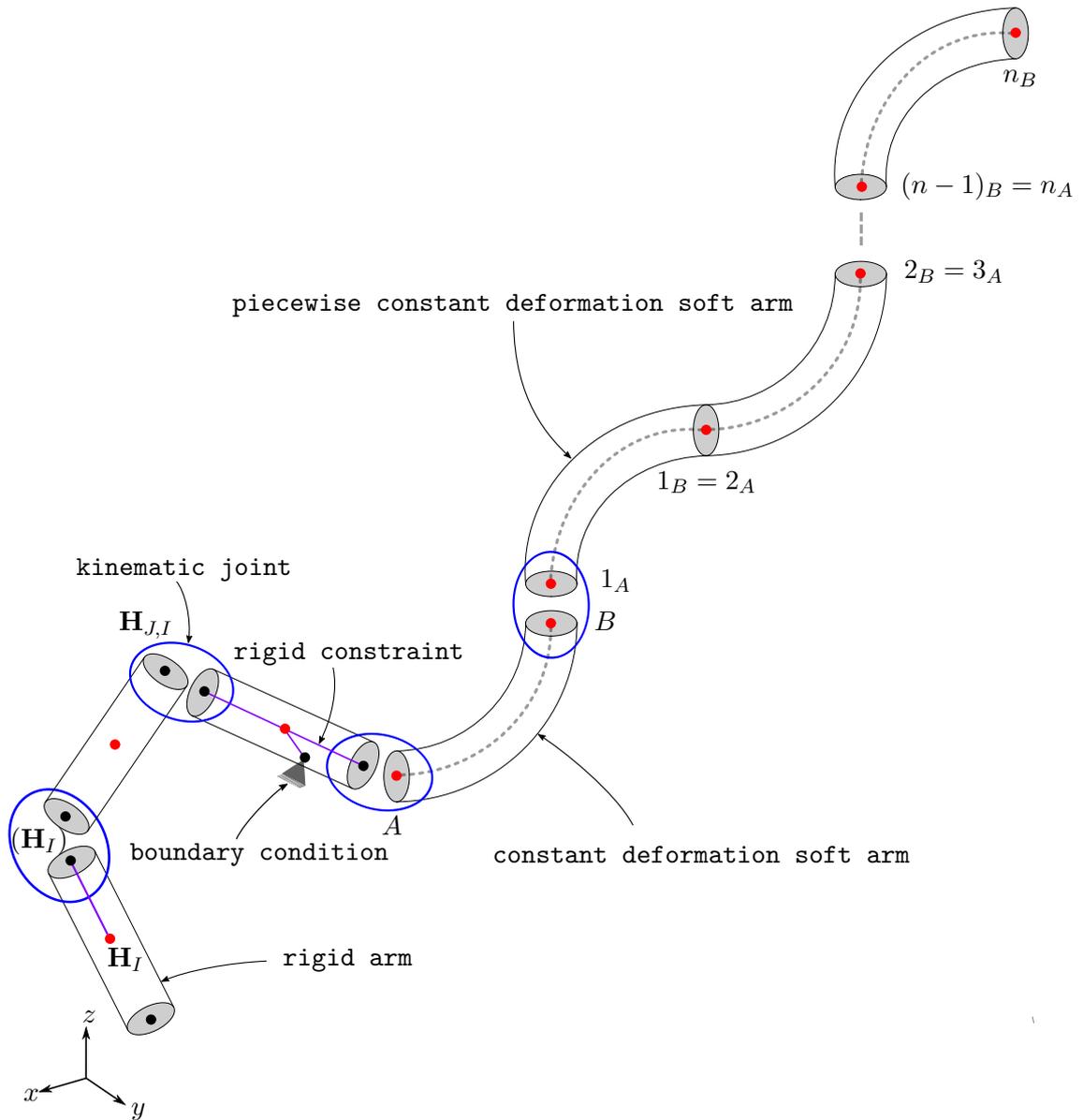


Figure 5.3: A generic robotic mechanism. \mathbf{H}_I are nodal frames, $\mathbf{H}_{J,I}$ are relative frames and (\mathbf{H}_I) are optional frames.

5.6 Test examples

In the following we consider some simple mechanisms as test examples of the constrained dynamics.

5.6.1 Trees

According to the definition made by Selig in [43], tree and star mechanisms are characterized by having several terminal links and no loops. The terminal links are called *leaf-links*, and they have a single joint. All other links are *internal links* and these have at least two, possibly many, joints. A tree mechanism is grounded, i.e., one of the links is connected to a fixed base through a single joint, while star mechanisms are not grounded. This is the only difference between the two. Examples of star mechanisms include multi-leg walking robots and multiple body spacecrafts. Figure 5.4 shows an example of tree mechanism.

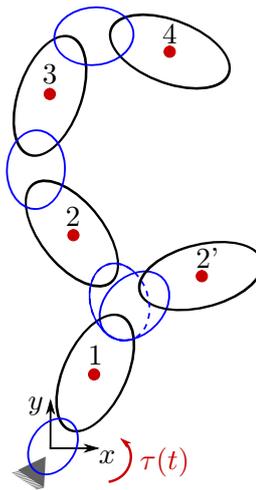


Figure 5.4: A tree structured mechanism.

node	x	y	z	m	$J_{I,11}$	$J_{I,22}$	$J_{I,33}$
1	0.1	0.3	0	0.2	3e-4	3e-4	3e-4
2	0.1	0.8	0	0.2	1e-4	1e-4	1e-4
2'	0.6	0.8	0	0.3	2e-4	2e-4	2e-4
3	0.1	1.25	0	0.15	1e-4	1e-4	1e-4
4	0.35	1.40	0	0.15	1e-4	1e-4	1e-4

Table 5.2: Initial configuration and inertia properties in principal axes for the tree structured mechanism (position in [m], mass in [kg], rotation inertia in [kgm]).

Table 5.2 gives the geometric properties that we use to simulate the motion of the model illustrated in Fig. 5.4. All the joints of the mechanism are revolute joints about the z -axis.

In this simulation test case, a bang-bang input torque of 0.05 Nm is applied in the first joint. The positive torque lasts for 1s, while the negative torque lasts for another second. We observe the motion of the central joint of the tree as well as the motion of node 4 for 2.5 s. The simulated response is illustrated in Fig. 5.5.

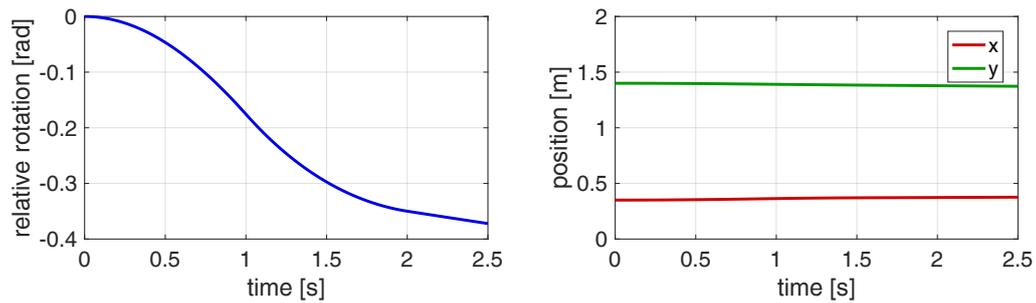


Figure 5.5: Motion of the central joint and node 4 of the tree structured mechanism.

5.6.2 Loops

If the motion of a leaf-link of a tree mechanism is constrained, we have a closed-loop mechanism [43]. The manipulators with internal loops are called parallel manipulators. One of the most classical example of parallel mechanism is the four-bar mechanism, illustrated in Fig. 5.6.

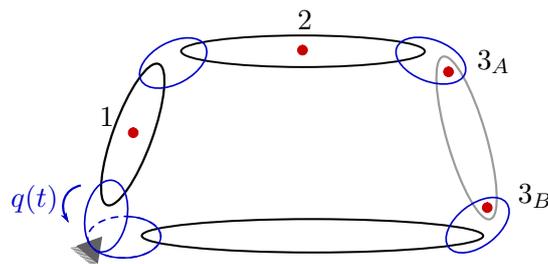


Figure 5.6: A four-bar mechanism.

Table 5.3 gives the geometric properties that we use to simulate the motion of the model illustrated in Fig. 5.6. All the joints of the mechanism are revolute joints about the z -axis. The element which connects the nodes 3_A and 3_B is a constant deformation soft arm. Its cross-section properties are given by

- $\mathbf{M} = \text{diag}(0.0067, 0.0067, 0.0067, 4.5083e - 10, 9.9165e - 10, 4.5083e - 10)$
- $\mathbf{K} = \text{diag}(6.8808e4, 1.7839e5, 6.8808e4, 0.0121, 0.0093, 0.0121)$

node	x	y	z	m	$J_{I,11}$	$J_{I,22}$	$J_{I,33}$
1	0.1	0.3	0	0.1	1e-4	1e-4	1e-5
2	0.6	0.8	0	0.2	4e-4	4e-4	4e-5
3_A	1	1	0	-	-	-	-
3_B	1.2	0	0	-	-	-	-

Table 5.3: Initial configuration and inertia properties in principal axes for the four bar mechanism (position in [m], mass in [kg], rotation inertia in [kgm]).

We simulate the motion of the four bar mechanism by imposing a simple law to the first joint as $q_1(t) = 0.25 \cdot t^2$ for 2.5 s. Figure 5.7 illustrates the simulated motion of the third joint as well as the motion of node 2 of the mechanism.

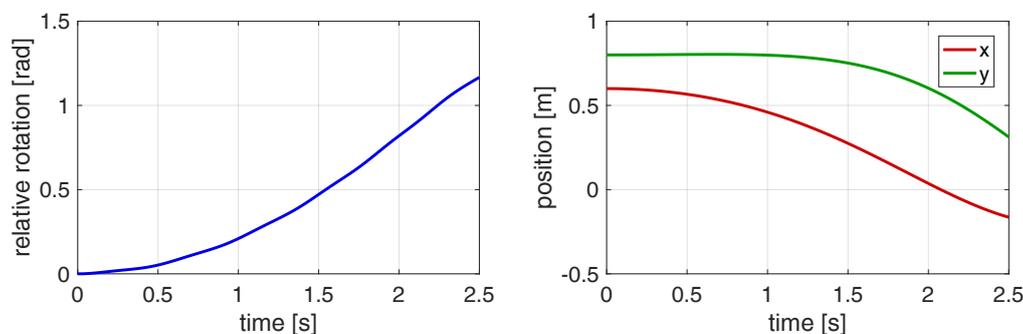


Figure 5.7: Motion of the third joint and node 2 of the loop structured mechanism.

5.6.3 A generic robotic mechanism

As an illustrative example, let us consider the generic robotic mechanism shown in Fig. 5.8. It consists of rigid (solid lines) and soft (dashed lines) elements connected by revolute joints (blue circles) in a kinematic structure which includes one tree and one constrained tree, i.e., a closed loop. The two dark gray dashed elements ($3'4'$ and 56) have been modeled using the constant deformation (CD) model. The light gray dashed element (from node 6 to node 9) has been modeled using the piecewise constant deformation (PWCD) model with three segments. Table 5.4 gives the geometric properties of the model, while Tables 5.5 and 5.6 give the mass and stiffness matrices of the cross-sections of the soft arms.

Not all the revolute joints of the manipulators are actuated. The four actuated joints are indicated in the Figure with $q_i(t), i = 1, \dots, 4$, and they are actuated using a trapezoidal velocity profile for 2 s with the following conditions:

- $q_1(0) = 0; \quad q_1(2) = \pi/6; \quad \ddot{q}_1 = 1;$
- $q_2(0) = 0; \quad q_2(2) = \pi/6; \quad \ddot{q}_2 = 1;$

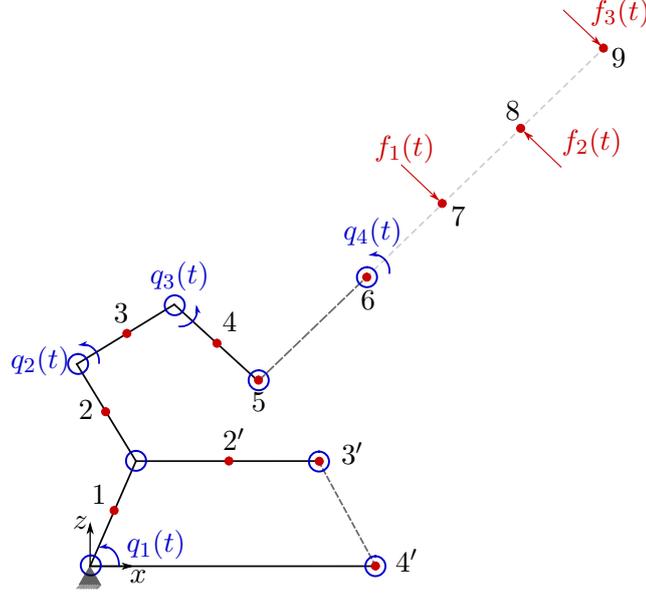


Figure 5.8: Schematic model of a generic robotic mechanism composed by rigid and soft arms structured in a parallel and serial topology.

node	x	y	z	m	$J_{I,11}$	$J_{I,22}$	$J_{I,33}$
1	0.1	0.3	0	0.2	3e-4	3e-4	3e-4
2'	0.6	0.8	0	0.3	2e-4	2e-4	2e-4
3'	1	1	0	-	-	-	-
4'	1.2	0	0	-	-	-	-
2	0.1	0.8	0	0.2	1e-4	1e-4	1e-4
3	0.1	1.25	0	0.15	1e-4	1e-4	1e-4
4	0.1	1.25	0	0.15	1e-4	1e-4	1e-4
5	0.5	1.3	0	-	-	-	-
6	0.8	1.6	0	-	-	-	-
7	1	1.8	0	-	-	-	-
8	1.2	2	0	-	-	-	-
9	1.4	2.2	0	-	-	-	-

Table 5.4: Initial configuration and inertia properties in principal axes for the generic robotic mechanism (position in [m], mass in [kg], rotation inertia in [kgm]).

- $q_3(0) = 0$; $q_3(2) = \pi/2$; $\ddot{q}_3 = 3$;
- $q_4(0) = 0$; $q_4(2) = \pi/12$; $\ddot{q}_4 = 0.3$;

The nodes 7, 8, 9 which correspond to the end-elements of each constant deformation segment of the PWCD arm, are actuated with three linear forces which follow a trapezoidal profile with $\ddot{f} = \pm 100$ and

- $f_{1,x}(0) = 0$; $f_{1,x}(2) = 100$; $f_{1,y}(0) = 0$; $f_{1,y}(2) = -100$;

element	M_{11}	M_{22}	K_{33}	K_{44}	K_{55}	K_{66}
$CD_{3'4'}$	0.12	0.12	0.12	0.35	1.60	0.11
CD_{56}	0.10	0.10	0.10	0.30	1.40	0.09
$PWCD_1$	0.08	0.08	0.08	0.26	1.35	0.08
$PWCD_2$	0.08	0.08	0.08	0.22	1.28	0.06
$PWCD_3$	0.07	0.07	0.07	0.19	1.21	0.06

Table 5.5: Mass matrices for the generic robotic mechanism.

element	K_{11}	K_{22}	K_{33}	K_{44}	K_{55}	K_{66}
$CD_{3'4'}$	3.2e6	0.9e6	1.1e6	3.83	32.8	2.85
CD_{56}	2.8e6	0.6e6	0.9e6	3.12	36.2	2.42
$PWCD_1$	2.8e6	0.6e6	0.9e6	3.06	34.6	2.12
$PWCD_2$	2.5e6	0.6e6	0.8e6	3.02	32.2	2.12
$PWCD_3$	2.5e6	0.6e6	0.8e6	3.02	31.8	2.06

Table 5.6: Stiffness matrices for the generic robotic mechanism.

- $f_{2,x}(0) = 0$; $f_{2,x}(2) = -100$; $f_{2,y}(0) = 0$; $f_{2,y}(2) = 100$;
- $f_{3,x}(0) = 0$; $f_{3,x}(2) = 100$; $f_{3,y}(0) = 0$; $f_{3,y}(2) = -100$;

We observe the motion of the generic robotic mechanism for 2.5 s. We present three different conditions, which differ for the joint modeling aspect. The second simulation adds a spring element for each revolute joint ($K=10$ Nm/rad), while the third simulation adds a spring and a damper elements for each revolute joint ($K=10$ Nm/rad, $D=10$ Nms/rad). Figure 5.9 shows the motion of the free joint which connects the rigid part and the soft part of the mechanism (in correspondence of node 5), as well as the tip displacements of node 9.

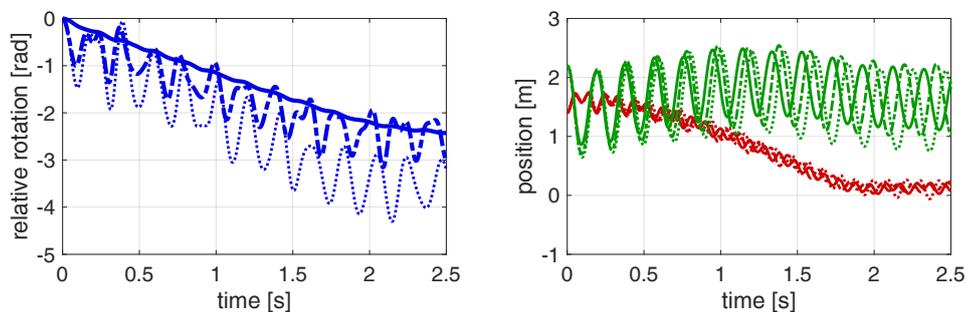


Figure 5.9: Motion of the free joint and the tip node of the generic manipulator. Blue: joint; red: x; green: y. Dotted: rigid joints; Dashdotted: adding of spring elements; Solid: adding of spring and damper elements.

5.7 Summary

- The *equations of motion* of a generic robotic mechanism are given by

$$\dot{\mathbf{H}} = \mathbf{H}\widetilde{\mathbf{A}}\boldsymbol{\eta} \quad (5.58)$$

$$\mathbf{g}_{inc}(\mathbf{H}, \boldsymbol{\eta}, \dot{\boldsymbol{\eta}}) + \mathbf{g}_{int}(\mathbf{H}) + \mathbf{A}^T \boldsymbol{\varphi}_q^T(\mathbf{H})\boldsymbol{\lambda} - \mathbf{g}_{ext}(\mathbf{H}) = \mathbf{0}_{(6n+k) \times 1} \quad (5.59)$$

$$\boldsymbol{\varphi}(\mathbf{H}) = \mathbf{0}_{6m \times 1} \quad (5.60)$$

- The discretized equations of motion of a generic robotic mechanism are given by

$$\mathbf{H}_{n+1} = \mathbf{H}_n \exp_{SE(3)}(\widetilde{\mathbf{A}}\mathbf{n}_{n+1}) \quad (5.61)$$

$$\mathbf{g}(\mathbf{H}_{n+1}, \boldsymbol{\eta}_{n+1}, \dot{\boldsymbol{\eta}}_{n+1}) + \mathbf{A}^T \boldsymbol{\varphi}_q^T(\mathbf{H}_{n+1})\boldsymbol{\lambda}_{n+1} = \mathbf{0}_{(6M+k) \times 1} \quad (5.62)$$

$$\boldsymbol{\varphi}(\mathbf{H}_{n+1}) = \mathbf{0}_{6m \times 1} \quad (5.63)$$

Chapter 6

SimSOFT: a physics engine for soft robots

SimSOFT is the c++ physics engine for soft robots which implements the finite element formulation for robotics systems discussed in the thesis. In this Chapter, we first describe its main features, after we present some benchmark problems taken from the literature, and we illustrate its potential through basic application examples, i.e., (i) the dynamic simulations of robotics systems in three different scenarios; (ii) the development of computationally efficient mathematical models for soft robots by using deep learning techniques; (iii) the development of vibration control strategies for remote procedures in hazardous domains.

6.1 Description

The main aspects of the architecture of SimSOFT are summarized in Fig. 6.1. The geometric formalism described in Chapter 1 is the key-concept of the framework. Based on this, the main elements of the mechanical library are developed, i.e., the rigid and soft arm models. The continuum formulation of the soft arms produces *partial-differential* equations (PDE) which can not be solved using the architecture of modern computers. For this reason, we discretize the equations of motion by using the *piecewise constant deformation assumption* in the context of the finite element method. In this way, we achieve a set of *ordinary-differential* equations (ODE) which describe the equilibrium of the elements. In order to couple rigid and soft arms in the same dynamic analysis, we model the connecting joints as algebraic constraints, which limit the motion of the resulting mechanism according to the particular joint. By assembling in the same model all the main components of the library, namely the rigid and soft arms and the rigid and flexible joints, we produce a set of *differential-algebraic* equations (DAE). The resulting equations of motions need to be integrated in time using an implicit time integration scheme. Finally, some post-processing tools allow to plot the kinematics quantities and the forces and torques of nodes as well as the strains and the stresses of the elements for the duration of the simulation.

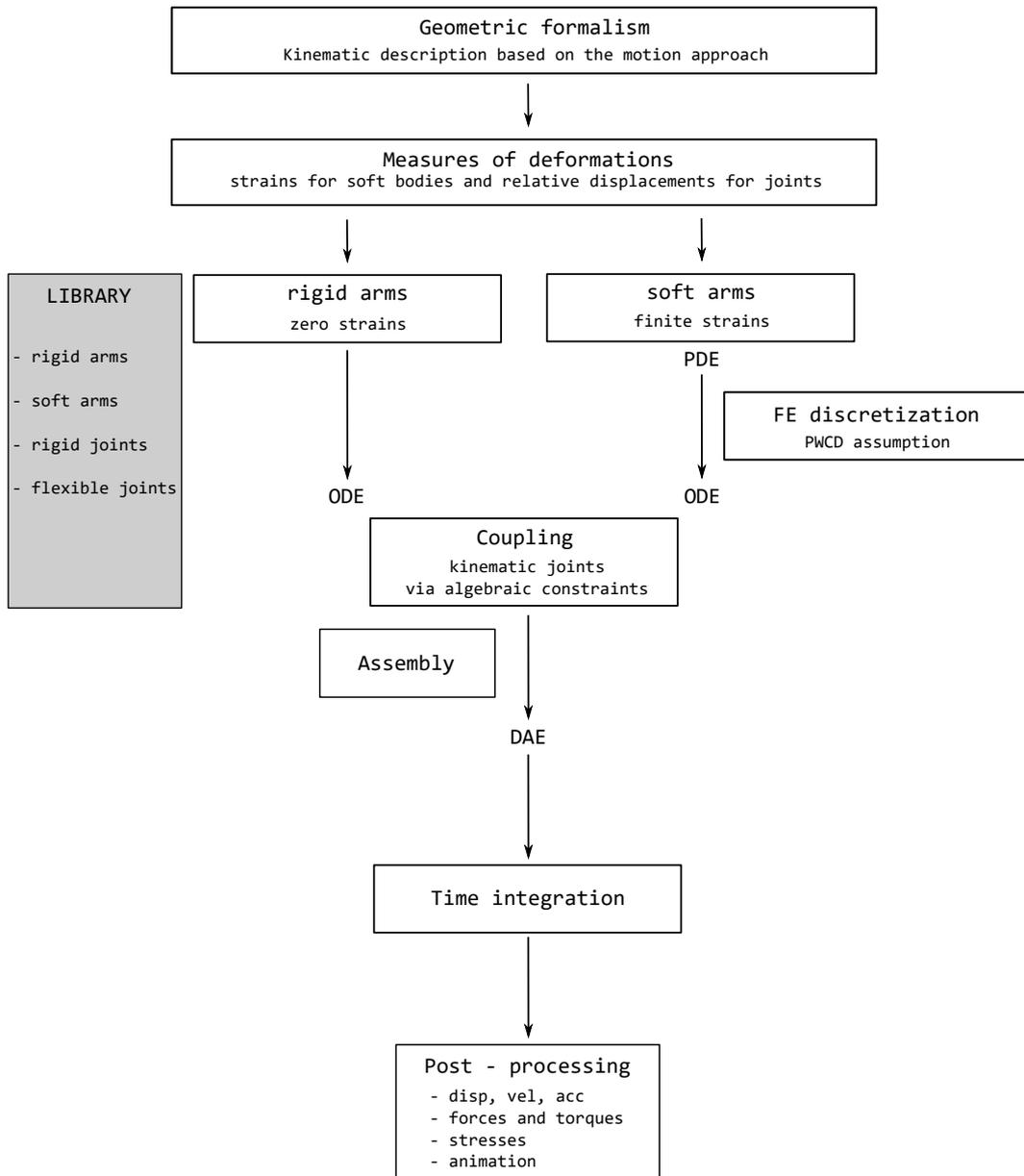


Figure 6.1: Conceptual architecture of SimSOFT.

6.2 Benchmark problems

Planar two-link rigid manipulator

Let us consider a planar two-link rigid manipulator with the physical parameters given in Table 6.1 subject to the joint motion

- $q_1(t) = t$; $q_2(t) = -t/4$, $0 \leq t \leq 2\pi s$

link	l	m	I
1	0.25	25	5
2	0.25	25	5

Table 6.1: Physical parameters for the two-link rigid manipulator (length in [m], mass in [kg], rotation inertia in [kg m²]).

Figure 6.2 illustrates the evolution, evaluated in the inertial reference frame, of the tip-displacements of the manipulator obtained using the classical rigid-body dynamics [3] and the dynamics implemented in SimSOFT.

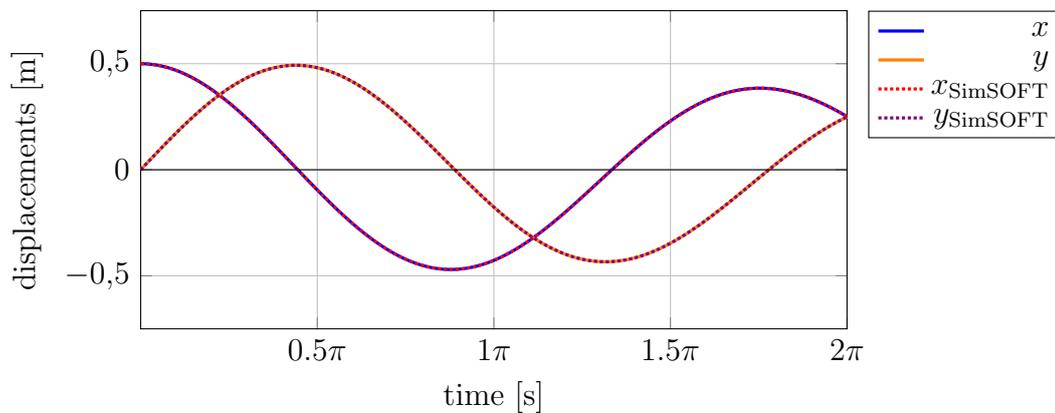


Figure 6.2: Displacements of the tip-position of the two-link rigid manipulator in the inertial reference frame.

Free-falling flexible double pendulum

This simulation refers to the dynamic analysis of a flexible double pendulum with two revolute passive joints, namely the benchmark problem of the Multibody Dynamics module of Comsol Multiphysics [62]. The initial configuration of the flexible double pendulum is given in Table 6.2 and it is illustrated in Fig. 6.3, while Table 6.3 defines the two kinematic joints. The dimensions of the rectangular cross-sections of the two links are: $b = 0.5$ m and $h = 1$ m. The material is structural steel with density $\rho = 7850$ kg m³ and Young modulus $E = 200 \times 10^9$ Pa.

In the simulation case, the mechanical system just falls down, being subjected to gravity in the z - direction. The motion of the mechanism is observed for 20 s. Figure 6.4 shows the comparison of the joint displacements recorded using our 1D FEM formulation with respect to the ones achieved using the 3D FEM formulation implemented in a commercial software, Comsol Multiphysics. The slightly upfront response of the 3D simulation might be due to an overdamped integrator which the commercial software use for solving the equations of motion.

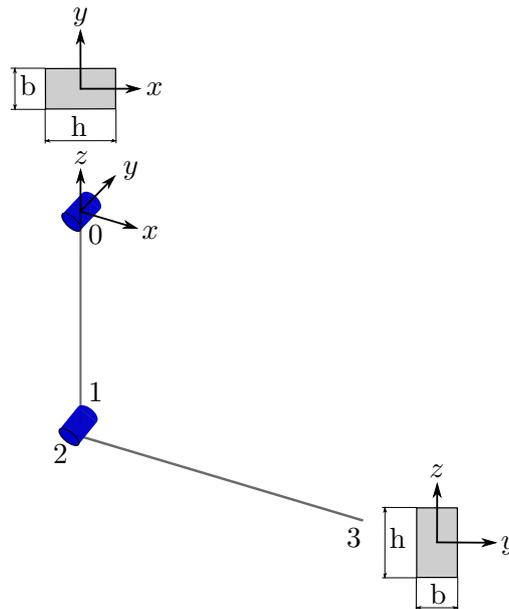


Figure 6.3: Flexible double pendulum in its initial configuration.

node	x	y	z
0	0	0	0
1	0	0	-10
2	0	-1.25	-10
3	10	-1.25	-10

Table 6.2: Initial configuration of the flexible double pendulum. Position in [m].

joint	k -th column of \mathbf{A}_J	\mathbf{e}_U	\mathbf{e}_Ω
A	1	$\mathbf{0}_{3 \times 1}$	$[0 \ 1 \ 0]^T$
B	1	$\mathbf{0}_{3 \times 1}$	$[0 \ 1 \ 0]^T$

Table 6.3: Kinematic joint definition of the flexible double pendulum.

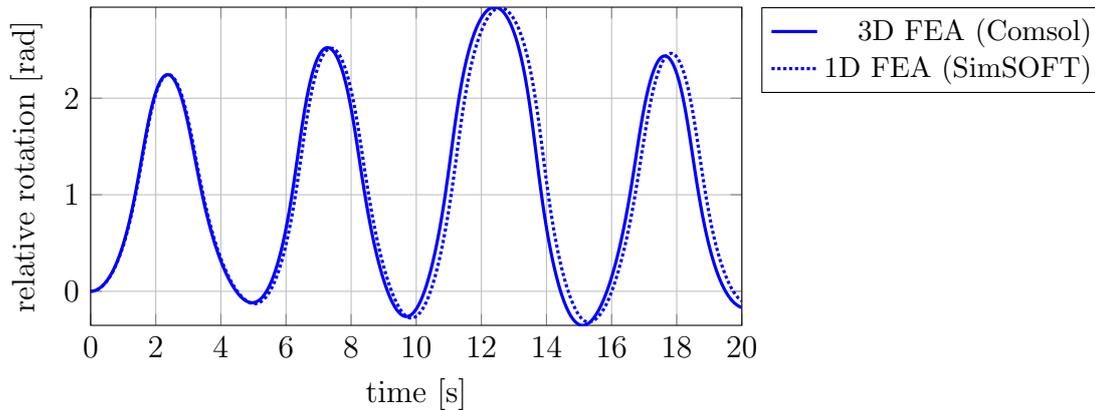


Figure 6.4: Relative angular position of the revolute joint for the flexible double pendulum. Solid: 1D FEM, dashdotted: 3D FEM

Spatial flexible manipulator

This benchmark problem is the dynamic analysis of a spatial manipulator with two flexible links and two active rigid joints, the case study from [63, 64]. The initial configuration of the manipulator is given in Table 6.4 and it is illustrated in Fig. 6.5, while Tables 6.5 defines the two kinematic joints. The dimensions of the rectangular cross-sections of the two links are: $b = 5 \times 10^{-3}$ m and $h = 7 \times 10^{-3}$ m. The material is aluminium alloy with density $\rho = 2700$ kg m³, Young modulus $E = 69 \times 10^9$ Pa and shear modulus $G = 26.9 \times 10^9$ Pa. Three simulation cases are presented, corresponding to the three point-to-point joint trajectories indicated in Table 6.6. The motion from each of the two points of the trajectory follows a bang-bang acceleration profile of 2 s, which leads to a *S-curve* in the positions. The positive acceleration lasts for 1 second followed by a negative acceleration for the same time duration. For the three simulations, $|a| = |v_{max}| = |s_f|$, i.e. the value of the joint acceleration is equal to the maximum value of the joint velocity and the final position of the joint. The system presents no damping and it is not subjected to gravity. The motion of the mechanism is observed for 5 s. Figures 6.6, 6.7, 6.8 show the tip displacements for the three moves. The resulting trajectories are in close agreement with the ones that Kivila et al. in [64] achieved by using system modes, 3D FEA, and experiments.

node	x	y	z
0	0	0	0
1	1	0	0
2	1	0	-1

Table 6.4: Initial configuration of the spatial flexible manipulator. Measurements in [m].

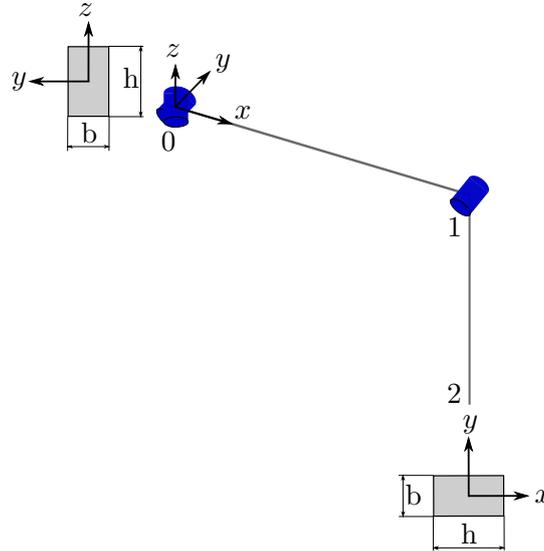


Figure 6.5: The spatial flexible manipulator, constituted by two links with two active joints, one universal, one revolute.

joint	k -th column of \mathbf{A}_J	\mathbf{e}_U	\mathbf{e}_Ω
A	1	$\mathbf{0}_{3 \times 1}$	$[0 \ 0 \ 1]^T$
	2	$\mathbf{0}_{3 \times 1}$	$[0 \ 1 \ 0]^T$
B	1	$\mathbf{0}_{3 \times 1}$	$[0 \ 1 \ 0]^T$

Table 6.5: Kinematic joint definition of the spatial flexible manipulator.

simulation	A_1	A_2	B
1	0 to -1	0 to -1	0 to -1
2	0 to 0	0 to -1	0 to -1
3	0 to π	0 to 0	0 to 0

Table 6.6: Joint initial and final positions for the three simulations. Measurements in [rad]

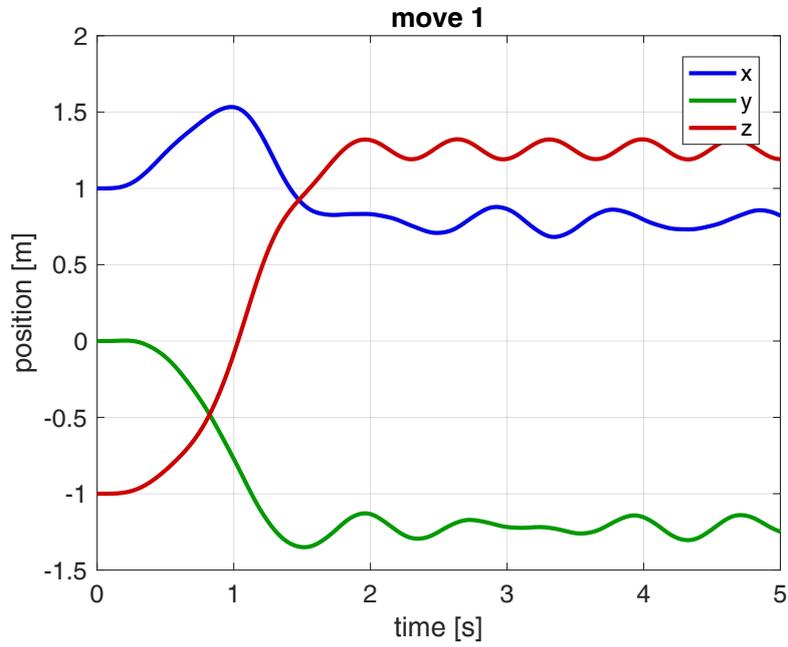


Figure 6.6: Tip displacements of the spatial flexible manipulator for move 1

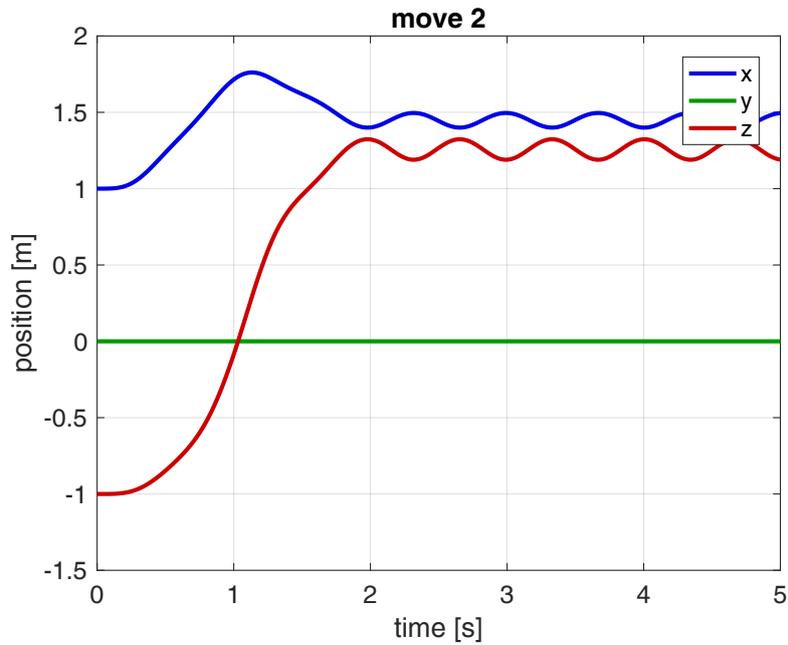


Figure 6.7: Tip displacements of the spatial flexible manipulator for move 2

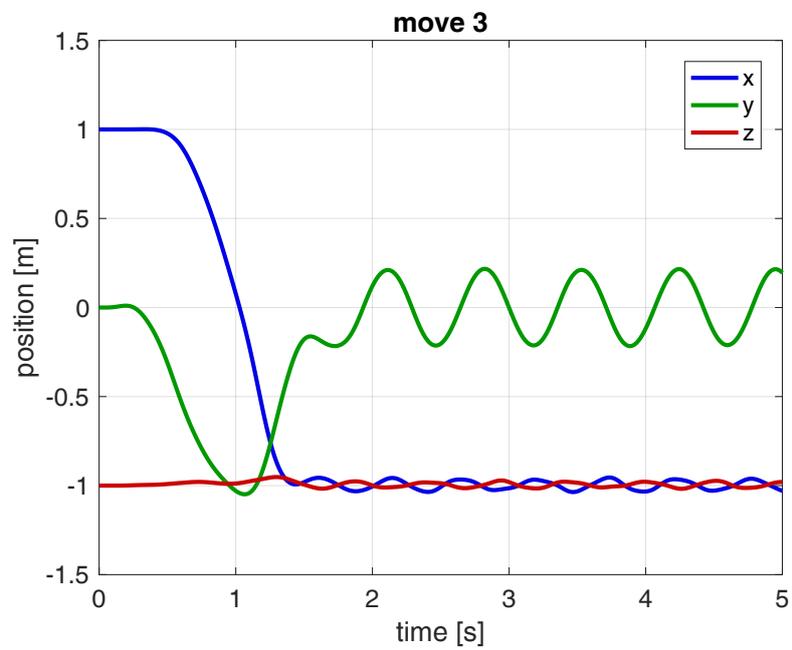


Figure 6.8: Tip displacements of the spatial flexible manipulator for move 3

Princeton experiment

We replicate in simulation the Princeton experiment in [65], which is a great test bed for the constant deformation soft arm model. The experimental setup comprises a cantilevered arm subjected to the out-of-plane load P , applied at different angle conditions, which induce coupled effects of bending, torsion and shear. The setup is illustrated in Fig. 6.9. The length of the arm is $L = 0.508$ m and the cross-section properties are given in Table 6.7. The load conditions are given by

- $P_1 = 4.448$ N; $P_2 = 8.896$ N; $P_3 = 13.345$ N
- $\theta_i = 15i$ deg, $i = 0, 1, \dots, 6$

element	11	22	33	44	55	66
M	0.1062	0.1062	0.1062	0.3038	1.356	0.09078
K	2.842e6	0.6401e6	0.9039e6	3.103	36.28	2.429

Table 6.7: Cross-section properties of the Princeton experiment.

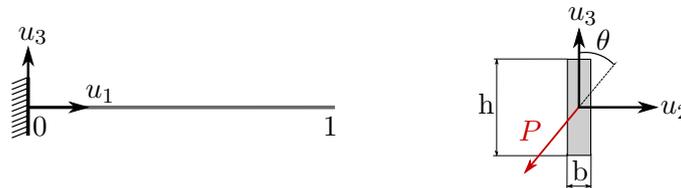


Figure 6.9: Schematic model of the Princeton experiment, a cantilevered arm subjected to the load P .

Figure 6.10 illustrates the comparison between the experimental response and the simulated response of the tip displacements, for the different loading conditions. A close agreement is observed.

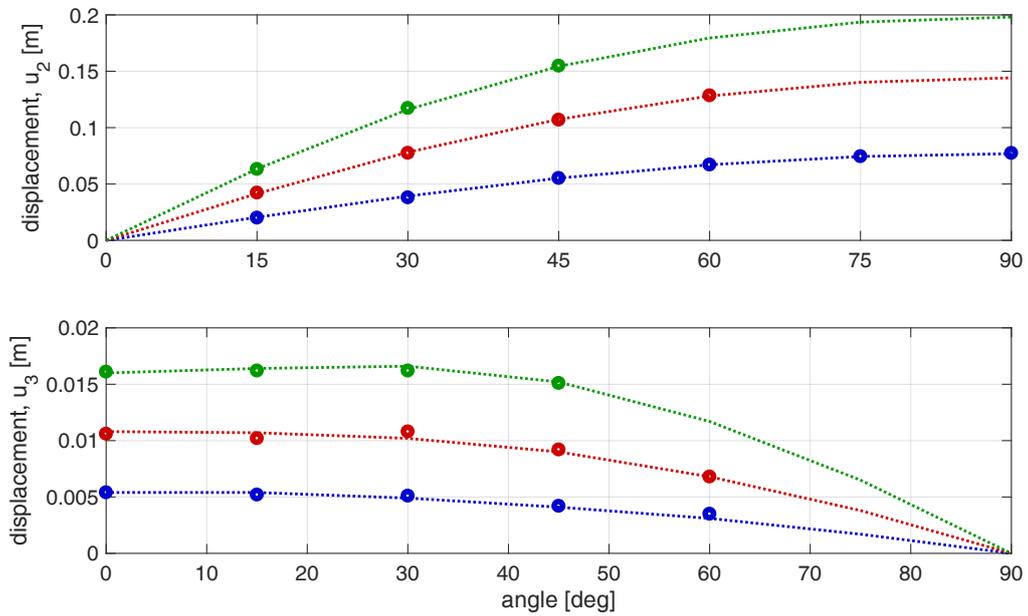


Figure 6.10: Tip displacements of the Princeton cantilevered arm subjected to the end load P with three different intensities and seven configurations. Blue: P_1 , red: P_2 , green: P_3 . Scatter: experimental data, dotted: numerical predictions from SimSOFT.

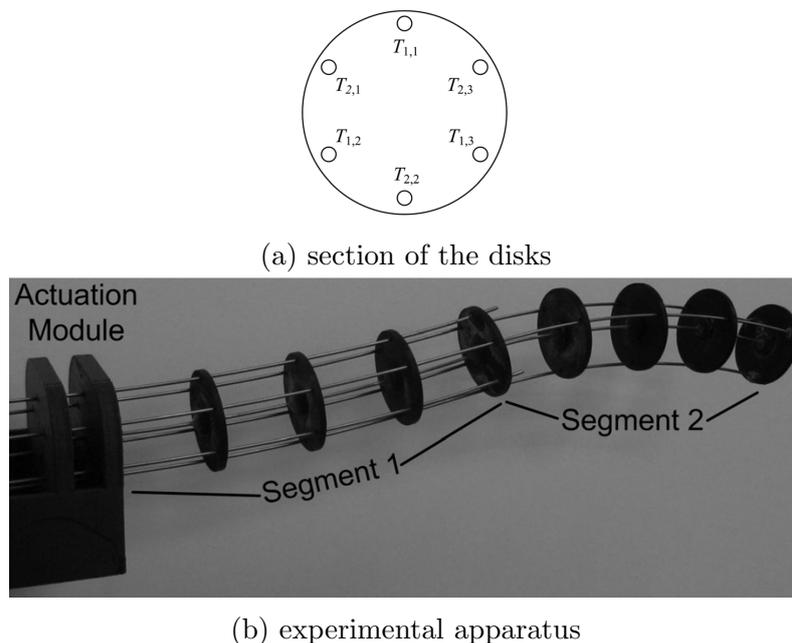


Figure 6.11: Rod-driven, two-segment prototype from Rone and Ben-Tzvi.

Multisegment rod-driven continuum manipulator

In this example, we simulate the experiments performed by Rone and Ben-Tzvi in [34] on a multisegment rod-driven continuum manipulator. The experimental apparatus is illustrated in Fig.6.11. It comprises two continuum backbones with eight disks rigidly mounted along the elastic core at a distance of 30 mm to each other, with six actuation rods which control the shape of the overall manipulator. Three rods terminate at the fourth disk, and three rods terminate at the eighth disk. The initial configuration of the manipulator is straight, with the elastic backbone core along the x - axis, while z - axis is the usual vertical axis which is in the same direction of the gravity vector. The cross-section properties of the central backbone are given in Table 6.8, while Table 6.9 reports the overall inertia properties of the two segments.

element	11	22	33	44	55	66
M	0.0067	0.0067	0.0067	9.0165e-10	4.5083e-10	4.5083e-10
K	1.7839e5	6.8808e4	6.8808e4	0.093	0.0121	0.0121

Table 6.8: Cross-section properties of the central backbone of the multisegment rod-driven manipulator.

In the experimental benchmark, the manipulator is subjected to an applied tension of 25 N in rod 1-1 and tension of 5, 10, and 15 N in rod 2-2. The central backbone has been modeled using the PWCD model, with $n=2$.

rigid body	m_I	$J_{I,11}$	$J_{I,22}$	$J_{I,33}$
segment 1	2.869e-3	3.531e-7	7.690e-6	7.690e-6
segment 2	2.269e-3	2.593e-7	3.894e-6	3.894e-6

Table 6.9: Inertia properties in principal axes for the two segments of the multi-segment rod-driven manipulator (mass in [kg] and rotation inertia in [kgm²])

Figure 6.12 shows the simulated response of the experiment in SimSOFT. The results are in close agreement with the ones achieved experimentally by Rone and Ben-Tzvi in [34], in Sec. 6.6.

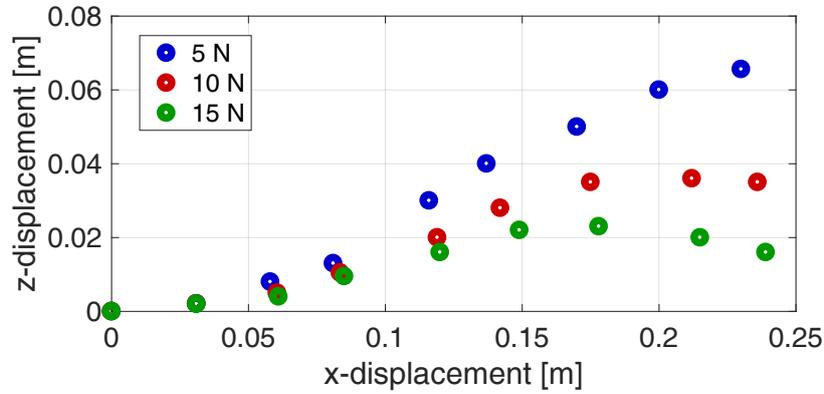


Figure 6.12: Static equilibrium for the multisegment rod-driven continuum manipulator.

6.3 Applications

Three applications are presented to show how a general purpose physics engine for rigid and soft robotics systems might help in: dynamic simulations, learning from simulation, vibration control.

6.3.1 Dynamic simulations in three scenarios: nuclear fusion, surgical, rehabilitation

The main advantages which historically have motivated the development of flexible, even continuum, manipulators are their ability to:

- access by remotely in complex environments: this ability is appealing in maintenance, inspection and repair (MIR) operations, and in minimally invasive surgery (MIS).
- adapt their shape to perform whole-arm manipulation: soft robots might help in applications which require a soft and safe physical human robot interaction (s^2 -pHRI).

For this reason, we select three scenario in which the use of such robotics systems is of current interest: nuclear fusion, surgical, rehabilitation.

Nuclear fusion

Robotics in nuclear fusion plays a relevant role in maintenance and inspection of mechanical components inside fusion vessels, which are challenging to access [66]. Snake and hyper-redundant manipulators have been widely used in the past and recent years. In this work we describe the dynamic analysis of an hybrid robotic mechanism that has been recently proposed for DEMO multi-module blanket segment (MMS) remote handling [67]. Due to its weight and the difficulty to reach it inside the fusion reactor, MMS is the most challenging in-vessel component from the maintenance perspective. Currently, the proposed strategy for DEMO remote maintenance is based on a vertical scheme, which is illustrated in Fig. 6.13 and explained in [68]. Following this approach, the concept for a vertical maintenance system for multi-module blanket segments was proposed in [69]. It involves the use of a vertical maintenance crane which transports, through the vertical maintenance ports, a robotic mechanism which has to operate inside the vessel for the effective maintenance.

The hybrid kinematic mechanism (HKM) which have been selected for DEMO MMS remote handling is shown in Fig. 6.14, while Fig. 6.15 reports a snapshot of the HKM in SimSOFT. The mechanism comprises three linear actuators (T_i) which position the mechanism in space. The three revolute joints A , B and C allow for a rotation about the three axis x , y and z . Note that the joint B is not actuated: the rotation about y -axis is provided by the synchronous movements of the two linear actuators L_i . For a more detailed description, see [67].

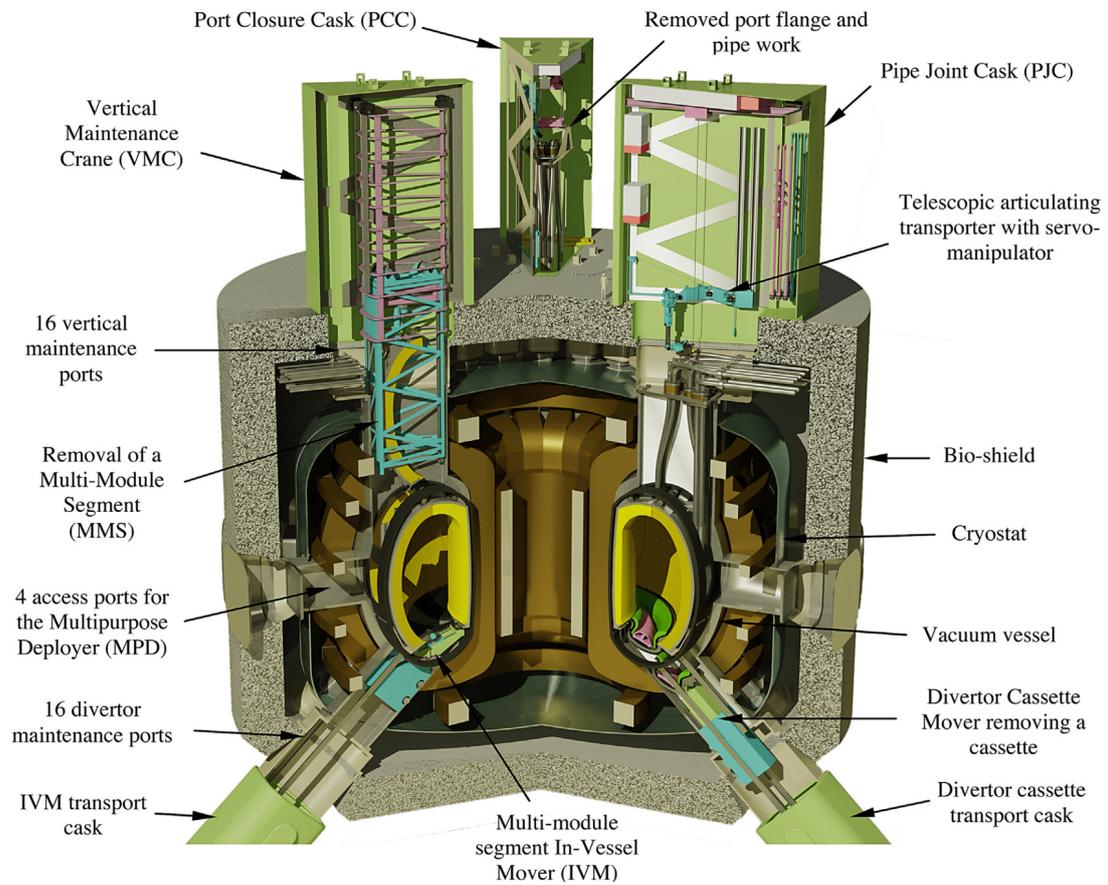


Figure 6.13: A qualitative picture of the DEMO vertical maintenance system architecture.

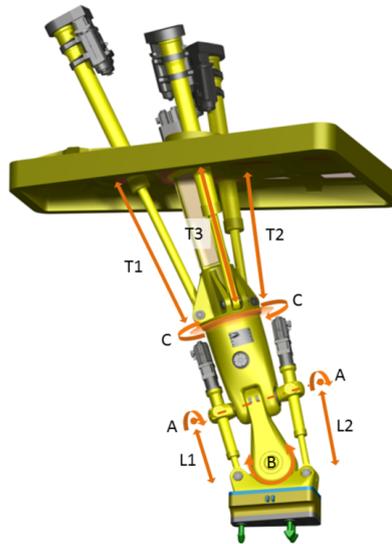


Figure 6.14: Hybrid kinematic mechanism for DEMO MMS remote handling

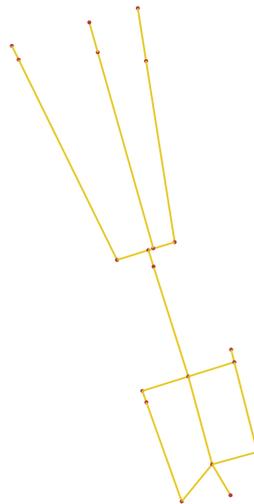


Figure 6.15: Snapshot of the hybrid kinematic mechanism within SimSOFT

The kinematic model of the HKM is illustrated in Fig. 6.16. It comprises four rigid bodies, five universal joints, five prismatic joints, five hinges and three revolute joints. Tables 6.10, 6.11 and 6.12 give the geometric data and the initial configuration of the HKM. Table 6.13 reports the inertia properties of the rigid bodies, while Tables 6.14, 6.15 and 6.16 give the definitions of the geometric properties for the soft elements, as well as the mass and stiffness matrices. Finally, 6.17 defines the joints of the HKM.

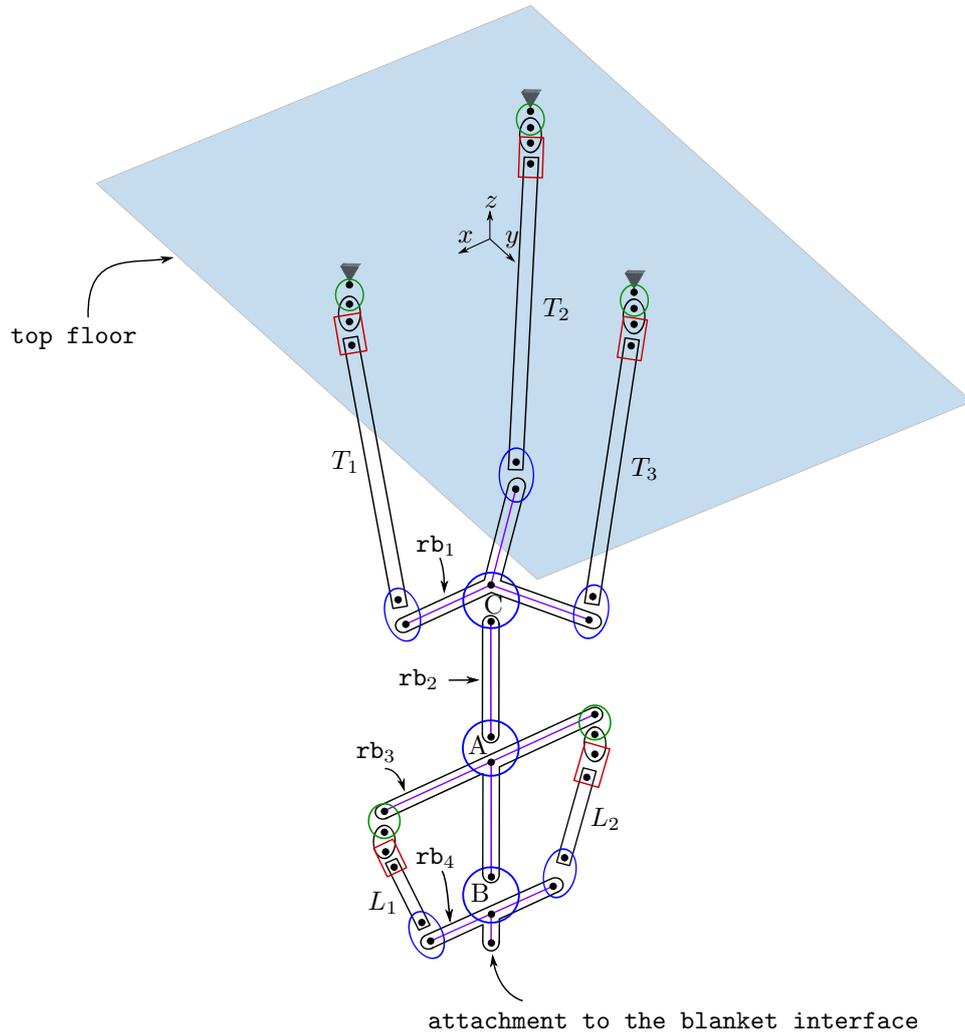


Figure 6.16: Kinematic model of the hybrid kinematic mechanism

parameter	length [m]
l_0	1.732
l_{01}	0.823
l_2	1.350
l_4	1.250
z_{01}	-3.200
z_1	-3.435
z_2	-5.000
z_3	-6.250
z_4	-6.450
z_5	-6.750

Table 6.10: Geometric linear data of the hybrid kinematic mechanism

parameter	value [rad]
$\alpha_{xz,T1}$	0.1626
$\alpha_{xz,T2}$	0.0818
$\alpha_{yz,T2}$	0.1411
$\alpha_{xz,T3}$	0.0818
$\alpha_{yz,T3}$	0.1411
$\alpha_{xz,L1}$	0.0345
$\alpha_{xz,L2}$	0.0345

Table 6.11: Geometric angular data of the hybrid kinematic mechanism

node	x	y	z
$0_{1_{cl}}, 0_1, 0_{1_{bis}}, T_{1_{up}}$	$l_0/2/\cos(\pi/6)$	0	0
$0_{2_{cl}}, 0_2, 0_{2_{bis}}, T_{2_{up}}$	$-l_0/2/\tan(\pi/6)$	$-l_0/2$	0
$0_{3_{cl}}, 0_3, 0_{3_{bis}}, T_{3_{up}}$	$-l_0/2/\tan(\pi/6)$	$l_0/2$	0
$01_{0_{rb1}}$	0	0	z_{01}
$01_1, T_{1_{dw}}$	$l_{01}/2/\cos(\pi/6)$	0	z_{01}
$01_2, T_{2_{dw}}$	$-l_{01}/2/\tan(\pi/6)$	$-l_{01}/2$	z_{01}
$01_3, T_{3_{dw}}$	$-l_{01}/2/\tan(\pi/6)$	$l_{01}/2$	z_{01}
$1_{0_{rb2}}$	0	0	z_1
2_0	0	0	z_2
$2_{0_{rb3}}$	0	0	z_2
$2_{1_{rb3}}, 2_1, 2_{1_{bis}}, L_{1_{up}}$	$l_2/2$	0	z_2
$2_{2_{rb3}}, 2_2, 2_{2_{bis}}, L_{2_{up}}$	$-l_2/2$	0	z_2
3_0	0	0	z_3
$3_{0_{rb4}}$	0	0	z_3
$4_1, L_{1_{dw}}$	$l_4/2$	0	z_4
$4_2, L_{2_{dw}}$	$-l_4/2$	0	z_4
$5_0 = EE$	0	0	z_5

Table 6.12: Initial configuration of the hybrid kinematic mechanism

rigid body	m_I	$J_{I,11}$	$J_{I,22}$	$J_{I,33}$
rb1	668.55	923.14	923.22	62.60
rb2	486.89	142.08	142.08	72.85
rb3	436.21	155.33	172.47	37.74
rb4	344.68	23.96	68.34	57.83
T_i	53.77	71.00	71.00	0.4133
L_i	12.87	2.965	2.965	0.0647

Table 6.13: Inertia properties in principal axes for the rigid bodies of the hybrid kinematic mechanism (mass in [kg] and rotation inertia in [kgm])

A_T	$\pi(0.170)^2/4$
A_L	$\pi(0.120)^2/4$
I_T	$\pi(0.170)^4/32$
I_L	$\pi(0.120)^4/32$
I_{z_T}	$\pi(0.170)^4/64$
I_{z_L}	$\pi(0.120)^4/64$
ρ	7850
E	210e9
ν	0.33
G	$E/(2(1 + \nu))$

Table 6.14: Geometric data and material properties for the soft elements of the hybrid kinematic mechanism

element	M_{11}	M_{22}	M_{33}	M_{44}	M_{55}	M_{66}
T_i	ρA_T	ρA_T	ρA_T	ρI_T	ρI_T	ρI_{z_T}
L_i	ρA_L	ρA_L	ρA_L	ρI_L	ρI_L	ρI_{z_L}

Table 6.15: Cross-section mass matrices for the soft elements of the hybrid kinematic mechanism

element	K_{11}	K_{22}	K_{33}	K_{44}	K_{55}	K_{66}
T_i	GA_T	GA_T	EA_T	EI_T	EI_T	GI_{z_T}
L_i	GA_L	GA_L	EA_L	EI_L	EI_L	GI_{z_L}

Table 6.16: Cross-section stiffness matrices for the soft elements of the hybrid kinematic mechanism.

joint	k -th column of \mathbf{A}_J	e_U	e_Ω
unjT1	1	$\mathbf{0}_{3 \times 1}$	$[-\cos(\alpha_{xz,T1}) \ 0 \ \sin(\alpha_{xz,T1})]^T$
	2	$\mathbf{0}_{3 \times 1}$	$[0 \ 1 \ 0]^T$
unjT2	1	$\mathbf{0}_{3 \times 1}$	$[\cos(\alpha_{xz,T2}) \ 0 \ \sin(\alpha_{xz,T2})]^T$
	2	$\mathbf{0}_{3 \times 1}$	$[0 \ \cos(\alpha_{yz,T2}) \ \sin(\alpha_{yz,T2})]^T$
unjT3	1	$\mathbf{0}_{3 \times 1}$	$[\cos(\alpha_{xz,T3}) \ 0 \ \sin(\alpha_{xz,T3})]^T$
	2	$\mathbf{0}_{3 \times 1}$	$[0 \ -\cos(\alpha_{yz,T3}) \ \sin(\alpha_{yz,T3})]^T$
prjT1	1	$[\sin(\alpha_{xz,T1}) \ 0 \ \cos(\alpha_{xz,T1})]$	$\mathbf{0}_{3 \times 1}$
prjT2	1	$[-\sin(\alpha_{xz,T2}) \ -\sin(\alpha_{yz,T2}) \ \cos(\alpha_{xz,T2}) \cdot \cos(\alpha_{yz,T2})]$	$\mathbf{0}_{3 \times 1}$
prjT3	1	$[-\sin(\alpha_{xz,T3}) \ \sin(\alpha_{yz,T3}) \ \cos(\alpha_{xz,T3}) \cdot \cos(\alpha_{yz,T3})]$	$\mathbf{0}_{3 \times 1}$
hgjT1	1	$\mathbf{0}_{3 \times 1}$	$[0 \ 1 \ 0]^T$
hgjT2	1	$\mathbf{0}_{3 \times 1}$	$[\cos(\pi/6) \ -\sin(\pi/6) \ 0]^T$
hgjT3	1	$\mathbf{0}_{3 \times 1}$	$[\cos(\pi/6) \ \sin(\pi/6) \ 0]^T$
revjC	1	$\mathbf{0}_{3 \times 1}$	$[0 \ 0 \ 1]^T$
revjA	1	$\mathbf{0}_{3 \times 1}$	$[1 \ 0 \ 0]^T$
revjB	1	$\mathbf{0}_{3 \times 1}$	$[0 \ 1 \ 0]^T$
unjL1	1	$\mathbf{0}_{3 \times 1}$	$[-\cos(\alpha_{xz,L1}) \ 0 \ \sin(\alpha_{xz,L1})]^T$
	2	$\mathbf{0}_{3 \times 1}$	$[0 \ 1 \ 0]^T$
unjL2	1	$\mathbf{0}_{3 \times 1}$	$[\cos(\alpha_{xz,L2}) \ 0 \ \sin(\alpha_{xz,L2})]^T$
	2	$\mathbf{0}_{3 \times 1}$	$[0 \ 1 \ 0]^T$
prjL1	1	$[\sin(\alpha_{xz,L1}) \ 0 \ \cos(\alpha_{xz,L1})]$	$\mathbf{0}_{3 \times 1}$
prjL2	1	$[-\sin(\alpha_{xz,L2}) \ 0 \ \cos(\alpha_{xz,L2})]$	$\mathbf{0}_{3 \times 1}$
hgjL1	1	$\mathbf{0}_{3 \times 1}$	$[0 \ 1 \ 0]^T$
hgjL2	1	$\mathbf{0}_{3 \times 1}$	$[0 \ 1 \ 0]^T$

Table 6.17: Kinematic joint definition for the hybrid kinematic mechanism

As an illustrative example, we perform the dynamic simulation of the HKM in the first point-to-point motion, with triangular velocity profile, which has been planned for the MMS removing process. Starting from the initial configuration given in Table 6.12, the initial and final configuration of the joints (q_i and q_f) as well as the acceleration of the bang-bang profile \ddot{q} and the final time t_f are given as follows

- T1: $q_i = 0$; $q_f = -0.0293$; $\ddot{q} = -0.1172$; $t_f = 1$
- T2: $q_i = 0$; $q_f = 0.2068$; $\ddot{q} = 0.8272$; $t_f = 1$
- T3: $q_i = 0$; $q_f = -0.0720$; $\ddot{q} = -0.2880$; $t_f = 1$
- θ_x : $q_i = 0$; $q_f = 0.0628$; $\ddot{q} = 0.2512$; $t_f = 1$
- θ_y : $q_i = 0$; $q_f = -0.00488$; $\ddot{q} = -0.01952$; $t_f = 1$
- θ_z : $q_i = 0$; $q_f = -0.00314$; $\ddot{q} = -0.01257$; $t_f = 1$

where the linear measurements are given in [m], while the angular ones in [rad]. For a first analysis, we simulate the MMS as a point mass which produces a force weight $Fb_z = -784.5\text{kN}$ applied at the end-effector of the manipulator, along the z -axis.

Fig. 6.17 shows the displacements of the end-effector of the HKM, i.e., the central node of the attachment to the blanket interface, during the simulated test trajectory. Two models are compared: RF, using all rigid elements but flexible prismatic joints, and FR, using flexible elements for the linear actuators but rigid joints. Moreover, we record the resulting reaction forces at the VTS interface, during the motion, as we can see in Fig. 6.18.

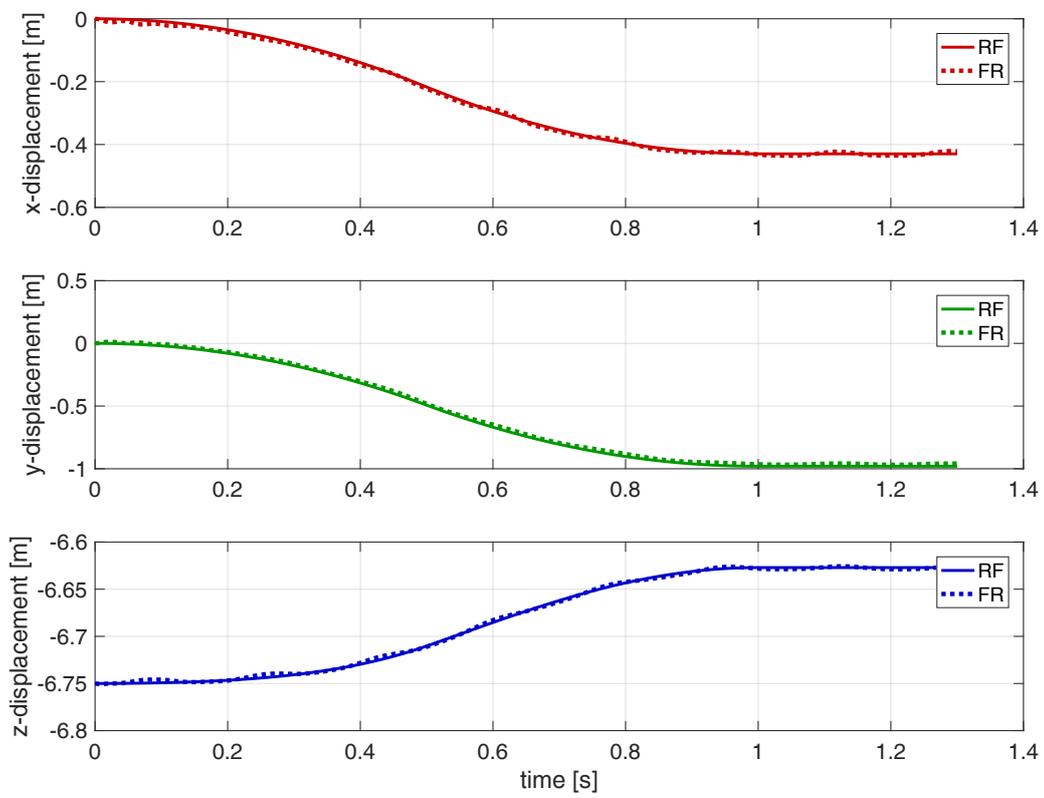


Figure 6.17: Displacements of end-effector of the hybrid kinematic mechanism in the test trajectory.

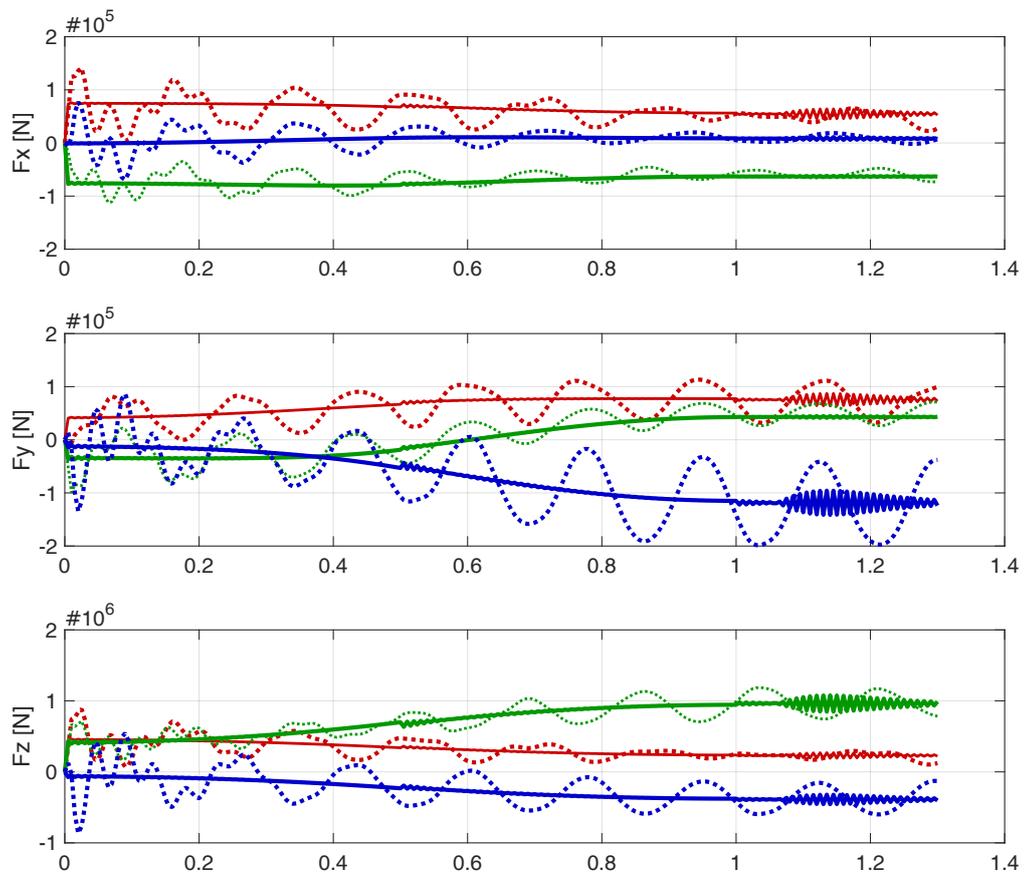


Figure 6.18: Reaction forces of the hybrid kinematic mechanism at the boundaries with the vertical transport system interface, during the test trajectory. Red: 01cl; Green: 02cl; Blue: 03cl. Solid: RF; Dotted: FR.

Surgical

The second scenario is robotic surgery. In this context, flexible, even continuum, manipulators are used for minimally invasive intervention in many procedures [16]. Recently, the need of working through smaller incision within smaller body cavities motivates the development of miniaturized continuum instruments also for laparoscopic-aided surgery [70]. In the following, we perform the dynamic analysis of a robotic catheter used in vascular surgery. In this context, dynamics plays an essential role in planning shaping operations and simulating the overall 3D motion in a constrained environment.

The MagellanTM Robotic Catheter 10Fr has been selected for the analysis ¹. This is a robotic catheter used for intravascular shaping operations, and it is composed by a guide and a robotically steerable inner leader. Both the guide and the leader have the possibility to bend, as we can see from Fig. 6.19. We consider the robotic catheter in the minimal extension configuration of the leader, and we model it using the PWCD model, with $n=2$. The schematic model that we used is illustrated in Fig. 6.20. The initial configuration is totally straight, and it is given by Table 6.18. The mass and stiffness matrices for the *guide* and the *leader* are given respectively by Table 6.19 and 6.20. The material is structural steel, and the cross-sections are circular crowns.

As illustrative example, we consider the following input torques applied on nodes 3 and 4 as

- $\tau_{3,x} = 0.4t; \quad \tau_{3,y} = 4t; \quad \tau_{3,z} = 40t, \quad 0 \leq t \leq 1s$
- $\tau_{4,x} = -0.15t; \quad \tau_{4,y} = -1.5t; \quad \tau_{4,z} = -15t, \quad 0 \leq t \leq 1s$

This results in out-of-plane motion of the robotic catheter, a typical motion in robotic steerable shaping operations, including both bendings and also torsion. A snapshot of the model in SimSOFT at the end-configuration in the test example is illustrated in Fig. 6.21. Figure 6.22 shows the recording of the tip displacements for the node 3 and the node 4 of the robotic catheter during the simulation.

node	x	y	z
1	0	0	0
2	0.442	0	0
3	0.500	0	0
4	0.530	0	0

Table 6.18: Initial configuration of the Magellan Robotic Catheter 10Fr

¹<http://www.hansenmedical.com>

element	M_{11}	M_{22}	M_{33}	M_{44}	M_{55}	M_{66}
guide	ρA_G	ρA_G	ρA_G	ρJ_G	ρI_G	ρI_G
leader	ρA_L	ρA_L	ρA_L	ρJ_L	ρI_L	ρI_L

Table 6.19: Cross-section mass matrices for the elements of the Magellan Robotic Catheter 10Fr

element	K_{11}	K_{22}	K_{33}	K_{44}	K_{55}	K_{66}
guide	EA_G	GA_G	GA_G	GJ_G	EI_G	EI_G
leader	EA_L	GA_L	GA_L	GJ_G	EI_L	EI_L

Table 6.20: Cross-section stiffness matrices for the elements of the Magellan Robotic Catheter 10Fr.

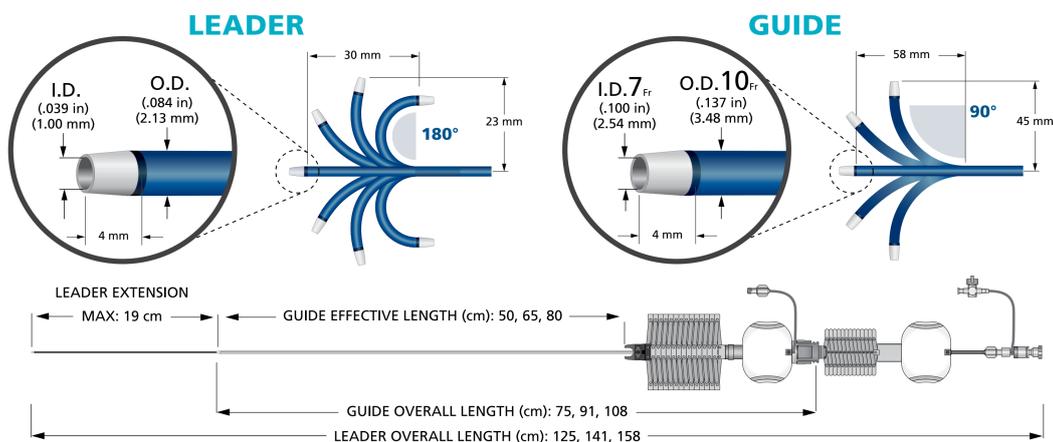


Figure 6.19: Geometric data for the Magellan Robotic Catheter 10Fr



Figure 6.20: Schematic model (not in scale) of the the Magellan Robotic Catheter 10Fr in the configuration of minimum extension for the leader



Figure 6.21: The Magellan Robotic Catheter 10Fr in the configuration of minimum extension for the leader, as it appears in SimSOFT

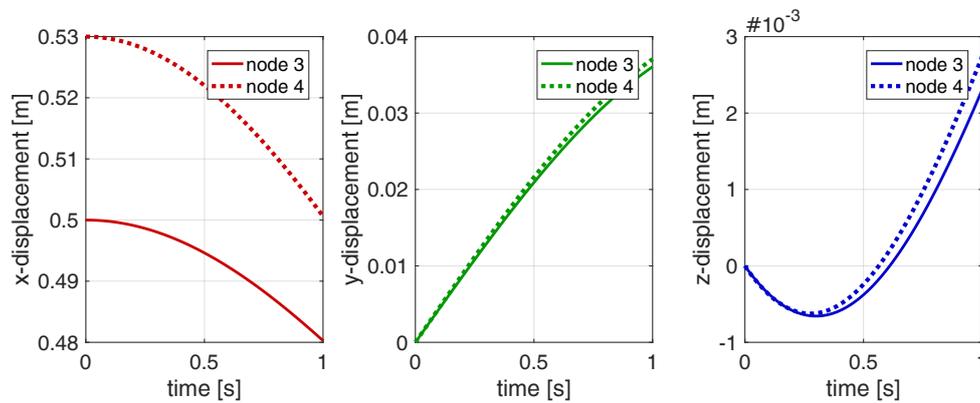


Figure 6.22: Displacements of the Magellan Robotic Catheter 10Fr in the test simulation. As we can see, the trajectory induces a three-dimensional out-of-plane motion.

Rehabilitation

The last dynamic example is the modeling of soft bending actuators used for whole-body manipulation. In the recent years, a growing interest on the use of such actuators can be found in the development of soft orthotics and/or prosthetics for rehabilitation purposes [71, 72]. In this context, dynamics using a fast 1D FEA plays an essential role in the first analysis on the manipulator, when we are interested in the simulation of the overall motion of the actuator. A typical circumstance could be the one involved in robust design optimization, i.e., how changing the design parameters regarding the topology and the geometry of the actuator, could affect the performances of the system. In this case, indeed, a fast 1D FEA-based physics engine might predict the resulting motion in a relatively short time, if compared to 3D FEA softwares. After, once defined the overall parameters of the actuator, one could, eventually, optimize the internal design of the actuator's chambre through 3D FEA softwares as Comsol or Abaqus.

An illustrative example of such actuators is shown in Fig. 6.23. We model the soft bending actuator as a CD soft arm, with the following features: length $L = 0.15m$, elastomeric material with $E = 110e3Pa$, $\nu = 0.5$, $G = E/(2(1 + \nu))Pa$, $\rho = 1080kg/m^3$. The initial configuration is straight, with its main axis in the x -direction. The objective of this example is to show the capability of SimSOFT in performing dynamic analysis in presence of hyper-elastic materials, as the elastomer. These actuators are typically pneumatically actuated. When the air flows in, the actuator acts as a balloon, and it tries to expand in all directions. For obtaining a bending moment when inflated, usually a tiny sheet of inextensible material is added on one side. Since the distributed pressure loads acts, experimentally, as a tip downward force of a certain magnitude [73], as illustrative example we apply a tip force on the end node with law $f_y(t) = -0.01t$, $0 \leq t \leq 1s$. This results in plane motions which are typically used for hand rehabilitation. Figure 6.24 illustrate some captions of the simulation, while Fig. 6.25 shows the recording of the end-node trajectory along the x and y directions. The system was simulated without gravity.



Figure 6.23: Example of soft bending actuators. Picture from the soft robotics toolkit^{a=}

^{a=} <https://softroboticstoolkit.com/>

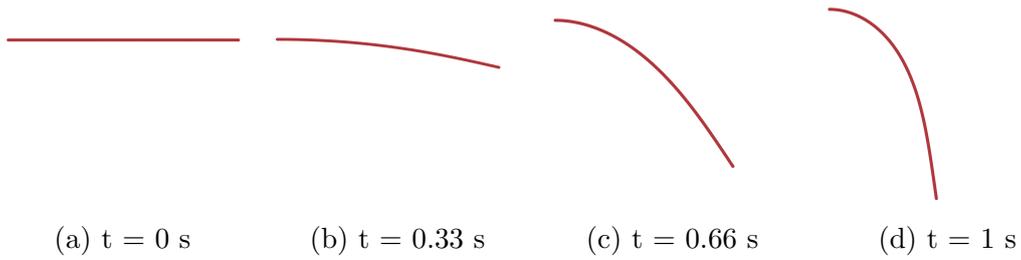


Figure 6.24: Snapshot of the soft bending actuator in SimSOFT.

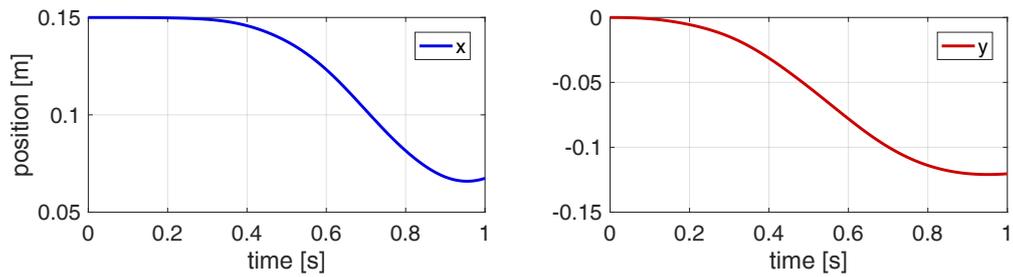


Figure 6.25: Tip displacements of the soft bending actuators along the test trajectory

6.3.2 Learning from simulation

One of the greatest advantages of this formulation is its ability to catch the main physical behaviour of soft robotics systems in a computational efficient way, thanks to the use of the geometric implicit time integration scheme formulated on a Lie group. SimSOFT has been implemented on a Intel® Core™ i7-4910MQ CPU (quad-core 2.50 GHz, Turbo 3.50 GHz), 32 Gb RAM 1600MHz DDR3L, NVIDIA®Quadro®K2100M w/2GB GDDR5 VGA machine, running Ubuntu 14, 64 bits. The mean computational time for solving the equations of motion of soft model, in a test case of 1 s of simulation, is 2 s if using $h = 10^{-2}$ s and 20 s if using $h = 10^{-3}$ s. This is quite a good result if compared to other formulations, which require an average time of 28 min for 1 s of simulation [74].

This advantage allows to use SimSOFT as *dataset generator* for developing even faster mathematical models for soft robots, using methods from the computational intelligence community, as deep learning [75]

Even if the development of a full learned soft robotic simulator is hugely expensive from the computational point of view for both the dataset generation in all the possible situations and the training procedure, we think that, in a first phase, we can just develop fast models for specific and already designed robotic structures. A typical situation could be: the use a computational mechanics based simulation as SimSOFT in the mechanical design and analysis, and the development of ad-hoc light mathematical models based on deep learning techniques and a reliable dataset generator, once that the design has been fixed, the system is fully defined and the operational range in which the robot operates is known.

The methodology used in this section for developing light models for soft robots comprises four steps as (1) dataset generation; (2) training process; (3) model generation; (4) model validation.

We illustrate this idea using two fully defined benchmark problems, respectively presented in Sec. 3.2.5 and 4.2.3. We are currently working in other more complex examples, as we are in progress of the development of the *soft robotics unina dataset*.

Static benchmark

The static benchmark refers to the example discussed in Sec. 3.2.5.

parameter	values
τ	1:1:10 N m
EI	1:1:10 Nm ²
L	0:0.01:0.1 and 0.2:0.1:1 m.

Table 6.21: Dataset for the constant deformation soft arm.

Table 6.21 gives the dataset used for this example. Notice that the dataset was tightened near the origin to catch up the difficult orizontal tangent behavior next

to the constraint for the cantilevered arm. The simulations for generating the dataset have been performed using SimSOFT. We used a two-layer feedforward neural network with 100 hidden neurons, and we updated its weights and biases using the Bayesian regularization backpropagation algorithm [76, 77]. The training process lasted 75 s. The regression value R and the mean square error MSE of the net result being $R = 1$ and $MSE = 9e - 7$. We generate a learned model and we compare its performances by using as error measurement the euclidean distance between the exact and the learned solution normalized along the arm length as

$$e(\alpha)[\%] = \sqrt{\sum_{i=1}^3 e_i^2} \quad (6.1)$$

where

$$e_i(\alpha) = \frac{p_{i,l}(\alpha) - p_{i,e}(\alpha)}{\alpha} \cdot 100 \quad (6.2)$$

being p_{i_l} and p_{i_e} respectively the learned and the exact displacements along the three linear directions and $\alpha \in [0, 1]$ the curvilinear abscissa along the arm length. In the central part of the training dataset, i.e., $\tau = 5 \text{ N m}$; $EI = 5 \text{ Nm}^2$, and using $L = 1 \text{ m}$, we obtain

$$e_m \pm e_\sigma[\%] = 0.16 \pm 0.14 \quad (6.3)$$

where e_m and e_σ indicate respectively the mean and the standard deviation of the error e . The model takes an average execution time of 6.7 ms to run, while the reference exact models (analytical and numerical) require respectively the average execution time of 0.2 ms and 40 ms.

The statical learned model was validated by randomly varying one of its parameters (in this case, the applied torque τ) within the range of the dataset. Figure 6.26 shows the errors of the learned model in nine different configurations. In all these cases, the average errors are below 1%.

Dynamic benchmark

The dynamic benchmark refers to the example discussed in Sec. 4.2.3.

parameter	values
ω	$\pi:5\pi \text{ rad s}^{-1}$
ρ	$1000:500:5000 \text{ kg m}^3$
E	$10 \times 10^6:10 \times 10^6:10 \times 10^7 \text{ kPa}$
L	$0.1:0.1:1 \text{ m}$
t	$0.1:0.01:1 \text{ s}$

Table 6.22: Dataset for the free-rotating soft arm.

Table 6.22 gives the dataset used for this example. The simulations for generating the dataset have been performed using SimSOFT. We used again a two-layer feedforward neural network with 100 hidden neurons in the fitting network's hidden

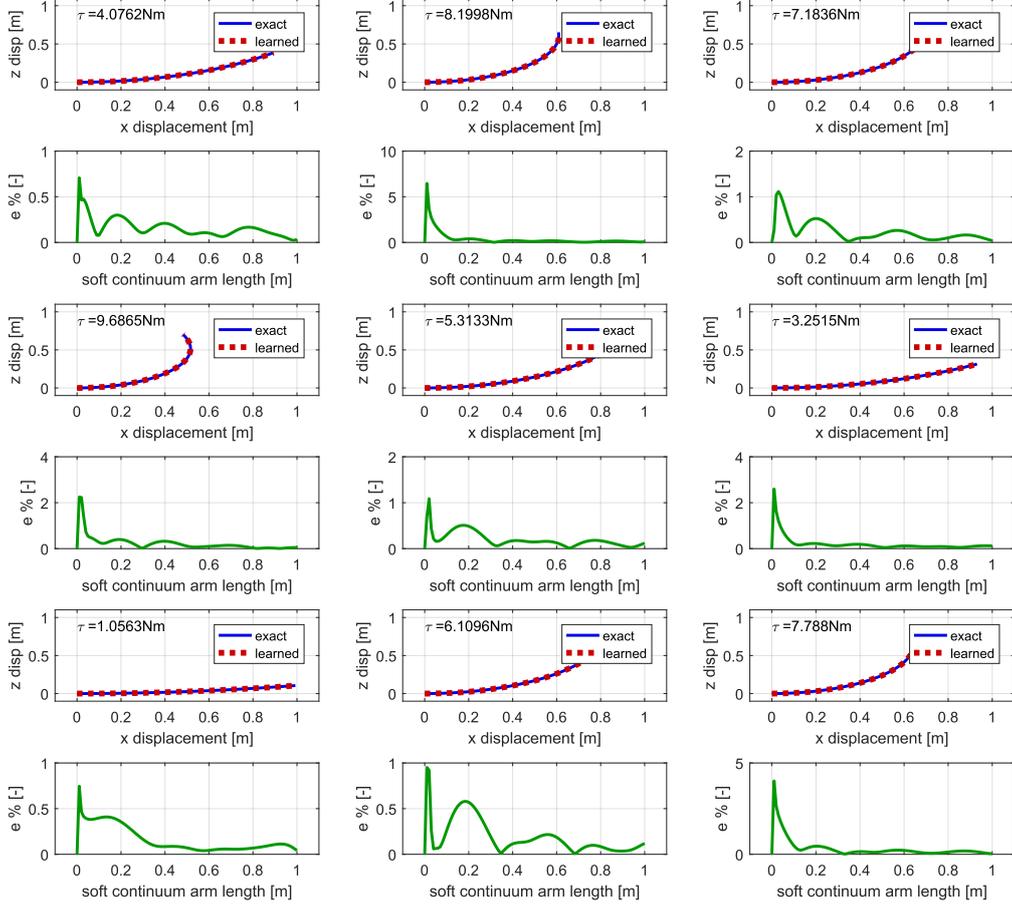


Figure 6.26: Learned continuum arm model at different load conditions.

layer and the Bayesian regularization backpropagation algorithm for the training process. The training time lasted $2.14e4$ s. The regression value R and the mean square error MSE of the net result being $R = 1$ and $MSE = 1e - 6$. The value for $e_i(\alpha)$ in this case is given by $e_i(\alpha) = (p_{i,l}(\alpha) - p_{i,e}(\alpha)) \cdot 100$.

In the central part of the training dataset, i.e., $\omega = 2\pi \text{ rad s}^{-1}$; $\rho = 2000 \text{ kg m}^3$, $E = 50 \times 10^6 \text{ kPa}$, $t = 0.5 \text{ s}$ and using $L = 1 \text{ m}$, we obtain

$$e_m \pm e_\sigma[\%] = 0.065 \pm 0.041 \quad (6.4)$$

The model takes an average execution time of 20 ms to run, while the reference exact models (analytical and numerical) require respectively the average execution time of 1 ms and 2 s.

The optimal learned model was validated by randomly varying one of its parameters (in this case, the applied velocity ω) within the range of the dataset.

Figure 6.27 shows the errors of the learned model in four different configurations. In all these cases, the average errors are below 1%.

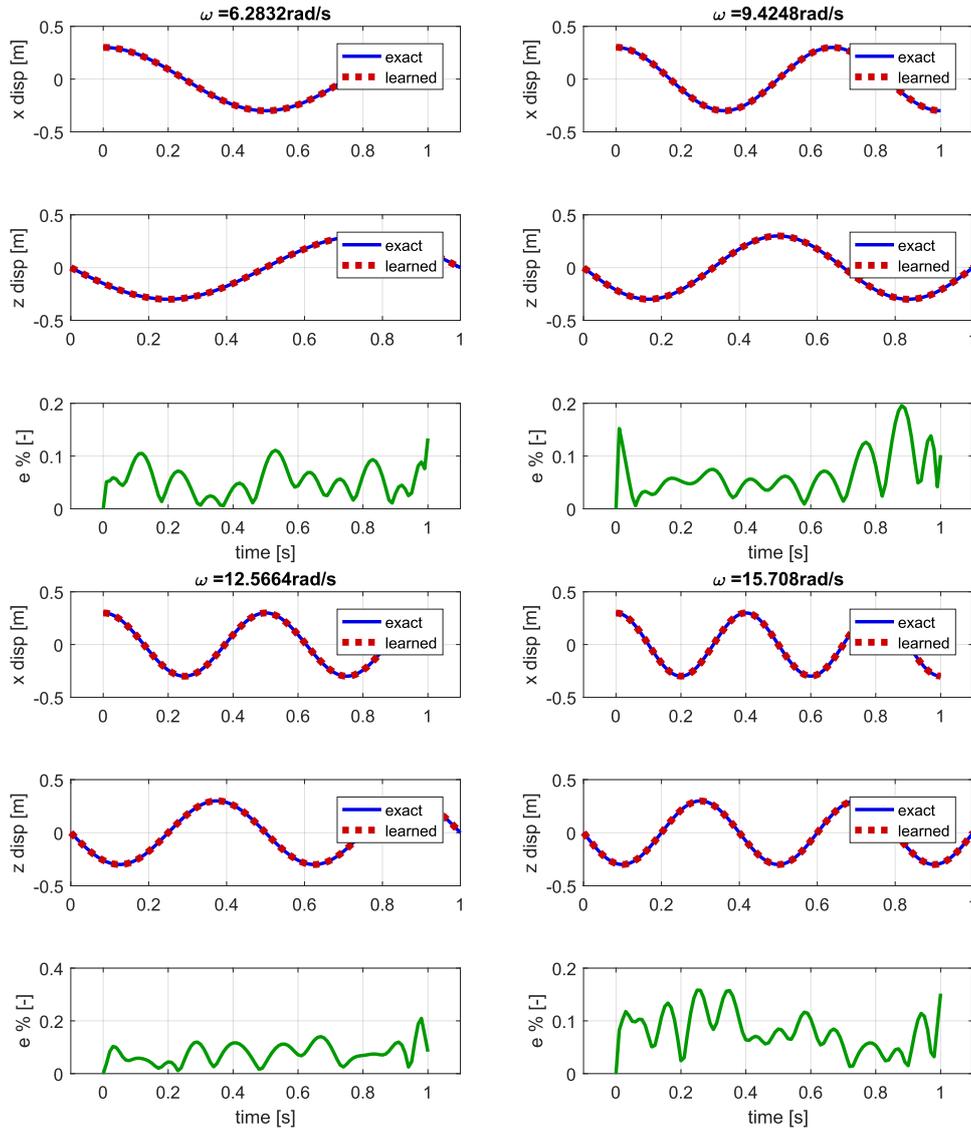


Figure 6.27: Learned continuum arm model at different rotating velocities.

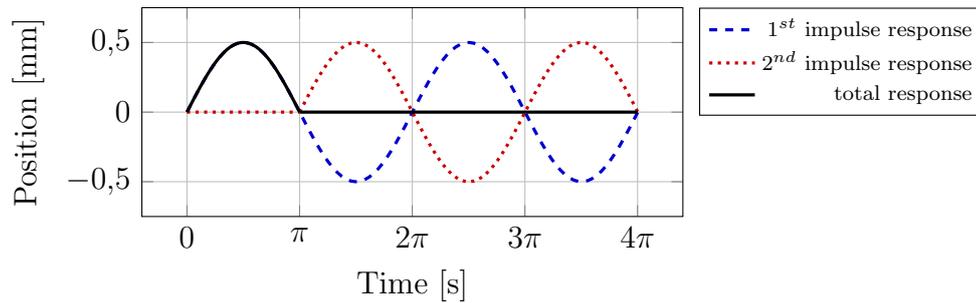


Figure 6.28: Zero vibration input shaping process

6.3.3 Vibration control for remote procedures in challenging environments

Vibration control has the objective to limit the transient and residual vibrations which might arise in complex motion of nonlinear mechanical systems. One of the most adopted vibration control strategy is the command–shaping method [78]: shaping the reference command such that the vibratory modes of the system are canceled reduces the vibrations. In these methods, the reference signal is convolved with a sequence of impulses, namely an input shaper (see, e.g. Fig. 6.28). The rise time of the command is lengthened by the duration of the shaper. Hence, to achieve high–speed motion, it is important to minimize the shaper duration. The main advantage of input shaping techniques is that the timing and the amplitude of the impulses are determined by solving a set of constraint equations using only estimates of the system natural frequencies and damping [78].

However, shapers can be sensitive to parameter estimation, *i.e.* if the frequencies and the damping are not well estimated or they are time–varying, then the effectiveness of the method is reduced. For some shapers, residual vibration increases rapidly as the actual parameters deviate from the modeled or estimated parameters. One possible strategy to overcome this problem is the use of robust input shapers [79], which allow a significative vibration suppression even in presence of uncertainty in the plant parameters. The disadvantage in this case is increasing the rise time, which becomes higher as the robustness of the shaper increases. Moreover, robust shapers still suffer in systems with time–varying parameters, since they allow the suppression of the vibrations in a pre–defined frequency interval. In sum, the use of robust shapers, even if it allows for vibration suppression in presence of uncertainty, it still presents two main problems: *(i)* increasing of the rise time; *(ii)* requiring a–priori knowledge of the frequency range in which the system operates. Moreover, input shapers have been mainly used in the context of linear systems, or, using linearized models of nonlinear systems, which allows to determine an estimation of the natural frequencies of the original nonlinear system and, based on these, compute time location and amplitudes of the shaper.

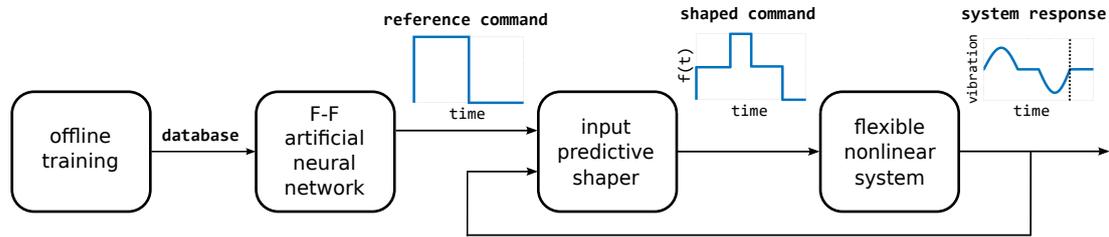


Figure 6.29: Input predictive shaping process.

Since the basic idea of the input shapers is to apply input commands when the oscillatory mechanical system crosses the zero deflection on a time-position graph, the availability of efficient physics engine to simulate the motion of flexible nonlinear mechanical systems in different operative conditions, might leverage the development of a new approach in designing input shapers, using a predictive approach. The basic idea behind the Input Predictive Shaping (IPS) process is illustrated in Fig. 6.29.

Zero vibration input predictive shaping for a nonlinear pendulum

As illustrative example of the method, we develop the zero-vibration input predictive shaping (ZV-IPS) [80] for the most well-known nonlinear mechanical system, namely a nonlinear pendulum, illustrated in Fig. 6.30. This system can approximate the behavior of many real mechanical systems, as an overhead crane with a suspended payload. In the offline training, we simulate the motion of the system with the starting conditions indicated in Table 6.23, where l and \dot{x} indicate respectively the hoist length and the velocity of the trolley along the x -direction. These conditions make the method valid for most industrial cranes. After, we train a 10 hidden neurons feedforward network which updates weights according to Levenberg-Marquardt optimization algorithm [81]. The analytic expressions for the *ZV-IPS for a nonlinear pendulum* are given in Table 6.24.

We test the vibration control algorithm in the simple test trajectory illustrated in Fig. 6.32. The ideal reference velocity command in x -direction is a step function of duration 3.315 s and value $2 \times 10^{-1} \text{ ms}^{-1}$, which makes the trolley moving of 1.3 m. As shown in Fig. 6.32, the real reference velocity command acting on the experimental crane results having a trapezoidal profile, of total duration 3.723 s, since we deal with actual commands. The real command shaped using the developed ZV-IPS is shown in the same figure: the result is a total duration of 4.59 s. All these data refer to the performed experiments. In the ZV-IPS command, we can note that the durations of the two commands acting at the velocity $1 \times 10^{-1} \text{ ms}^{-1}$ are different: the first one of 0.36 s, while the second one of 1.02 s. This happens because the z -coordinate corresponding to the initial crane hoist is smaller than the z -coordinate corresponding to the final crane hoist. In fact, as the frequency decreases, the rise time of the command increases. The movement in z -direction is of 0.8 m, with constant velocity.

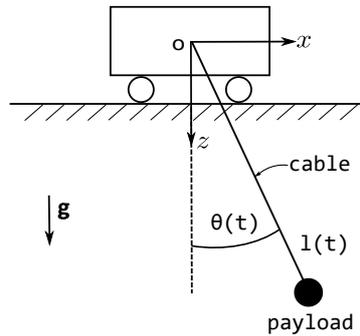


Figure 6.30: Nonlinear pendulum.

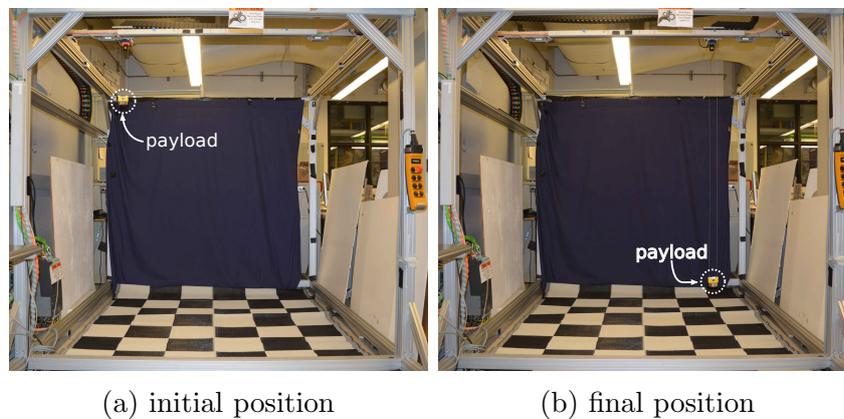


Figure 6.31: Experimental setup, the Georgia Tech bridge crane

Figure 6.33 compares the payload deflections of the ZV-IPS command with the unshaped response, in both simulations and experiments performed on the Georgia Tech bridge crane, which is shown in Fig. 6.31

Moreover, we compared the performance of the ZV-IPS in terms of both shaper duration and residual vibration amplitude with respect to the classical ZV shaper and the robust two-hump extra insensitive (EI) shaper. Experiments were recorded for the same trajectory. The comparison is illustrated in Fig. 6.34. ZV-IPS shows the same residual vibrations of the two-hump EI shaper, but with a shaper duration similar to the ZV, the fastest non-negative shaper. Thus, the input predictive shaper, using only two informations from the cranes states (hoist cable length and trolley velocity), measured on the fly, is a good compromise in obtaining almost zero residual vibrations having at the same time a fast response.

parameter	values
l	$0.1:0.1:25\pi$ m
\dot{x}	$-2 \times 10^{-1}:0.1:2 \times 10^{-1}$ m s ⁻¹

Table 6.23: Dataset for the ZV-IPS for a nonlinear pendulum.

$$A_1 = \frac{2.065}{e^{-0.6987i+0.9542l+1.300} + 1} - \frac{5.306}{e^{-0.3422i-0.7200l-3.123} + 1} - \frac{9.254}{e^{-0.4359i+0.8196l+2.543} + 1} + \frac{1.355}{e^{0.2482i+0.0055l+2.524} + 1} + 5.709 \quad (6.5)$$

$$A_2 = 1 - A_1 \quad (6.6)$$

$$t_1 = 0 \quad (6.7)$$

$$t_2 = \frac{2.438}{e^{0.0979i-0.3130l-0.1394} + 1} + \frac{20.62}{e^{0.0992i-5.124l-4.502} + 1} - \frac{5.566}{e^{-0.1764i+0.0964l-1.143} + 1} + \frac{32.78}{e^{0.0290i-0.9456l-4.046} + 1} - 49.55 \quad (6.8)$$

Table 6.24: ZV-IPS for a nonlinear pendulum

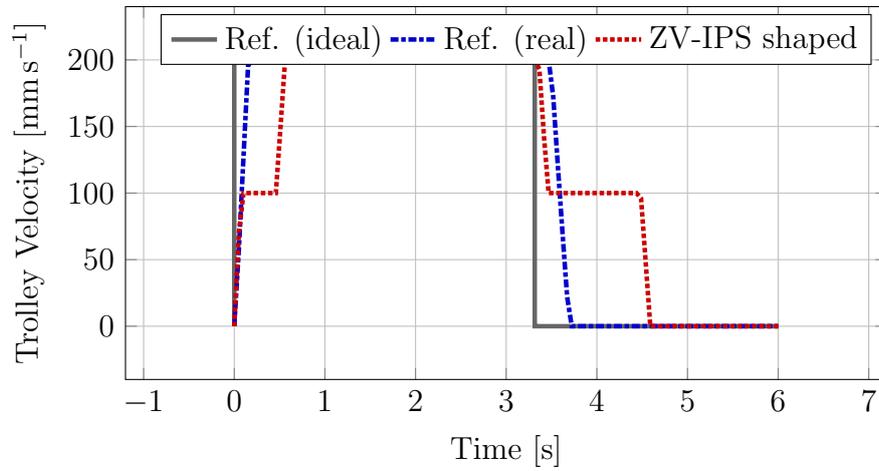


Figure 6.32: Velocity commands for the ZV-IPS experiments.

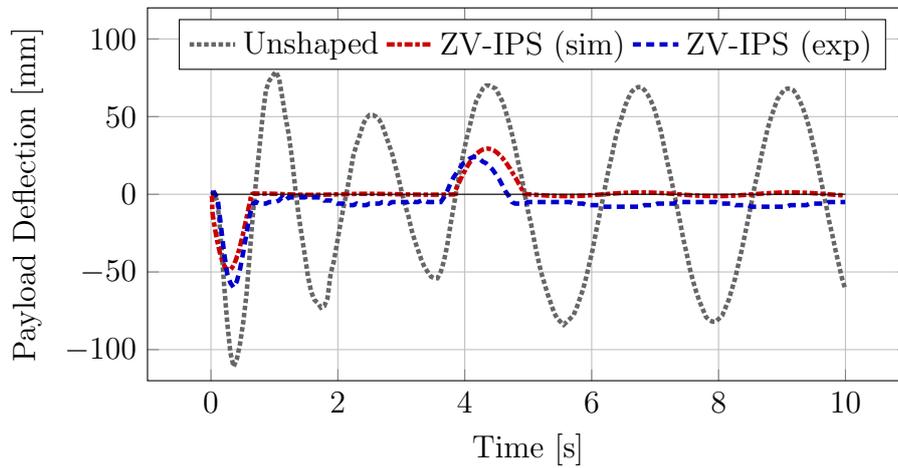


Figure 6.33: ZV-IPS: comparison of experimental and simulated response.

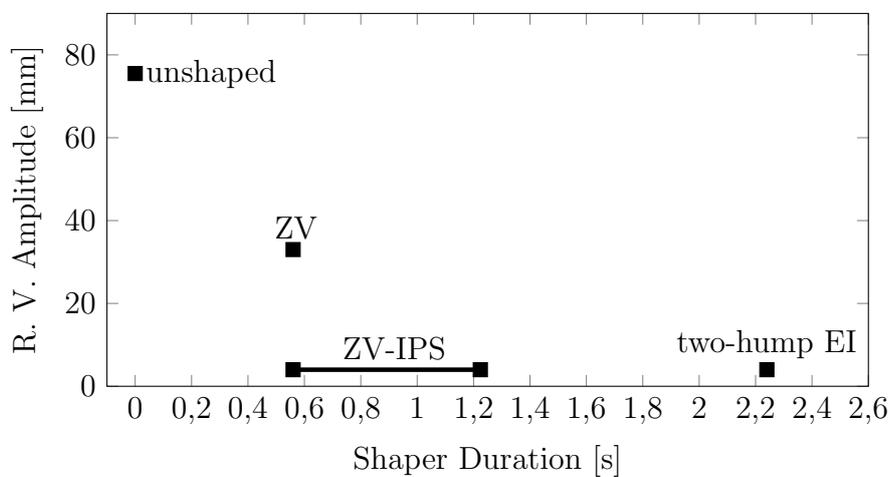


Figure 6.34: Comparison of input shapers. Experimental data.

6.4 Summary

- A general-purpose physics engine for rigid and soft robots is composed by a library of elements which are connected by joints. The dynamics of the elements (rigid or soft arms) results in ODE, while the dynamics of elements connected by joints results in DAE. These equations are then solved using an implicit time integration scheme, and some tools are available for post-processing analysis.
- Benchmarks including (i) planar two-link rigid manipulator; (ii) free-falling flexible double pendulum; (iii) spatial flexible manipulator; (iv) Princeton experiment; (v) multisegment rod-driven continuum manipulator, have been selected from the literature to validate the mathematical models discussed in the thesis.
- Dynamic models are used to simulate the motion, mechanical design and optimization, planning operations for robotic mechanisms. In the nuclear fusion scenario, the dynamic analysis leads to some data useful, for instance, for the mechanical design of the VTS and for validating conceptual design alternatives. In robotic surgery, the piecewise constant deformation dynamic model might be used to predict the displacements of the backbone of the manipulator, and thus, planning surgical operations in confined environment. In rehabilitation, the constant deformation dynamic model might be used in a high-level analysis, to size the soft actuator such that it will assume the desired shape for whole-body manipulation tasks.
- The availability of a fast and accurate physics engine for complex robotic mechanisms might pave the way to develop computationally efficient mathematical models using a deep learning approach.
- The possibility to predict the motion, and thus, the vibrations, of generic mechanical systems, might pave the way towards the development of predictive approaches for control, as in vibration control.
- The zero-vibration input predictive shaper offers a promising alternative to produce, for a nonlinear pendulum, fast motion without vibrations, as compared to other input shapers.
- ... Other potential applications which eventually would require the availability of a computational mechanics based engine for rigid and soft robots?

Conclusion

Summary

The main results of this thesis which go beyond the current state-of-the art in the robotics research are

- generalization of the *product of exponentials* formula for soft robotic manipulators using geometric methods and a finite element approach
- derivation of *soft geometric jacobians* for soft robotic arms using geometric methods and a finite element approach
- derivation of a *dynamic model* for soft robotic manipulators which has the same structure of the rigid robotic manipulators, using the Hamiltonian formulation and the calculus of variations.
- *dynamic coupling* of multiple soft and rigid arms in the same mathematical model using geometric methods for the formulation of rigid and flexible joints.

This allowed the development of SimSOFT, a c++ dynamic library which paves the way towards the development of interesting applications, as the ones presented as examples in the thesis.

Further work

Several points of the thesis can be further investigated. Some possibilities are

- development of closed-loop inverse kinematics algorithms for piecewise constant deformation soft robotic arms
- development of a robot control framework for piecewise constant deformation soft robotic arms
- development of a theoretical framework based on the geometric methods for whole-body grasping and manipulation using piecewise constant deformation soft robotic arms

The parallel development of further applications in SimSOFT can be addressed.

Appendix A

Lie group framework

A.1 Lie group fundamentals

Definition 2 A group (G, \cdot) is a set G of elements q with a composition operation (\cdot) which satisfies the axioms of closure, associativity, existence of the neutral and of the inverse elements.

Definition 3 A Lie group is a continuous group for which the composition rule and the inverse are smooth.

Thus, geometrically speaking, a Lie group is a differentiable manifold, and differential geometry can be used to perform operations on the group. In this context we deal with matrix Lie group, that is, a finite dimensional Lie group for which a matrix representation with the matrix product as composition rule is always possible.

The tangent space at $q \in G$ is denoted $T_q G$.

Definition 4 The Lie algebra \mathfrak{g} is the tangent space at the identity of a Lie group. It is isomorphic to \mathbb{R}^k through the invertible linear map

$$\widetilde{(\cdot)} : \mathbb{R}^k \rightarrow \mathfrak{g}, \quad \mathbf{a} \mapsto \widetilde{\mathbf{a}} \quad (\text{A.1})$$

The derivative of q with respect to $a \in \mathbb{R}$ reads

$$d_a(q) = q\widetilde{\mathbf{a}}_L \quad (\text{A.2})$$

$$= \widetilde{\mathbf{a}}_R q \quad (\text{A.3})$$

where $\widetilde{\mathbf{a}}_L \in \mathfrak{g}$ and $\widetilde{\mathbf{a}}_R \in \mathfrak{g}$ are respectively called a left and right invariant vector field.

The adjoint representation of a Lie algebra element is defined as

$$\text{Ad}_q : \mathfrak{g} \rightarrow \mathfrak{g}, \quad \widetilde{\mathbf{a}} \mapsto q\widetilde{\mathbf{a}}q^{-1} \quad (\text{A.4})$$

According to the isomorphism $\mathfrak{g} \simeq \mathbb{R}^k$, Ad_q is also used, with a slight abuse of notations, for the adjoint representation as a linear map acting \mathbb{R}^k isomorphic

elements of the Lie algebra, that is $\text{Ad}_q : \mathbb{R}^k \rightarrow \mathbb{R}^k$. Some properties of the adjoint representation:

$$(\text{Ad}_q)^{-1} = \text{Ad}_{q^{-1}} \quad (\text{A.5})$$

$$\text{Ad}_{q_1} \text{Ad}_{q_2} = \text{Ad}_{q_1 q_2} \quad (\text{A.6})$$

The Lie bracket operator is the bilinear operator defined as

$$[\cdot, \cdot] : \mathfrak{g} \times \mathfrak{g} \rightarrow \mathfrak{g}, \quad [\tilde{\mathbf{a}}, \tilde{\mathbf{b}}] \mapsto \tilde{\mathbf{a}}\tilde{\mathbf{b}} - \tilde{\mathbf{b}}\tilde{\mathbf{a}} \quad (\text{A.7})$$

Thanks to the Lie bracket operator, we can write the commutativity of the cross derivatives as

$$d_b(\tilde{\mathbf{a}}) - d_a(\tilde{\mathbf{b}}) = [\tilde{\mathbf{a}}, \tilde{\mathbf{b}}] \quad (\text{A.8})$$

According to the isomorphism $\mathfrak{g} \simeq \mathbb{R}^k$, Eq. A.8 can be expressed in terms of vectors in \mathbb{R}^k as

$$d_b(\mathbf{a}) - d_a(\mathbf{b}) = \hat{\mathbf{a}}\mathbf{b} \quad (\text{A.9})$$

where $\hat{\cdot}$ is a linear operator defined as

$$\widehat{(\cdot)} : \mathbb{R}^k \rightarrow \mathbb{R}^{k \times k}, \quad \mathbf{a} \mapsto \hat{\mathbf{a}} = \mathbf{A} \quad (\text{A.10})$$

The linear operator $\hat{\cdot}$ can be alternatively introduced from the adjoint representation and denoted as

$$\text{ad}_{\mathbf{a}}(\mathbf{b}) = \hat{\mathbf{a}}\mathbf{b} \quad (\text{A.11})$$

A.2 The special Orthogonal group $SO(3)$

Proposition 3 *The set of 3×3 matrices as*

$$\{\mathbf{R} \in \mathbb{R}^3 | \mathbf{R}^T \mathbf{R} = \mathbf{I}_{3 \times 3}, \det(\mathbf{R}) = +1\} \quad (\text{A.12})$$

together with the matrix product as composition operation, is a matrix Lie group, which is called the special Orthogonal group $SO(3)$.

Notice that $SO(3)$ is not commutative, i.e., $\mathbf{R}_1 \mathbf{R}_2 \neq \mathbf{R}_2 \mathbf{R}_1$, since the matrix product is not commutative.

Its Lie algebra, denoted as $\mathfrak{so}(3)$, is the space of skew-symmetric matrices

$$\tilde{\mathbf{a}} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \quad (\text{A.13})$$

It is isomorphic to \mathbb{R}^3 with

$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \quad (\text{A.14})$$

In particular, Eq. A.2 and Eq. A.3 read

$$d_a(\mathbf{R}) = \mathbf{R}\tilde{\mathbf{a}}_L \quad (\text{A.15})$$

$$= \tilde{\mathbf{a}}_R \mathbf{R} \quad (\text{A.16})$$

The adjoint representation in A.4 is given by

$$\text{Ad}_R(\tilde{\mathbf{a}}) = \mathbf{R}\tilde{\mathbf{a}}\tilde{\mathbf{R}}^T \quad (\text{A.17})$$

$$\text{Ad}_R(\mathbf{a}) = \mathbf{R}\mathbf{a} \quad (\text{A.18})$$

The $\hat{\cdot}$ is equivalent to $\tilde{\cdot}$.

A.3 The special Euclidean group $SE(3)$

Proposition 4 *The set of 4×4 matrices as*

$$\mathbf{H} = \mathcal{H}(\mathbf{R}, \mathbf{u}) = \begin{bmatrix} \mathbf{R} & \mathbf{u} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \quad (\text{A.19})$$

where $\mathbf{R} \in SO(3)$ and $\mathbf{u} \in \mathbb{R}^3$, together with the matrix product as composition rule, is a matrix Lie group, and is called the special Euclidean group $SE(3)$.

The Lie algebra, denoted $\mathfrak{se}(3)$, is the space of 4×4 matrices $\tilde{\mathbf{h}}$ as

$$\tilde{\mathbf{h}} = \mathcal{A}(\tilde{\mathbf{h}}_\omega, \mathbf{h}_u) = \begin{bmatrix} \tilde{\mathbf{h}}_\omega & \mathbf{h}_u \\ \mathbf{0}_{1 \times 3} & 0 \end{bmatrix} \quad (\text{A.20})$$

where $\tilde{\mathbf{h}}_\omega \in \mathfrak{so}(3)$ and $\mathbf{h}_u \in \mathbb{R}^3$. In the screw theory, the elements of $\mathfrak{se}(3)$ are called twists. The Lie algebra $\tilde{\mathbf{h}}_\omega \in \mathfrak{so}(3)$ is the skew-symmetric matrix

$$\tilde{\mathbf{h}}_\omega = \begin{bmatrix} 0 & -h_3 & h_2 \\ h_3 & 0 & -h_1 \\ -h_2 & h_1 & 0 \end{bmatrix} \quad (\text{A.21})$$

which is isomorphic to \mathbb{R}^3 with $\mathbf{h}_\omega = [h_1 \ h_2 \ h_3]^T$. Notice that $\mathfrak{se}(3)$ is isomorphic to \mathbb{R}^6 , with

$$\mathbf{h} = \begin{bmatrix} \mathbf{h}_u \\ \mathbf{h}_\omega \end{bmatrix} \quad (\text{A.22})$$

Using the left invariant vector field, the derivative of \mathbf{H} take the form

$$d_a(\mathbf{H}) = \mathbf{H}\tilde{\mathbf{h}} \quad (\text{A.23})$$

The adjoint representation is given by

$$\text{Ad}_{\mathbf{H}}(\tilde{\mathbf{h}}) = \mathbf{H}\tilde{\mathbf{h}}\mathbf{H}^{-1} \quad (\text{A.24})$$

$$\text{Ad}_{\mathbf{H}}(\mathbf{h}) = \begin{bmatrix} \mathbf{R} & \tilde{\mathbf{u}}\mathbf{R} \\ \mathbf{0}_{3 \times 3} & \mathbf{R} \end{bmatrix} \mathbf{h} \quad (\text{A.25})$$

The $\hat{\cdot}$ operator is given by

$$\hat{\mathbf{h}} = \begin{bmatrix} \tilde{\mathbf{h}}_\omega & \tilde{\mathbf{h}}_\omega \\ \mathbf{0}_{3 \times 3} & \tilde{\mathbf{h}}_\omega \end{bmatrix} \quad (\text{A.26})$$

Appendix B

The exponential map

In this section we introduce the exponential map obtained from the left invariant vector field of derivatives.

B.1 Exponential map fundamentals

Let us consider Eq. A.2 using the left invariant vector field as

$$d_a(q) = q\tilde{\mathbf{a}} \quad (\text{B.1})$$

Equation B.1 can be seen as a linear differential equation on the Lie group. If $\tilde{\mathbf{a}}$ does not depend on a , the solution is

$$q(a) = q_0 \exp(\tilde{\mathbf{a}}a) \quad (\text{B.2})$$

where q_0 is a constant of integration and \exp is the exponential operator, which maps an element of the Lie algebra into an element of the Lie group:

$$\exp : \mathfrak{g} \rightarrow G, \quad \tilde{\mathbf{a}} \mapsto \exp(\tilde{\mathbf{a}}) \quad (\text{B.3})$$

and it is given by

$$\exp(\tilde{\mathbf{a}}) = \sum_{i=0}^{\infty} \frac{\tilde{\mathbf{a}}^i}{i!} \quad (\text{B.4})$$

The inverse map of the exponential map is called the logarithmic map

$$\log : G \rightarrow \mathfrak{g}, \quad q \mapsto \log(q) = \tilde{\mathbf{a}} \quad (\text{B.5})$$

and it is given by

$$\log(q) = \sum_{i=0}^{\infty} \frac{(e - q)^i}{i} \quad (\text{B.6})$$

According to the isomorphism $\mathfrak{g} \simeq \mathbb{R}^k$, we can use $q = \exp(\mathbf{a})$ and $\log(q) = \mathbf{a}$, with $\mathbf{a} \in \mathbb{R}^k$. Since the Lie algebra is isomorphic to \mathbb{R}^k , the exponential map and

the logarithmic map introduce a local parametrization of the Lie group around any $q_0 \in G$. Thus, any $q \in G$ can be expressed as a function of $\tilde{\mathbf{u}} \in \mathfrak{g}$ using the exponential operator and the composition with q_0 as

$$q = q_0 \exp(\tilde{\mathbf{u}}) \quad (\text{B.7})$$

Considering B.7 and B.2, we can introduce the tangent operator of the exponential map

$$\mathbf{T} : \mathbb{R}^k \rightarrow \mathbb{R}^k, \quad \mathbf{u} \mapsto \mathbf{T}(\mathbf{u})d_a(\mathbf{u}) = \mathbf{a} \quad (\text{B.8})$$

which is given by

$$\mathbf{T}(\mathbf{u}) = \sum_{i=0}^{\infty} (-1)^i \frac{\hat{\mathbf{u}}^i}{(i+1)!} \quad (\text{B.9})$$

Expressing B.7 as $\tilde{\mathbf{u}} = \log(q_0^{-1}q)$, we can introduce the inverse of the tangent operator of the exponential map

$$\mathbf{T}^{-1} : \mathbb{R}^k \rightarrow \mathbb{R}^k, \quad \mathbf{u}, \mathbf{a} \mapsto \mathbf{T}^{-1}(\mathbf{u})\mathbf{a} = d_a(\mathbf{u}) \quad (\text{B.10})$$

which is given by

$$\mathbf{T}^{-1}(\mathbf{u}) = \sum_{i=0}^{\infty} (-1)^i B_i \frac{\hat{\mathbf{u}}^i}{(i)!} \quad (\text{B.11})$$

where B_i is the Bernoulli number of the first kind.

B.2 The exponential map on $SO(3)$

The exponential map on $SO(3)$ is given by

$$\exp_{SO(3)}(\mathbf{h}_\omega) = \mathbf{I}_{3 \times 3} + \alpha(\mathbf{h}_\omega)\tilde{\mathbf{h}}_\omega + \frac{\beta(\mathbf{h}_\omega)}{2}\tilde{\mathbf{h}}_\omega^2 \quad (\text{B.12})$$

where

$$\alpha(\mathbf{h}_\omega) = \frac{\sin(\|\mathbf{h}_\omega\|)}{\|\mathbf{h}_\omega\|} \quad \beta(\mathbf{h}_\omega) = 2 \frac{1 - \cos(\|\mathbf{h}_\omega\|)}{\|\mathbf{h}_\omega\|^2} \quad \gamma(\mathbf{h}_\omega) = \frac{\|\mathbf{h}_\omega\|}{2} \cot\left(\frac{\|\mathbf{h}_\omega\|}{2}\right) \quad (\text{B.13})$$

Equation B.12 is known as Rodriguez' formula. The logarithmic map is given by

$$\log_{SO(3)}(\mathbf{R}) = \frac{\theta}{2\sin\theta}(\mathbf{R} - \mathbf{R}^T) \quad (\text{B.14})$$

with

$$\theta = \arccos\left(\frac{1}{2}(\text{trace}(\mathbf{R}) - 1)\right), \quad \theta < \pi \quad (\text{B.15})$$

The tangent operator is given by

$$\mathbf{T}_{SO(3)}(\mathbf{h}_\omega) = \mathbf{I}_{3 \times 3} - \frac{\beta(\mathbf{h}_\omega)}{2}\tilde{\mathbf{h}}_\omega + \frac{1 - \alpha(\mathbf{h}_\omega)}{\|\mathbf{h}_\omega\|^2}\tilde{\mathbf{h}}_\omega^2 \quad (\text{B.16})$$

The inverse of the tangent operator is given by

$$\mathbf{T}_{SO(3)}^{-1}(\mathbf{h}_\omega) = \mathbf{I}_{3 \times 3} + \frac{1}{2}\tilde{\mathbf{h}}_\omega + \frac{1 - \gamma(\mathbf{h}_\omega)}{\|\mathbf{h}_\omega\|^2}\tilde{\mathbf{h}}_\omega^2 \quad (\text{B.17})$$

B.3 The exponential map on $SE(3)$

The exponential map on $SE(3)$ is given by

$$\exp_{SE(3)}(\mathbf{h}) = \begin{bmatrix} \exp_{SO(3)}(\mathbf{h}_\omega) & \mathbf{T}_{SO(3)}^T(\mathbf{h}_\omega)\mathbf{h}_u \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \quad (\text{B.18})$$

The logarithmic map is given by

$$\log_{SE(3)}(\mathbf{H}) = \begin{bmatrix} \tilde{\mathbf{h}}_\omega & \mathbf{T}_{SO(3)}^{-T}(\mathbf{h}_\omega)\mathbf{h}_u \\ \mathbf{0}_{1 \times 3} & 0 \end{bmatrix} \quad (\text{B.19})$$

where $\tilde{\mathbf{h}}_\omega = \log_{SO(3)}(\mathbf{R})$ and

$$\log_{SO(3)}(\mathbf{R}) = \frac{\theta}{2\sin\theta}(\mathbf{R} - \mathbf{R}^T) \quad (\text{B.20})$$

with

$$\theta = \arccos\left(\frac{1}{2}(\text{trace}(\mathbf{R}) - 1)\right), \quad \theta < \pi \quad (\text{B.21})$$

The tangent operator is given by

$$\mathbf{T}_{SE(3)}(\mathbf{h}) = \begin{bmatrix} \mathbf{T}_{SO(3)}(\mathbf{h}_\omega) & \mathbf{T}_{u\omega+}(\mathbf{h}_u, \mathbf{h}_\omega) \\ \mathbf{0}_{3 \times 3} & \mathbf{T}_{SO(3)}(\mathbf{h}_\omega) \end{bmatrix} \quad (\text{B.22})$$

where

$$\mathbf{T}_{u\omega+}(\mathbf{h}_\omega, \mathbf{h}_u) = \frac{-\beta}{2}\tilde{\mathbf{h}}_\omega + \frac{1-\alpha}{\|\mathbf{h}_\omega\|^2}[\mathbf{h}_\omega, \mathbf{h}_u] + \frac{\mathbf{h}_u^T \mathbf{h}_\omega}{\|\mathbf{h}_u\|^2} \left((\beta - \alpha)\tilde{\mathbf{h}}_\omega + \left(\frac{\beta}{2} - \frac{3(1-\alpha)}{\|\mathbf{h}_u\|^2}\right)\tilde{\mathbf{h}}_\omega^2 \right) \quad (\text{B.23})$$

The inverse of the tangent operator is given by

$$\mathbf{T}_{SE(3)}^{-1}(\mathbf{h}) = \begin{bmatrix} \mathbf{T}_{SO(3)}^{-1}(\mathbf{h}_\omega) & \mathbf{T}_{u\omega-}(\mathbf{h}_u, \mathbf{h}_\omega) \\ \mathbf{0}_{3 \times 3} & \mathbf{T}_{SO(3)}^{-1}(\mathbf{h}_\omega) \end{bmatrix} \quad (\text{B.24})$$

where

$$\mathbf{T}_{u\omega-}(\mathbf{h}_\omega, \mathbf{h}_u) = \frac{1}{2}\tilde{\mathbf{h}}_\omega + \frac{1-\gamma}{\|\mathbf{h}_\omega\|^2}[\mathbf{h}_\omega, \mathbf{h}_u] + \frac{\mathbf{h}_u^T \mathbf{h}_\omega}{\|\mathbf{h}_u\|^4} \left(\left(\frac{1}{\beta} + \gamma - 2\right)\tilde{\mathbf{h}}_\omega^2 \right) \quad (\text{B.25})$$

with $\alpha = \alpha(\mathbf{h}_\omega)$, $\beta = \beta(\mathbf{h}_\omega)$, $\gamma = \gamma(\mathbf{h}_\omega)$

Appendix C

Lie subgroup framework

C.1 Lie subgroup fundamentals

Definition 5 (H, \cdot) is a subgroup of a group (G, \cdot) if H is a subset of G such that

- (i) H is closed under \cdot , i.e., $\forall q_1, q_2 \in H, q_1 \cdot q_2 = q_3 \in H$
- (ii) the neutral element of (G, \cdot) , e , is also the neutral element of (H, \cdot)
- (iii) the inverse of an element of H is also in H , i.e., $\forall q \in H, q^{-1} \in H$

Definition 6 A Lie subgroup is a subgroup of a Lie group which is also a differentiable manifold

We are only interested in matrix Lie groups, namely the matrix representation of finite dimensional Lie groups. In the same fashion, we speak of matrix Lie subgroup.

A Lie subgroup H is a Lie subgroup which is associated with a Lie subalgebra \mathfrak{h} . A Lie group and its Lie subgroups are associated as follows

Definition 7 A subalgebra \mathfrak{h} of an algebra \mathfrak{g} is a subspace of \mathfrak{g} which is closed under the Lie bracket operator, i.e., $\forall \tilde{\mathbf{a}}, \tilde{\mathbf{b}} \in \mathfrak{h}, [\tilde{\mathbf{a}}, \tilde{\mathbf{b}}] \in \mathfrak{h}$

Proposition 5 Let G and H be Lie groups, and \mathfrak{g} and \mathfrak{h} be the associated Lie algebra's. H is a Lie subgroup of G if and only if \mathfrak{h} is a subalgebra of \mathfrak{g}

Accordingly, \mathfrak{h} is isomorphic to \mathbb{R}^l , a subspace of \mathbf{R}^k , where k is the dimension of the Lie algebra \mathfrak{g} . In particular, for each subgroup, there exists a linear map

$$\mathfrak{h} \rightarrow \mathfrak{g}, \tilde{\mathbf{a}} \mapsto \tilde{\mathbf{A}}\mathbf{a} \tag{C.1}$$

$$\mathbb{R}^l \rightarrow \mathbb{R}^k, \mathbf{a} \mapsto \mathbf{A}\mathbf{a} \tag{C.2}$$

where \mathbf{A} is a $k \times l$ full rank matrix. Using this notation, the Lie bracket operator yields

$$\forall \tilde{\mathbf{a}}, \tilde{\mathbf{b}} \in \mathfrak{h}, \exists \tilde{\mathbf{c}} \in \mathfrak{h} : [\widetilde{\mathbf{Aa}}, \widetilde{\mathbf{Ab}}] = \widetilde{\mathbf{Ac}} \quad (\text{C.3})$$

and the derivative with respect to some parameter $a \in \mathbb{R}$ of a curve $q(a)$ in a Lie subgroup of G can be expressed as

$$d_a(q) = q\widetilde{\mathbf{Aa}} \quad (\text{C.4})$$

The commutativity of the derivatives reads

$$\mathbf{Ad}_b(\tilde{\mathbf{a}}) - \mathbf{Ad}_a(\tilde{\mathbf{b}}) = [\widetilde{\mathbf{Aa}}, \widetilde{\mathbf{Ab}}] \quad (\text{C.5})$$

and

$$\mathbf{Ad}_b(\mathbf{a}) - \mathbf{Ad}_a(\mathbf{b}) = \widehat{\mathbf{AaAb}} \quad (\text{C.6})$$

C.2 Subgroups of the special Orthogonal group $SO(3)$

The only non-trivial Lie subgroup of $SO(3)$ is $SO(2)$, which represents planar rotations, i.e., rotations about a given axis. In particular, the product of two rotations about a given axis results in a rotation about that given axis.

The dimension of the Lie algebra associated to $SO(2)$ is 1, $\mathbf{a} \in \mathbb{R}$ is the derivative of the rotation angle, and \mathbf{A} in Eq. C.2 becomes a 3×1 vector, namely the unit vector parallel to the rotation axis, say $\mathbf{n} \in \mathbb{R}^3$. Denoting two elements of the Lie algebra as $a_1\mathbf{n}$ and $a_2\mathbf{n}$, the Lie bracket in (C.5) yields the element $a_3\mathbf{n}$ as

$$[a_1\tilde{\mathbf{n}}, a_2\tilde{\mathbf{n}}] = a_3\tilde{\mathbf{n}} \quad (\text{C.7})$$

with $a_3 = 0$ since

$$[a_1\tilde{\mathbf{n}}, a_2\tilde{\mathbf{n}}] = a_1\tilde{\mathbf{n}}a_2\tilde{\mathbf{n}} - a_2\tilde{\mathbf{n}}a_1\tilde{\mathbf{n}} = \mathbf{0}_{3 \times 1} \quad (\text{C.8})$$

Thus, $SO(2)$ is Abelian since the Lie bracket vanishes.

C.3 Subgroups of the special Euclidean group $SE(3)$

The matrix \mathbf{A} in (C.5) is a $6 \times l$ matrix for $SE(3)$, being l the dimension of the subgroup. The most useful Lie subgroups of $SE(3)$ represent the lower pair joints listed in Table 5.1. Notice that all the subgroups in this Table, except \mathbb{R}^2 and $SO(3)$ (interpreted as planar and spherical joints), are Abelian.

Appendix D

Time integration on a Lie group

D.1 Implicit time integration on a Lie group

The general form of differential-algebraic equations on a Lie group take the form

$$\dot{q} = q \widetilde{\mathbf{A}} \boldsymbol{\eta} \quad (\text{D.1})$$

$$\mathbf{g}(q, \boldsymbol{\eta}, \dot{\boldsymbol{\eta}}) + \mathbf{A}^T \boldsymbol{\varphi}_q^T(q) \boldsymbol{\lambda} = \mathbf{0} \quad (\text{D.2})$$

$$\boldsymbol{\varphi}(q) = \mathbf{0} \quad (\text{D.3})$$

The integration method relies on the discretized differential-algebraic equations on a Lie group

$$\dot{q}_{n+1} = q_n \exp(\widetilde{\mathbf{A}} \mathbf{n}_{n+1}) \quad (\text{D.4})$$

$$\mathbf{g}(q_{n+1}, \boldsymbol{\eta}_{n+1}, \dot{\boldsymbol{\eta}}_{n+1}) + \mathbf{A}^T \boldsymbol{\varphi}_q^T(q_{n+1}) \boldsymbol{\lambda}_{n+1} = \mathbf{0} \quad (\text{D.5})$$

$$\boldsymbol{\varphi}(q_{n+1}) = \mathbf{0} \quad (\text{D.6})$$

and the integration formulae

$$\mathbf{n}_{n+1} = h \boldsymbol{\eta}_n + (0.5 - \beta) h^2 \mathbf{a}_n + \beta h^2 \mathbf{a}_{n+1} \quad (\text{D.7})$$

$$\boldsymbol{\eta}_{n+1} = \boldsymbol{\eta}_n + (1 - \gamma) h \mathbf{a}_n + \gamma h \mathbf{a}_{n+1} \quad (\text{D.8})$$

$$\mathbf{a}_{n+1} = \frac{1}{1 - \alpha_m} \left((1 - \alpha_f) \dot{\boldsymbol{\eta}}_{n+1} + \alpha_f \dot{\boldsymbol{\eta}}_n - \alpha_m \mathbf{a}_n \right) \quad (\text{D.9})$$

where n refers to the time step, h is the time step size and \mathbf{a} is a vector of pseudo-accelerations. Notice that \mathbf{n} has an index $n+1$ to indicate that it changes at each time step, but the value at the step n is not involved at step $n+1$. $\mathbf{n}_{n+1} \in \mathfrak{g}$ is a vectorial quantity interpreted as an increment about $q_n \in G$ and is projected onto the Lie group thanks to the exponential map, which introduces a *local parametrization* about q_n . The numerical parameters of the method can be selected to achieve a desired spectral radius $\rho \in [0, 1)$ at high frequency as

$$\alpha_m = \frac{2\rho - 1}{\rho + 1}; \quad \alpha_f = \frac{\rho}{\rho + 1}; \quad \gamma = \frac{3 - \rho}{2(\rho + 1)}; \quad \beta = \frac{1}{(\rho + 1)^2} \quad (\text{D.10})$$

Equations (D.4)–(D.6) are solved at each time step for the unknowns \mathbf{n}_{n+1} , $\boldsymbol{\eta}_{n+1}$, $\dot{\boldsymbol{\eta}}_{n+1}$ and \mathbf{a}_{n+1} . Since they are usually non-linear, a *Newton-Raphson* iterative procedure, which requires the linearization of the equations of motion, is used. A finite variation due to the Newton-Raphson procedure is denoted with $\Delta(\cdot)$.

Let us first consider the linearization with respect to q_{n+1} of Eq. D.4. According to (C.4) and considering the finite variation resulting from the local parametrization in (D.4) we have respectively

$$\Delta(q_{n+1}) = q_{n+1} \widetilde{\mathbf{A}} \Delta \mathbf{q}_{n+1} \quad (\text{D.11})$$

$$\Delta(q_{n+1}) = q_{n+1} (\mathbf{T}(\mathbf{n}_{n+1}) \mathbf{A} \Delta \mathbf{n}_{n+1}) \widetilde{} \quad (\text{D.12})$$

where \mathbf{T} is the tangent map operator. By comparing (D.11) and (D.12), we obtain

$$\mathbf{A} \Delta \mathbf{q}_{n+1} = \mathbf{T}(\mathbf{n}_{n+1}) \mathbf{A} \Delta \mathbf{n}_{n+1} \quad (\text{D.13})$$

Denoting the left side of Eq. D.5 as $\mathbf{r}(q_{n+1}, \boldsymbol{\eta}_{n+1}, \dot{\boldsymbol{\eta}}_{n+1}, \boldsymbol{\lambda}_{n+1})$, the linearization of (D.5)–(D.6) reads

$$D\mathbf{r} \cdot (\mathbf{A} \Delta \mathbf{q}_{n+1}) = \mathbf{K}_T \mathbf{A} \Delta \mathbf{q}_{n+1} = \mathbf{K}_T \mathbf{T}(\mathbf{n}_{n+1}) \mathbf{A} \Delta \mathbf{n}_{n+1} \quad (\text{D.14})$$

$$D\mathbf{r} \cdot (\Delta \boldsymbol{\eta}_{n+1}) = \mathbf{C}_T \Delta \boldsymbol{\eta}_{n+1} \quad (\text{D.15})$$

$$D\mathbf{r} \cdot (\Delta \dot{\boldsymbol{\eta}}_{n+1}) = \mathbf{M}_T \Delta \dot{\boldsymbol{\eta}}_{n+1} \quad (\text{D.16})$$

$$D\mathbf{r} \cdot (\Delta \boldsymbol{\lambda}_{n+1}) = \mathbf{A}^T \boldsymbol{\varphi}_q^T \Delta \boldsymbol{\lambda}_{n+1} \quad (\text{D.17})$$

$$D\boldsymbol{\varphi} \cdot (\mathbf{A} \Delta \mathbf{q}_{n+1}) = \boldsymbol{\varphi}_q \mathbf{A} \Delta \mathbf{q}_{n+1} = \boldsymbol{\varphi}_q \mathbf{T}(\mathbf{n}_{n+1}) \mathbf{A} \Delta \mathbf{n}_{n+1} \quad (\text{D.18})$$

where \mathbf{K}_T , \mathbf{C}_T and \mathbf{M}_T are respectively interpreted as the tangent stiffness, tangent damping and tangent mass matrices. These matrices are obtained from the variations of the internal and inertial forces. The linearization of the integration formula in (D.7)–(D.9) yields

$$\Delta \boldsymbol{\eta}_{n+1} = \gamma' \Delta \mathbf{n}_{n+1} \quad (\text{D.19})$$

$$\Delta \dot{\boldsymbol{\eta}}_{n+1} = \beta' \Delta \mathbf{n}_{n+1} \quad (\text{D.20})$$

where $\gamma' = \gamma/(\beta h)$ and $\beta' = (1 - \alpha_m)/(\beta h^2(1 - \alpha_f))$.

Eventually, the finite variations of the configuration increments \mathbf{n}_{n+1} and of the Lagrange multipliers $\boldsymbol{\lambda}_{n+1}$ are computed at each Newton-Raphson step by solving

$$\mathbf{S}_T \begin{bmatrix} \Delta \mathbf{n}_{n+1} \\ \Delta \boldsymbol{\lambda}_{n+1} \end{bmatrix} = - \begin{bmatrix} \mathbf{r}^* \\ \boldsymbol{\varphi}^* \end{bmatrix} \quad (\text{D.21})$$

where \mathbf{r}^* and $\boldsymbol{\varphi}^*$ are respectively the value of the residual and the constraints where the system is linearized and \mathbf{S}_T is the iteration matrix given by

$$\mathbf{S}_T = \begin{bmatrix} \beta' \mathbf{M}_T + \gamma' \mathbf{C}_T + \mathbf{K}_T \mathbf{T}(\mathbf{n}_{n+1}) \mathbf{A} & \mathbf{A}^T \boldsymbol{\varphi}_q^T \\ \boldsymbol{\varphi}_q \mathbf{T}(\mathbf{n}_{n+1}) \mathbf{A} & \mathbf{0} \end{bmatrix} \quad (\text{D.22})$$

The finite variations of the velocities and the accelerations are computed from $\Delta \mathbf{n}_{n+1}$ according to (D.19)–(D.20).

D.2 Time integration on $SO(3)$

The implicit time integration method applied to a rotating rigid body leads to

$$\mathbf{R}_{n+1} = \mathbf{R}_n \exp_{SO(3)}(\tilde{\mathbf{n}}_{n+1}) \quad (\text{D.23})$$

$$\mathbf{J}\dot{\boldsymbol{\omega}}_{n+1} + \tilde{\boldsymbol{\omega}}_{n+1}\mathbf{J}\boldsymbol{\omega}_{n+1} = \mathbf{0}_{3 \times 1} \quad (\text{D.24})$$

together with the following integration formulae

$$\mathbf{n}_{n+1} = h\boldsymbol{\omega}_n + (0.5 - \beta)h^2\mathbf{a}_n + \beta h^2\mathbf{a}_{n+1} \quad (\text{D.25})$$

$$\boldsymbol{\omega}_{n+1} = \boldsymbol{\omega}_n + (1 - \gamma)h\mathbf{a}_n + \gamma h\mathbf{a}_{n+1} \quad (\text{D.26})$$

$$\mathbf{a}_{n+1} = \frac{1}{1 - \alpha_m} ((1 - \alpha_f)\dot{\boldsymbol{\omega}}_{n+1} + \alpha_f\dot{\boldsymbol{\omega}}_n - \alpha_m\mathbf{a}_n) \quad (\text{D.27})$$

The iteration matrix in Eq. D.22 is given by

$$\mathbf{S}_T = \beta'\mathbf{J} + \gamma'(\tilde{\boldsymbol{\omega}}\mathbf{J} - \tilde{\mathbf{J}}\boldsymbol{\omega}) \quad (\text{D.28})$$

Notice that there is no tangent stiffness matrix here since the equilibrium equations do not depend on \mathbf{R} .

D.3 Time integration on $SE(3)$

The implicit time integration method applied to the equations of motion of a rigid body leads to

$$\mathbf{H}_{n+1} = \mathbf{H}_n \exp_{SE(3)}(\tilde{\mathbf{n}}_{n+1}) \quad (\text{D.29})$$

$$\mathbf{M}\dot{\boldsymbol{\eta}}_{n+1} - \hat{\boldsymbol{\eta}}_{n+1}^T \mathbf{M}\boldsymbol{\eta}_{n+1} = \mathbf{0}_{6 \times 1} \quad (\text{D.30})$$

The time integration formulae are readily identical to (D.7)–(D.9). The iteration matrix in Eq. D.22 is given by

$$\mathbf{S}_T = \begin{bmatrix} \beta'm\mathbf{I}_{3 \times 3} + \gamma'm\tilde{\boldsymbol{\omega}} & -\gamma'm\tilde{\mathbf{u}} \\ \mathbf{0}_{3 \times 3} & \beta'\mathbf{J} + \gamma'(\tilde{\boldsymbol{\omega}}\mathbf{J} - \tilde{\mathbf{J}}\boldsymbol{\omega}) \end{bmatrix} \quad (\text{D.31})$$

Notice that there is no tangent stiffness matrix here because the equilibrium equations do not depend on \mathbf{H} .

Bibliography

- [1] T. M. Wasfy and A. K. Noor, “Computational strategies for flexible multi-body systems,” *Applied Mechanics Reviews*, vol. 56, no. 6, pp. 553–613, 2003.
- [2] A. A. Shabana, *Dynamics of multibody systems*. Cambridge university press, 2013.
- [3] R. Featherstone, *Rigid body dynamics algorithms*. Springer, 2014.
- [4] M. Géradin and A. Cardona, *Flexible multibody dynamics: a finite element approach*. Wiley, 2001.
- [5] B. Simeon, “Computational flexible multibody dynamics,” in *A Differential-Algebraic Approach. Differential-Algebraic Equations Forum*. Springer, Heidelberg, Springer, 2013.
- [6] W. J. Book, “Recursive lagrangian dynamics of flexible manipulator arms,” *The International Journal of Robotics Research*, vol. 3, no. 3, pp. 87–101, 1984.
- [7] T. Belytschko and B. Hsieh, “Non-linear transient finite element analysis with convected co-ordinates,” *International Journal for Numerical Methods in Engineering*, vol. 7, no. 3, pp. 255–271, 1973.
- [8] T. Belytschko, W. K. Liu, B. Moran, and K. Elkhodary, *Nonlinear finite elements for continua and structures*. John wiley & sons, 2013.
- [9] M. A. Crisfield, J. J. Remmers, C. V. Verhoosel, *et al.*, *Nonlinear finite element analysis of solids and structures*. John Wiley & Sons, 2012.
- [10] J. Housner, “Convected transient analysis for large space structures maneuver and deployment,” *NASA report*, 1984.
- [11] J. C. Simo and L. Vu-Quoc, “A three-dimensional finite-strain rod model. part ii: Computational aspects,” *Computer methods in applied mechanics and engineering*, vol. 58, no. 1, pp. 79–116, 1986.
- [12] J. Simo and L. Vu-Quoc, “On the dynamics of flexible beams under large overall motions—the plane case: Part ii,” *Journal of applied mechanics*, vol. 53, no. 4, pp. 855–863, 1986.

- [13] O. A. Bauchau, *Flexible multibody dynamics*, vol. 176. Springer Science & Business Media, 2010.
- [14] V. Sonneville, *A geometric local frame approach for flexible multibody systems*. 2015.
- [15] O. A. Bauchau, P. Betsch, A. Cardona, J. Gerstmayr, B. Jonker, P. Masarati, and V. Sonneville, “Validation of flexible multibody dynamics beam formulations using benchmark problems,” *Multibody system dynamics*, vol. 37, no. 1, pp. 29–48, 2016.
- [16] J. Burgner-Kahrs, D. C. Rucker, and H. Choset, “Continuum robots for medical applications: A survey,” *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1261–1280, 2015.
- [17] Y. Ganji, F. Janabi-Sharifi, *et al.*, “Catheter kinematics for intracardiac navigation,” *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 3, pp. 621–632, 2009.
- [18] M. D. Kutzer, S. M. Segreti, C. Y. Brown, M. Armand, R. H. Taylor, and S. C. Mears, “Design of a new cable-driven manipulator with a large open lumen: Preliminary applications in the minimally-invasive removal of osteolysis,” in *2011 IEEE International Conference on Robotics and Automation*, pp. 2913–2920, IEEE, 2011.
- [19] R. J. Webster III and B. A. Jones, “Design and kinematic modeling of constant curvature continuum robots: A review,” *The International Journal of Robotics Research*, vol. 29, no. 13, pp. 1661–1683, 2010.
- [20] M. W. Hannan and I. D. Walker, “Kinematics and the implementation of an elephant’s trunk manipulator and other continuum style robots,” *Journal of Field Robotics*, vol. 20, no. 2, pp. 45–63, 2003.
- [21] B. A. Jones and I. D. Walker, “Kinematics for multisection continuum robots,” *IEEE Transactions on Robotics*, vol. 22, no. 1, pp. 43–55, 2006.
- [22] P. Sears and P. Dupont, “A steerable needle technology using curved concentric tubes,” in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2850–2856, IEEE, 2006.
- [23] R. J. Webster III, J. M. Romano, and N. J. Cowan, “Mechanics of precurved-tube continuum robots,” *IEEE Transactions on Robotics*, vol. 25, no. 1, pp. 67–78, 2009.
- [24] P. E. Dupont, J. Lock, B. Itkowitz, and E. Butler, “Design and control of concentric-tube robots,” *IEEE Transactions on Robotics*, vol. 26, no. 2, pp. 209–225, 2010.

- [25] D. C. Rucker, R. J. Webster III, G. S. Chirikjian, and N. J. Cowan, “Equilibrium conformations of concentric-tube continuum robots,” *The International journal of robotics research*, vol. 29, no. 10, pp. 1263–1280, 2010.
- [26] D. C. Rucker and R. J. Webster III, “Statics and dynamics of continuum robots with general tendon routing and external loading,” *IEEE Transactions on Robotics*, vol. 27, no. 6, pp. 1033–1044, 2011.
- [27] T. Mahl, A. Hildebrandt, and O. Sawodny, “A variable curvature continuum kinematics for kinematic control of the bionic handling assistant,” *IEEE transactions on robotics*, vol. 30, no. 4, pp. 935–949, 2014.
- [28] G. S. Chirikjian, *Theory and applications of hyper-redundant robotic manipulators*. 1992.
- [29] G. S. Chirikjian and J. W. Burdick, “Kinematically optimal hyper-redundant manipulator configurations,” *IEEE transactions on Robotics and Automation*, vol. 11, no. 6, pp. 794–806, 1995.
- [30] G. S. Chirikjian and J. W. Burdick, “The kinematics of hyper-redundant robot locomotion,” *IEEE transactions on robotics and automation*, vol. 11, no. 6, pp. 781–793, 1995.
- [31] T. Zheng, D. T. Branson, R. Kang, M. Cianchetti, E. Guglielmino, M. Follador, G. A. Medrano-Cerda, I. S. Godage, and D. G. Caldwell, “Dynamic continuum arm model for use with underwater robotic manipulators inspired by octopus vulgaris,” in *2012 IEEE International Conference on Robotics and Automation*, pp. 5289–5294, IEEE, 2012.
- [32] J. Jung, R. S. Penning, and M. R. Zinn, “A modeling approach for robotic catheters: Effects of nonlinear internal device friction,” *Advanced Robotics*, vol. 28, no. 8, pp. 557–572, 2014.
- [33] S. Misra, K. B. Reed, B. W. Schafer, K. Ramesh, and A. M. Okamura, “Mechanics of flexible needles robotically steered through soft tissue,” *The International journal of robotics research*, vol. 29, no. 13, pp. 1640–1660, 2010.
- [34] W. S. Rone and P. Ben-Tzvi, “Continuum robot dynamics utilizing the principle of virtual power,” *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 275–287, 2014.
- [35] K. Xu and N. Simaan, “Analytic formulation for kinematics, statics, and shape restoration of multibackbone continuum robots via elliptic integrals,” *Journal of Mechanisms and Robotics*, vol. 2, no. 1, p. 011006, 2010.
- [36] J. Selig and X. Ding, “A screw theory of timoshenko beams,” *Journal of Applied Mechanics*, vol. 76, no. 3, p. 031003, 2009.

- [37] D. Trivedi, A. Lotfi, and C. D. Rahn, “Geometrically exact models for soft robotic manipulators,” *IEEE Transactions on Robotics*, vol. 24, no. 4, pp. 773–780, 2008.
- [38] V. Sonneville, A. Cardona, and O. Brüls, “Geometrically exact beam finite element formulated on the special euclidean group $se(3)$,” *Computer Methods in Applied Mechanics and Engineering*, vol. 268, pp. 451–474, 2014.
- [39] G. S. Chirikjian, “Conformational modeling of continuum structures in robotics and structural biology: A review,” *Advanced Robotics*, vol. 29, no. 13, pp. 817–829, 2015.
- [40] R. Ball, “A treatise on the theory of screws, 1900,” 1900 (Reprinted 1998).
- [41] R. W. Brockett, “Robotic manipulators and the product of exponentials formula,” in *International symposium on the mathematical theory of networks and systems*, pp. 120–129, 1983.
- [42] R. M. Murray, Z. Li, S. S. Sastry, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.
- [43] J. M. Selig, *Geometric fundamentals of robotics*. Springer Science & Business Media, 2004.
- [44] K. M. Lynch and F. C. Park, *Modern Robotics: Mechanics, Planning, and Control*. Cambridge University Press, 2017.
- [45] O. C. Zienkiewicz, R. L. Taylor, O. C. Zienkiewicz, and R. L. Taylor, *The finite element method*, vol. 3. McGraw-hill London, 1977.
- [46] A. Ibrahimbegovic, *Nonlinear solid mechanics: theoretical formulations and finite element solution methods*, vol. 160. Springer Science & Business Media, 2009.
- [47] J. Bonet and R. D. Wood, *Nonlinear continuum mechanics for finite element analysis*. Cambridge university press, 1997.
- [48] V. Sonneville and O. Brüls, “A formulation on the special euclidean group for dynamic analysis of multibody systems,” *Journal of Computational and Nonlinear Dynamics*, vol. 9, no. 4, p. 041002, 2014.
- [49] C. Lanczos, *The variational principles of mechanics*. Courier Corporation, 2012.
- [50] P. E. Crouch and R. Grossman, “Numerical integration of ordinary differential equations on manifolds,” *Journal of Nonlinear Science*, vol. 3, no. 1, pp. 1–33, 1993.

- [51] H. Munthe-Kaas, “Runge-kutta methods on lie groups,” *BIT Numerical Mathematics*, vol. 38, no. 1, pp. 92–111, 1998.
- [52] O. Brüls and A. Cardona, “On the use of lie group time integrators in multi-body dynamics,” *Journal of Computational and Nonlinear Dynamics*, vol. 5, no. 3, p. 031002, 2010.
- [53] O. Brüls, A. Cardona, and M. Arnold, “Lie group generalized- α time integration of constrained flexible multibody systems,” *Mechanism and Machine Theory*, vol. 48, pp. 121–137, 2012.
- [54] Z. Terze, A. Müller, and D. Zlatar, “Lie-group integration method for constrained multibody systems in state space,” *Multibody system dynamics*, vol. 34, no. 3, pp. 275–305, 2015.
- [55] F. Demoures, F. Gay-Balmaz, S. Leyendecker, S. Ober-Blöbaum, T. S. Ratiu, and Y. Weinand, “Discrete variational lie group formulation of geometrically exact beam dynamics,” *Numerische Mathematik*, vol. 130, no. 1, pp. 73–123, 2015.
- [56] L. Sciavicco and B. Siciliano, *Modelling and control of robot manipulators*. Springer Science & Business Media, 2012.
- [57] R. Christensen, *Theory of viscoelasticity: an introduction*. Elsevier, 2012.
- [58] A. De Luca and B. Siciliano, “Closed-form dynamic model of planar multilink lightweight robots,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 21, no. 4, pp. 826–839, 1991.
- [59] H. Goldstein, *Classical mechanics*. Addison-Wesley, Reading, MA, second edition, 1980.
- [60] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, “Robotics: modelling, planning and control, ser. advanced textbooks in control and signal processing,” *Springer*, vol. 26, p. 29, 2009.
- [61] S. Grazioso, V. Sonneville, G. Di Gironimo, O. Bauchau, and B. Siciliano, “A nonlinear finite element formalism for modelling flexible and soft manipulators,” in *IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots*, pp. 185–190, IEEE, 2016.
- [62] A. Comsol, “Comsol multiphysics user’s guide,” *Version: September*, vol. 10, p. 333, 2005.
- [63] A. Kivila, W. Book, and W. Singhose, “Exact modeling of n-link spatial serial structures using transfer matrices,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 139, no. 11, p. 114502, 2017.

- [64] W. Kivila, Arto Book and W. Singhose, “Modeling spatial multi-link flexible manipulator arms based on system modes,” in *2017 IEEE 56th Annual Conference on Decision and Control*, p. to appear, IEEE, 2017.
- [65] E. Dowell and J. J. Traybar, “An experimental study of the nonlinear stiffness of a rotor blade undergoing flap, lag and twist deformations,” 1975.
- [66] R. Buckingham and A. Loving, “Remote-handling challenges in fusion research and beyond,” *Nature Physics*, vol. 12, no. 5, pp. 391–393, 2016.
- [67] J. Keep, S. Wood, N. Gupta, M. Coleman, and A. Loving, “Remote handling of demo breeder blanket segments: Blanket transporter conceptual studies,” *Fusion Engineering and Design*, 2017.
- [68] A. Loving, O. Crofts, N. Sykes, D. Iglesias, M. Coleman, J. Thomas, J. Harman, U. Fischer, J. Sanz, M. Siuko, *et al.*, “Pre-conceptual design assessment of demo remote maintenance,” *Fusion Engineering and Design*, vol. 89, no. 9, pp. 2246–2250, 2014.
- [69] M. Coleman, N. Sykes, D. Cooper, D. Iglesias, R. Bastow, A. Loving, and J. Harman, “Concept for a vertical maintenance remote handling system for multi module blanket segments in demo,” *Fusion Engineering and Design*, vol. 89, no. 9, pp. 2347–2351, 2014.
- [70] P. Francis, K. Eastwood, V. Bodani, K. Price, K. Upadhyaya, D. Podolsky, H. Azimian, T. Looi, and J. Drake, “Miniaturized instruments for the da vinci research kit: Design and implementation of custom continuum tools,” *IEEE Robotics & Automation Magazine*, 2017.
- [71] P. Polygerinos, Z. Wang, K. C. Galloway, R. J. Wood, and C. J. Walsh, “Soft robotic glove for combined assistance and at-home rehabilitation,” *Robotics and Autonomous Systems*, vol. 73, pp. 135–143, 2015.
- [72] P. Maeder-York, T. Clites, E. Boggs, R. Neff, P. Polygerinos, D. Holland, L. Stirling, K. Galloway, C. Wee, and C. Walsh, “Biologically inspired soft robot for thumb rehabilitation,” *Journal of Medical Devices*, vol. 8, no. 2, p. 020933, 2014.
- [73] P. Polygerinos, S. Lyne, Z. Wang, L. F. Nicolini, B. Mosadegh, G. M. Whitesides, and C. J. Walsh, “Towards a soft pneumatic glove for hand rehabilitation,” in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1512–1517, IEEE, 2013.
- [74] F. Renda, M. Giorelli, M. Calisti, M. Cianchetti, and C. Laschi, “Dynamic model of a multibending soft robot arm driven by cables,” *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1109–1122, 2014.

- [75] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>.
- [76] D. J. MacKay, “Bayesian interpolation,” *Neural computation*, vol. 4, no. 3, pp. 415–447, 1992.
- [77] F. D. Foresee and M. T. Hagan, “Gauss-newton approximation to bayesian learning,” in *International Conference on Neural Networks*, vol. 3, pp. 1930–1935, IEEE, 1997.
- [78] W. Singhose, “Command shaping for flexible systems: A review of the first 50 years,” *International Journal of Precision Engineering and Manufacturing*, vol. 10, no. 4, pp. 153–168, 2009.
- [79] J. Vaughan, A. Yano, and W. Singhose, “Comparison of robust input shapers,” *Journal of Sound and Vibration*, vol. 315, no. 4, pp. 797–815, 2008.
- [80] S. Grazioso, G. Di Gironimo, W. Singhose, and B. Siciliano, “Input predictive shaping for vibration control of flexible systems,” in *2017 IEEE Conference on Control Technology and Applications*, pp. 305–310, IEEE, 2017.
- [81] J. J. Moré, “The levenberg-marquardt algorithm: implementation and theory,” in *Numerical analysis*, pp. 105–116, Springer, 1978.