

# Abstract

---

## Design, Modeling, and Control Strategies for Soft Robots<sup>1</sup>

---

In the past two decades, the field of soft robotics has sparked significant interest among many scientific disciplines. Contrary to rigid robots, soft robots explore soft materials that significantly enhance the robot's dexterity, enable a rich family of motion primitives, and enhance environmental robustness regarding contact and impact. Since its inception, soft robotics has exemplified its potential in diverse areas such as safe manipulation, adaptive grasping, exploration under environmental uncertainty, rehabilitation, and the bio-mimicry of many animals. By exploring the uncharted versatile nature of soft materials, soft robotics places the first steppingstones towards achieving biological performance in modern-day's robotics. This thesis aims to further advances in soft robotics by addressing the open multi-disciplinary challenges within this young field of research.

Although soft materials harbor many advantages akin to biology, which are difficult to achieve for rigid robotics, it also roots many fundamental problems. First is the issue of soft robotic design. Traditional robotic design emphasizes high structural rigidity and weight minimization – a well-established practice in engineering. On the other hand, soft robotic design relishes minimal structural rigidity for motion, leading to complex, highly nonlinear relations between input and output. Besides, distributed soft actuation, imparted by gravitational and inertial effects acting on the elastic continuum body, introduce joint mobilities that are in many cases uncontrollable nor aligned with the control objective, e.g., grasping and manipulation. As describing the underlying continuum mechanics and applying such mathematical theory to the systematic design of soft robots is challenging, most soft robotic systems are developed ad hoc. Second, a direct duality of the previous challenge, is dealing with the innate infinite-dimensionality from a model-based perspective – particularly with model-based feedback in mind. The transition from rigid to flexible has introduced a new control paradigm: the trade-off between precision and speed in a numerical setting. Not only is control theory for soft robotics in stages of inception, but deriving accurate and numerically efficient model-based controllers is challenging due to large nonlinear deformations of the soft robotic continuum.

In light of these challenges, this thesis proposes a unified framework of theoretical and experimental applications for (*i*) the structural design and fabrication of soft actuation elements embedded into homogeneous elastic bodies, (*ii*) the development of efficient dynamic models of soft continuum manipulators, and (*iii*) the transfer of control theory standard in rigid robotics to soft robotics.

---

<sup>1</sup>by B.J. Caasenbrood (Brandon) Last modified: June 21, 2022

The first part of the thesis addresses the design problem by proposing novel computer-automated design algorithms for developing efficient soft actuators. These algorithms account for the underlying continuum mechanics described by a set of partial differential equations. Tailoring a user-defined objective to the desired motion and control reachability, the algorithm produces an implicit representation of the optimal soft material distribution within a fixed design space. Many generative designs for a diverse set of soft actuation morphologies are produced including, but not limited to, soft rotational actuators, soft artificial muscles, and soft grippers. In what follows, an optimal design for a soft robotic manipulator with an adaptive gripper is synthesized, and through Additive Manufacturing (AM) of printable flexible material, the sim-to-real boundary is passed. The proposed approach does not only accelerate design convergence, it also extends to the development of unexplored soft robot morphologies currently unseen in literature.

The second part of the thesis answers the question on modeling for control applicable to a class of soft robotic systems – most notably soft continuum manipulators. The thesis proposes a reduced-order modeling strategy for soft robotics, whose dynamics are derived through differential geometric theory on spatial beams. Besides discussing earlier modeling strategies, the thesis also proposes a new strain-based parametrization approach that ensures the structural geometry information and the underlying continuum mechanics are preserved when synthesizing the reduced-order beam models – a possible answer to the aforementioned control paradigm. To enhance numerical performance further, new spatio-temporal integration schemes are also proposed that exploit the geometric structure of such soft beam models, resulting in real-time simulation with sufficient numerical precision purposefully tailored for controller design.

The third part of the thesis treats the development of model-based controllers that can be employed in various control scenarios akin to control for traditional rigid robotics, *e.g.*, inverse kinematics and motion planning, set-point stabilization, trajectory tracking, and multi-point grasping of arbitrary objects. The stabilizing controller is rooted in an energy-based formulism, providing robustness even when faced with material uncertainties. The controller’s effectiveness is demonstrated both in simulation and experiments for various soft robotic systems that share a resemblance to biology, *e.g.*, the elephant’s trunk or the tentacle of an octopus.

The main contribution of the thesis is the accumulation of multi-disciplinary tools compressed into one unifying framework for the design, modeling, and control of soft robots, which ranges from theoretical to experimental domain.

**Keywords:** Soft Robots, Hyper-redundant Robots, Design Optimization, Continuum Mechanics, Reduced-order Modeling, Model-based Control, Additive Manufacturing.

# Contents

<b>Abstract</b>	i
<b>Nomenclature</b>	v
<b>I Modeling of Soft Robots</b>	1
1 Dynamic modeling of Soft Robots – PCC case	3
1.1 Introduction . . . . .	4
1.2 System development . . . . .	7
1.3 Generalized models for soft manipulators . . . . .	9
1.3.1 Curve kinematics . . . . .	10
1.3.2 Curve dynamics using Euler-Lagrange . . . . .	17
1.3.3 Overall dynamic model . . . . .	18
1.4 Extension to multi-link systems . . . . .	19
1.5 Accelerated computation of PDE-like systems using Lagrangian formalism . . . . .	21
1.6 System identification . . . . .	23
1.6.1 Finite element method and hyper-elasticity . . . . .	23
1.6.2 Hyper-elasticity on joint space . . . . .	26
1.6.3 Visco-elastic creep . . . . .	27
1.7 Adaptive control . . . . .	30
1.7.1 Passivity-based Adaptive Control . . . . .	30
1.8 Numerical and experimental implementation . . . . .	32
1.8.1 Experimental platform . . . . .	35
1.9 Summary . . . . .	44
<b>Bibliography</b>	45



# Nomenclature

## Vector and matrix notation

$x$	Scalar notation
$\boldsymbol{x}$	Vector notation
$\boldsymbol{X}$	Matrix notation
$\boldsymbol{\chi}$	Tensor notation
$\mathcal{Q}$	Manifold
$T_{\mathcal{Q}}$	Tangent space

## Compact sets

$\emptyset$	Empty set
$\mathbb{R}$	Set of real numbers
$\mathbb{R}^n$	$n$ -dimensional Euclidean space
$\mathbb{R}_{>0}$	Strictly positive reals
$\mathbb{R}_{\geq 0}$	Positive reals
$\mathbb{N}$	Set of natural numbers
$\mathbb{T}$	Finite time horizon
$\mathbb{X}$	1-dimensional spatial set or domain ( <i>i.e.</i> , line)
$\mathbb{V}$	3-dimensional spatial set or domain ( <i>i.e.</i> , volume)

## Groups

$\text{id}$	Identity
$\text{SO}(n)$	Lie group of rotations on $\mathbb{R}^n$ ( <i>i.e.</i> , special orthonormal matrices)
$\text{SE}(n)$	Lie group of homogeneous transformations on $\mathbb{R}^n$
$\text{se}(n)$	Lie algebra of $\text{SO}(n)$
$\text{so}(n)$	Lie algebra of $\text{SE}(n)$

## Vector- and matrix operations

$(\cdot)^\top$	Transpose
$(\cdot)'$	First time derivative
$(\cdot)''$	Second time derivative
$(\hat{\cdot}), (\cdot)^\wedge$	Isomorphism from $\mathbb{R}^6 \rightarrow \text{se}(3)$
$(\cdot), (\cdot)^\vee$	Isomorphism from $\text{se}(3) \rightarrow \mathbb{R}^6$
$(\cdot)^\circ$	Reference or rest configuration
$(\cdot)^*$	Optimal solution
$(\cdot)^{-1}$	Square matrix inverse
$(\cdot)^\dagger$	Moore-Penrose pseudo inverse
$(\cdot)^+$	Generalized matrix inverse
$(\cdot)^\perp$	Annihilator

## Operators and letter-like symbols

$\delta$	Variation of a field
$\partial$	Boundary of a set
int	Interior of a set
$\sup_t$	Supremum over continuous time $t$
dim	Dimension of vector
trace	Trace of matrix
diag	Diagonal of matrix
$\ \cdot\ _{\text{ma}}$	Mean absolute norm
$\ \cdot\ _{\text{rms}}$	Root-mean-square norm

## Acronyms

CoM	Center of mass
CoR	Coefficient of restitution
PneuNet	Pneumatic network
SRM	Soft Robotic Manipulator
TopoOpt	Topology Optimization
TopoOpt	Topology Optimization

# I

## Modeling of Soft Robots



# 1

## Dynamic modeling – The Constant Strain Approach

**Abstract -** In this chapter, the continuum dynamics of the soft robot are derived through the differential geometry of spatial curves, which are then related to Finite-Element data to capture the intrinsic geometric and material nonlinearities. To enable fast simulations, a reduced-order integration scheme is introduced to compute the dynamic Lagrangian matrices efficiently, which in turn allows for real-time (multi-link) models with sufficient numerical precision. By exploring the passivity and using the parametrization of the hyper-elastic model, we propose a passivity-based adaptive controller that enhances robustness towards material uncertainty and unmodeled dynamics – slowly improving their estimates online. As a study case, a fully 3D-printed soft robot manipulator is developed, which shows good correspondence with the dynamic model under various conditions, *e.g.*, natural oscillations, forced inputs, and under tip-loads. The solidity of the approach is demonstrated through extensive simulations, numerical benchmarks, and experimental validations.

---

This chapter is based on: B.J. Caasenbrood, A.Y. Pogromsky, and H. Nijmijer. *Control-oriented Models for Hyper-elastic Soft Robots through Differential Geometry of Curves*. Soft Robotics, 2022.

A detailed list of the differences between this chapter and the article on which it is based is provided in the '*Modifications*' chapter of this thesis.

## 1.1 Introduction

Traditional robots are made from rigid and dense materials that ensure accurate and repeatable motions. While rigid robotics excel at fast and precise motion, their structural rigidity lacks the compliance and mechanical robustness needed for safe and passive interaction in an unknown environment. Soft robotics is a field of robotics that aims to improve the motion complexity and environmental robustness that is generally lacking its rigid counterpart. To further promote these topics in robotics, researchers aim to mimic living creatures by developing bio-inspired robots with similar morphologies and mechanical properties [1–6]. In soft robotics, the hyper-flexible and continuum-bodied structure provides them with a rich family of motion primitives. Besides bio-mimicry, soft robotics has proven to be a prominent alternative for rigid robotics with a variety of applications, e.g., manipulation and adaptive grasping [7], untethered locomotion and exploration through uncertain environments [5, 8, 9], rehabilitation [10], and even minimal-invasive surgery [11, 12]. Although the popularity of the field has increased exponentially in recent years, one of the first soft robots dates back already to the early 1990s, e.g., the work of Suzumori et al. (1991, [2]). Yet, despite years of soft robotics research, their intrinsic hyper-flexible nature still possesses numerous challenges on modeling and control.

One major challenge in modeling is that the soft robot’s elastic body undergoes large, continuous deformation. Since its inception, numerous works have addressed the kinematics for soft continuum robots [13–15]; yet, its original framework stems from hyper-redundant robotics nearly a decade earlier [16]. Similar to soft robots, hyper-redundant robots exploit their high joint redundancy to achieve a wider ranges of tasks (e.g., shape control and collision avoidance) besides end-effector manipulation. To some extent, soft robots can be seen as the successor to hyper-redundant robots in which rigid mechanical joints or links are substituted with hyper-flexible soft elements. As a result, their dynamics involve a continuously deformable inertial body rather than the classical notion of rigid bodies. As such, conventional modeling approaches cannot be applied directly to these continuously deformable robots, stressing the importance of novel modeling strategies. In this respect, the dynamics of a continuously deformable soft robot are in theory of an infinite-dimensional nature. This paradigm shift has further emphasized the challenges in control-oriented modeling of soft robots; as their physical description are often more suited for a Partial-Differential Equations (PDEs) rather than Ordinary Differential Equations (ODEs).

Recently, some significant steps have been made toward formulating reduced-order ODE models for elastic continuum soft robots, paving a path toward model-based controllers. Perhaps one of the most popular techniques of spatial reduction is the so-called Piece-wise Constant Curvature (PCC) model. The PCC model

## 1.1. Introduction

5

---

assumes that the continuum shape can be described using a number of spatially-constant curves which are parametrized using a set of generalized coordinates. Although PCC models can be seen as a significant oversimplification of true continuum mechanics at hand, these models have proven to be remarkably viable for various control applications. Besides its use in inverse kinematic control [5, 13, 17], PCC models have also shown to be suitable for feedforward controllers as demonstrated by Falkenhahn et al. (2015, [1]); and more recently, closed-loop feedback controllers by Della Santina et al. (2019-2020, [18, 19]). Although the aforementioned works utilize the lumped-mass description, others have employed PCC models with uniform mass distribution [3, 4, 20–22] and current models even extend beyond the constant curvature [15, 16, 18]. However, in the face of significant external loading or (distributed) contact with the environment, the PCC assumption is relatively conservative and leads to undesired kinematic constraints on the continuum deformation. Besides, these models often need additional identification to model compliance as they do not originate from a continuum mechanical framework.

On the other hand, Finite-Element Method (FEM) models do originate from continuum mechanics and due to their PDE description provide a more accurate representation of deformations; and are particularly suited to deal with geometric and material nonlinearities. Duriez et al. (2013, [23]) and related works [24–26] showed that reduced-order FEM models can play an important role in closed-loop control – allowing accurate volumetric deformation and hyper-elastic behavior. Although such real-time simulations for FEM-based models are possible, a significant state-reduction is required to ensure sufficient computational speed. In the process, FEM-based models often lose desirable control properties, e.g., passivity preservation, which might play an important role in control. An alternative modeling strategy is the recently emerging geometrically-exact Cosserat-beam model. Similar to the PCC models, the Cosserat models have the merit benefit that they can be structured into a standard Lagrangian form – the basis for robotics control theory. Rooted in a geometric method for describing the continuum mechanics using Lie theory proposed by Simo et al. (1986, [27]), Boyer et al. (2021, [28, 29]) proposed a geometrically-exact modeling framework for Cosserat beams using non-linear parametrization of the strain field. Other examples include the work of Renda et al. (2018, [20, 30]) providing various options for Piecewise-Constant Strain (PCS) and Variable Strain modeling approaches. Although recent variants of the Cosserat models offer good computational performance [31, 32], its use in model-based control is slowly upcoming.

In this respect, the topic of reduced-order modeling of soft robots is an active area of research. Yet, a challenge that is frequently overlooked in control-orientated research is the anisotropic material behavior, mechanical saturation, and more importantly, the nonlinear and possibly time-varying nature of the highly hyper-

elastic soft materials [1, 15, 21, 31]. This is further amplified by the fact that soft robots are known for their diversity in elastic materials and corresponding morphologies. Mustaza et al. (2019, [33]) proposed a modified nonlinear Kelvin-Voigt material model to embody the complex material behavior of silicone-composite manipulators (so-called STIFF-FLOP actuators). A similar silicone composite actuator was experimentally validated by Sadati et. al (2020, [34]) who proposed a novel modeling approach with an appendage-dependent Hookean model and viscous power-law to describe nonlinear and time-dependent material effects, respectively. Both nonlinear material models show good correspondence with physical soft robots under various dynamic conditions, yet they lack general transferability to the soft robots with different geometries – intrinsically captured by FEM-driven models. As of today, there are few control-oriented models that both offer geometry and material versatility similar to FEM models and the control convenience similar to spatial curve models.

Ultimately, the strong nonlinearities paired with its continuous nature encourage the use of model-based controllers. Nevertheless, regarding the aforementioned model-based control approaches [1, 18, 19], the stability and performance of the closed-loop system could be undermined by uncertainties in physical parameters or unmodelled dynamics. Particularly for state-feedback linearization (e.g., inverse dynamic), as the inversion of inaccurately estimated systems could lead to poor performance and even instability. Adaptive control [35, 36] or energy-based controllers [37] might offer the needed robustness towards material uncertainties and unmodelled dynamics. Unfortunately, up till now, the applicability of adaptive and energy-based control techniques on soft robotics is scarcely explored. Franco et al. (2020,[38]) used an adaptive energy-based controller that compensates for external disturbances on the end-effector, yet this controller can be extended to include various slowly-varying material uncertainties, e.g., hyper-elasticity, and viscosity.

The contributions of the work are two-fold. First, to derive a finite-dimensional dynamic model of a continuum soft robot, where we briefly recapitulate existing modeling techniques for soft robot manipulators. To address the issue of infinite-dimensionality, we explore the PCC condition that allows for a low-dimensional description of the continuum dynamics. Although such modeling approaches have been thoroughly developed, we will address two issues that will aid the development of model-based controllers. We aim to bridge the gap between the PCC model and the underlying continuum mechanics by matching the quasi-static behavior to a Finite-Element-driven model (FEM), and we propose a reduced-order integration scheme using Matrix-Differential Equations (MDEs) to compute the spatio-temporal dynamics in real-time. Preliminary results of this work were shown in Caasenbrood et al. (2020, [39]).

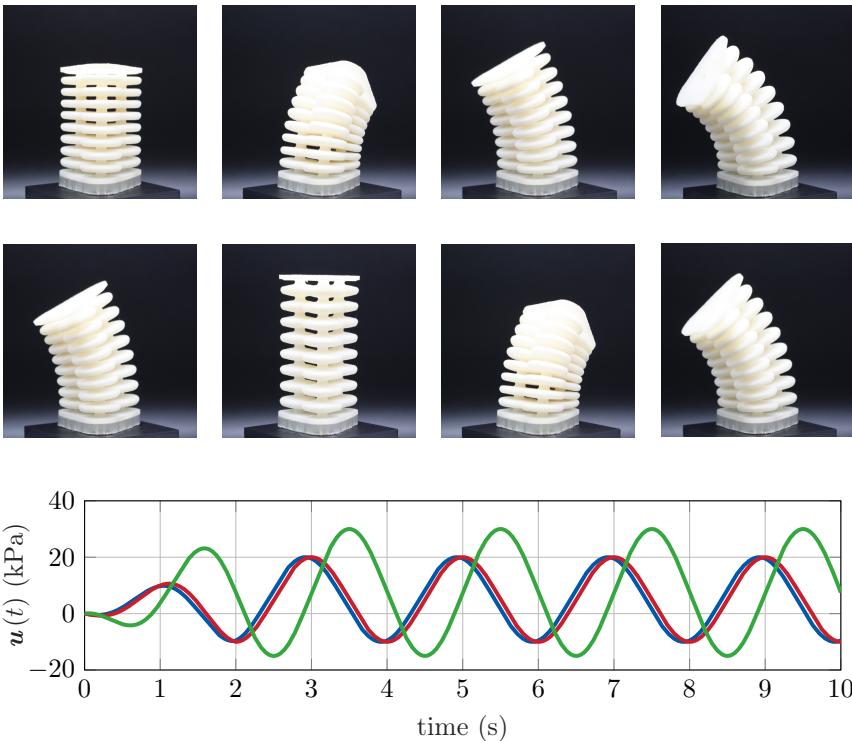
Second, in regards to the FEM-based hyper-elastic modeling and the possible

presence of unmodelled dynamics (e.g., material uncertainties or external loads on the end-effector), a passivity-based adaptive controller is proposed that enhances robustness towards material uncertainties and unmodelled dynamics in closed-loop, slowly improving their estimates online. All source code is made publicly available at Caasenbrood et al. (2020, [40]) (see the open software repository)

## 1.2 System development

By using additive manufacturing, we developed a soft and flexible robot manipulator that is suitable for pick-and-place applications. The 3-DOF soft robot can be seen in Figure 1.1. The soft robot manipulator in this work is loosely inspired by the elephant whose trunk-appendage consists mainly of parallel muscles without skeletal support. The anatomy of the elephant’s trunk provides an excellent study case, as they naturally exhibit continuum-body bending and moderate elongation [1, 13, 21]. Similar to the earlier soft robotic designs [1, 2], the developed soft robot can undergo three-dimensional movement by inflation or deflation of embedded pneumatic bellow network. The pneumatic network has three unique inputs, which labeled  $\mathbf{u} = (u_1, u_2, u_3)^\top$ . By varying the input  $\mathbf{u}$ , the soft robot can achieve bending in any preferred direction by differential pressurization of each channel (<0.1 MPa), e.g.,  $u_1 > u_2 = u_3 > 0$ . Whereas, simultaneous pressurization accomplishes moderate elongation, i.e.,  $u_1 = u_2 = u_3$ . As a demonstration, we provided the following pressure inputs to the system:

**Remark 1.1** (Additive manufacturing). *The soft robot is exclusively composed of a printable, flexible thermoplastic elastomer (Young’s modulus  $\leq 80$  MPa), which intrinsically promotes softness and dexterity. The elastomer material is developed explicitly for Selective Laser Sintering (SLS), a 3-Dimensional (3D) printing method that uses a laser to solidify powdered material. The main advantage of SLS printing over other techniques is that the printed parts are fully self-supported, which allows for complex and highly detailed structures. It should be mentioned that the layer-by-layer material deposition will introduce undesired anisotropic mechanical effects. To mitigate anisotropy, the bellows are printed orthogonal to the printing plane, thereby ensuring mechanical symmetry. For the majority of this work, the 3D-printed soft robot in Figure 1.1 will form the basis of the dynamical model. The 3D-model is made available at the open repository [39].*



**Figure 1.1.** (top) Soft robot manipulator with three parallel embedded pneumatic bellows. This manipulator changes its pose by inflation or deflation of the parallel pneumatic channels (<0.1 MPa). (bottom) (Differential) pressure signals applied on the internal bellows structures given by the input vector  $\mathbf{u} = (u_1 \ u_2, \ u_3)^\top$ , shown by the trajectories (—, —, —), respectively.

### 1.3 Generalized models for soft manipulators

As mentioned previously, soft robots are composed of soft bodies that may be regarded as a continuum body with (theoretically) infinitely many degrees-of-freedom (DOF). In this section, we aim to derive a compact and computationally efficient model that envelops the continuous dynamics of a soft robot through a small set of generalized coordinates  $\mathbf{q} \in \mathcal{Q}$  and their respective generalized velocities  $\dot{\mathbf{q}} \in T_{\mathbf{q}}\mathcal{Q}$  with  $n$  the number of active joint variables. We base the modeling framework on the work of Mochiyama et al. (2003, [15]) who outlined a theoretical foundation for continuum manipulators. Their work is extended upon by including extensibility, serial-chaining of multiple soft-links, pneumatic actuation, and the introduction of nonlinear and time-dependent material behavior. Earlier modeling strategies addressing similar issues can be found in from Godage et al. (2016, [3, 4]), Della Santina et al. (2020, [18, 41, 42]), Renda et al. (2018, [20]), and Boyer et al. (2021, [29]). Leveraging from the aforementioned works, the continuous dynamics of a soft robot manipulator can be written in the familiar Lagrangian form:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{J}^T(\mathbf{q})\boldsymbol{\lambda} + \boldsymbol{\tau}(\mathbf{q}, \mathbf{u}), \quad (1.1)$$

$$\boldsymbol{\tau} = \mathbf{G}^T(\mathbf{q})\mathbf{u}. \quad (1.2)$$

where  $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n \times n}$  denotes the generalized inertia matrix,  $\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^n$  a vector of nonlinear state-dependent force contributions. The nonlinear state-dependent contributions possess a structures as follows  $\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{f}^e(\mathbf{q}) + \mathbf{f}^g(\mathbf{q})$ .

**Assumption 1.1** (Finiteness generalized inertia). The inertia matrix is a positive definite symmetric matrix that is bounded from both sides  $\lambda^- \preceq \mathbf{M}(\mathbf{q}) \preceq \lambda^+$  for all configurations  $\mathbf{q}$ .

**Assumption 1.2** (Passivity). For any velocity  $\dot{\mathbf{q}}$ , it holds that  $\dot{\mathbf{q}}^T (\dot{\mathbf{M}} - 2\mathbf{C}) \dot{\mathbf{q}} = 0$  – the so-called passivity condition for Lagrangian systems. If the condition holds, it can easily be shown that map  $\mathbf{u} \mapsto \dot{\mathbf{q}}$  is passive, which implies that there exist a constant  $\beta \geq 0$ , such that the energy produced by the system  $E^u$  bounded from below [37]:

$$E^u := \int_0^T \dot{\mathbf{q}}^T(\tau) \mathbf{u}(\tau) d\tau > -\beta \quad \forall T > 0. \quad (1.3)$$

**Assumption 1.3** (Under-actuation). In many cases, a soft robot that falls under the category hyper-redundant is also intrinsically under-actuated. Mathematically, under-actuation is defined as follows. A second-order system  $\ddot{\mathbf{q}} = \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u}, t)$  is fully-actuated if, for any time  $t$  and state  $(\mathbf{q}, \dot{\mathbf{q}})$ , the flow map  $\mathbf{f}$  is surjective. In laymen's terms, for any acceleration  $\ddot{\mathbf{q}}$  there is exists a unique input  $\mathbf{u}$  that

produces such response. Otherwise, the system is under-actuated. Given the control affine structure in (1.2), the system is under-actuated if exist configurations  $\mathbf{q} \in \mathcal{Q}$  such that  $\text{rank}(\mathbf{G}(\mathbf{q})) < \dim(\mathbf{q})$ . Let it be clear that fully-actuated systems are dramatically easier to control than underactuated systems. However, for sake of simplicity at this stage, we assume the actuation matrix to be full rank and time-invariant, *i.e.*,  $\mathbf{G}(\mathbf{q}) \equiv \mathbf{G}$ . Under-actuation will be treated in Chapter ??.

In this chapter, a similar modeling framework is adopted to [15]; however, we propose an extension to incorporate FEM-driven data to more accurately reflect the underlying continuum mechanics – in particular hyper-elasticity. We also propose a numerical scheme that significantly accelerates the computation of the continuous dynamics.

### 1.3.1 Curve kinematics

To represent the hyper-flexible configuration of the soft robot, let us consider a smooth spatial curve that passes through the geometric center of the continuously deformable body, as shown in Figure 1.2. In literature, this curve is called the '*backbone curve*' as it simplifies the three-dimensional deformation imposed by distributed forces acting on the elastic body. The arc-length of the backbone corresponds to the extensible length of the soft robot denoted by the variable  $l(t)$  which we assume bounded  $l_- \leq l \leq l_+$ , and let  $L$  be a constant denoting the total unstressed length of the soft robot. Next, let us introduce a spatial variable  $\sigma \in \mathbb{X}$  that belongs to the one-dimensional material domain of the backbone curve, *i.e.*,  $\mathbb{X} = [0, L]$ . Let it be clear that the spatial variable  $\sigma$  represents the arc-length of a material coordinate along with the undeformed material domain of the soft robot manipulator.

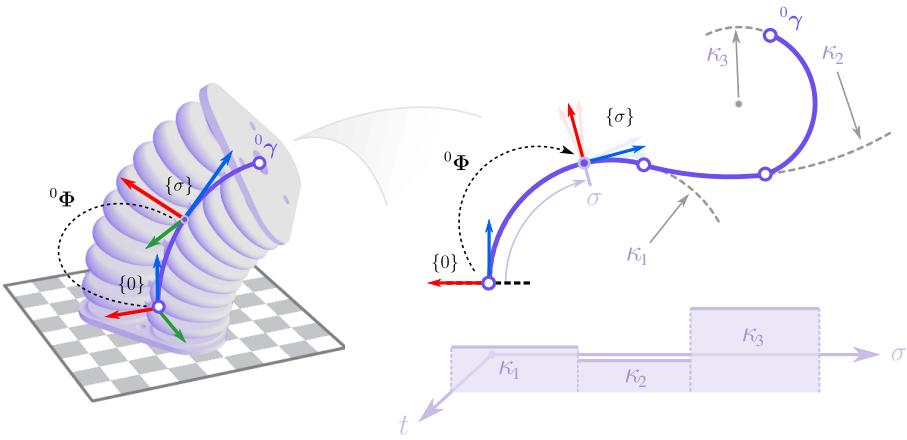
Given each material coordinate, we wish to find a suitable low-dimensional joint representation  $\mathbf{q}(t)$  such that the position vector  ${}^0\gamma$  anywhere on the continuous backbone can be written as a mapping from generalized coordinates and space into Euclidean space  $\mathbb{R}^3$ :

$${}^0\gamma : \mathbb{X} \times \mathcal{Q}(\mathbb{T}) \rightarrow \mathbb{R}^3; \quad (1.4)$$

and similarly the rotation matrix  ${}^0\Phi(\sigma, \mathbf{q})$  by a mapping from the generalized coordinates and space into  $\text{SO}(3)$ :

$${}^0\Phi : \mathbb{X} \times \mathcal{Q}(\mathbb{T}) \rightarrow \text{SO}(3), \quad (1.5)$$

where  $\text{SO}(3)$  denotes the special orthogonal group for rotations about the origin of  $\mathbb{R}^3$ ,  $n = \dim(\mathbf{q})$  the state dimension, and the manifold  $\mathcal{Q}$  is an embedding of temporal coordinates. We provided a more detail description on the rotation group  $\text{SO}(3)$  in Appendix ???. For sake of brevity, we drop the superscript that indicates



**Figure 1.2.** Schematic representation of the Piece-wise Constant Curvature model (PCC) for general soft robotic system, given by a parameterized curve  ${}^0\gamma : \mathbb{X} \times \mathbb{T} \rightarrow \mathbb{R}^3$  and orientation matrix  ${}^0\Phi : \mathbb{X} \times \mathbb{T} \rightarrow \text{SO}(3)$ . The frame  $\{\sigma\}$  is rigidly attached to  ${}^0\gamma$  such that variation of  $\sigma$  give insight into the differential geoemetry of the curve.

the frame of reference, *i.e.*,  ${}^0\Phi = \Phi$  and  ${}^0\gamma = \gamma$ . Under this notion, the position vectors  ${}^0\gamma(0, \mathbf{q})$  and  ${}^0\gamma(L, \mathbf{q})$  relate to the base and the end-effector of the soft robot, respectively. Please note that left-sided superscript are used to indicate the frame of reference. The set of all points on the backbone  $\mathcal{C} = \{\gamma \in \mathbb{R}^3 \mid \sigma \in \mathbb{X}\}$  draws a possible spatial configuration of the soft robot given a time instance  $t \in \mathbb{T}$  on a finite horizon  $\mathbb{T} = [0, T]$ .

**Definition 1.1** (Piece-wise Constant Curvature). Despite the inherent flexibility in soft robotics, it is sometimes sufficient to express the kinematics according to the Piecewise Constant Curvature (PCC or CC) condition. Mathematically, it implies that the curvature of the continuous body satisfies  $\kappa(\sigma_1, \mathbf{q}) = \kappa(\sigma_2, \mathbf{q})$  for a neighboring region of points  $\sigma_1, \sigma_2 \subseteq \mathbb{X}$ . As a result, this condition allows us to describe the full forward kinematics with a significantly reduced set of generalized coordinates, mitigating kinematic complexity in the model. Numerous works employ PCC models [1, 4, 17, 19, 21, 41], and depending on the elasticity, the PCC condition has been proven to be consistent for various soft robotic systems.

Following this Constant Curvature (CC) description, let us assign a coordinate frame that twists minimally along the backbone – formally called the ‘*Bishop frame*’ (see, [43]) – parametrized by the following generalized coordinate vector:

$$\mathbf{q} := (\varepsilon \quad \kappa_x \quad \kappa_y)^\top \in \mathcal{Q}, \quad (1.6)$$

where  $\varepsilon_- \leq \varepsilon \leq \varepsilon_+$  is the elongation strain, and  $\kappa_x, \kappa_y \in \mathbb{R}$  are the curvatures or angular strains in  $x$ - $z$  and  $y$ - $z$  plane, respectively; and  $\mathcal{Q} \subset \mathbb{R}^3$  is an admissible space on which  $\mathbf{q}$  evolves. Also, we denote the total curvature by  $\kappa = \langle \kappa_x, \kappa_y \rangle$  together with the curvature angle  $\phi = \text{atan2}(\kappa_y, \kappa_x)$ . It is worth mentioning that the joint description above is somewhat related to Renda. et al. (2018, [20]) who proposed a Piece-wise Constant Strain (PCS) parametrization with the exception of including the twist along the tangent.

By exploring the differential geometry of the smooth backbone curve similar to Mochiyama et al. (2003, [15]), we can write the position vector  $\boldsymbol{\gamma}(\sigma, \mathbf{q})$  and the orientation matrix  $\Phi(\sigma, \mathbf{q})$  for each material point  $\sigma$  along the smooth backbone as a differential equality of the form:

$$\frac{\partial \Phi}{\partial \sigma}(\sigma, \mathbf{q}) = \Phi(\sigma, \mathbf{q}) \boldsymbol{\Gamma}^\times(\sigma, \mathbf{q}) \quad (1.7)$$

$$\frac{\partial \boldsymbol{\gamma}}{\partial \sigma}(\sigma, \mathbf{q}) = \Phi(\sigma, \mathbf{q}) \mathbf{U}(\sigma, \mathbf{q}), \quad (1.8)$$

where  $\boldsymbol{\Gamma}^\times \in \text{SO}(3)$  is a skew-symmetric matrix composed of the entries of the vector  $\boldsymbol{\Gamma} \in \mathbb{R}^3$ , and  $\mathbf{U} \in \mathbb{R}^3$  a vector representing the tangent along the extensible backbone. For readers familiar with the Lie group theory, the operator  $[\cdot]^\times$  represents the isomorphism between the Lie algebra  $\text{so}(3)$  and  $\mathbb{R}^3$ . The vectors  $\boldsymbol{\Gamma}$  and  $\mathbf{U}$  are vectors that define the differential geometry of the backbone [15] which are unique entries that live in the tangent space of the rigid-body transformation group, i.e.,  $T_{\text{SE}(3)}$ . Given the Bishop parametrization as described by (1.6) and assuming the Constant-Strain (CC) condition, these geometric entities yield

$$\boldsymbol{\Gamma}(\sigma, t) \cong \boldsymbol{\Gamma}(\sigma, \mathbf{q}(t)) \xrightarrow{\text{CC condition}} \boldsymbol{\Gamma}(\mathbf{q}) = \begin{pmatrix} -\kappa_y \\ \kappa_x \\ 0 \end{pmatrix}, \quad (1.9)$$

$$\mathbf{U}(\sigma, t) \cong \mathbf{U}(\sigma, \mathbf{q}(t)) \xrightarrow{\text{CC condition}} \mathbf{U}(\mathbf{q}) = \begin{pmatrix} 0 \\ 0 \\ \varepsilon + 1 \end{pmatrix}, \quad (1.10)$$

Now, given an initial configuration of backbone's base, i.e.,  $\Phi(0, \mathbf{q}) = \Phi_0$  and  $\boldsymbol{\gamma}(0, \mathbf{q}) = \mathbf{0}_3$ , we can now solve for the position and orientation for each material coordinate  $\sigma$  along the backbone:

$$\Phi(\sigma, \mathbf{q}) = \Phi_0 \exp_{\text{SO}(3)}(\sigma \boldsymbol{\Gamma}^\times(\mathbf{q})), \quad (1.11)$$

$$\boldsymbol{\gamma}(\sigma, \mathbf{q}) = \int_0^\sigma {}^0\Phi(\eta, \mathbf{q}) \mathbf{U}(\mathbf{q}) d\eta, \quad (1.12)$$

where  $\exp_{\text{SO}(3)} : \text{so}(3) \rightarrow \text{SO}(3)$  is the exponential map. Luckily, there exist a compact expression for the exponential mapping related to the orthogonal group

of rotation matrices  $\text{SO}(3)$  called the '*Rodriguez formulas*'. Given the rotation angle  $\theta(\sigma, \mathbf{q}) := \kappa(\mathbf{q})\sigma$ , we can compactly rewrite the rotation matrix (1.11) in terms of  $\cos(\theta)$  and  $\sin(\theta)$  using the Rodriguez formulas as follows (see [44]):

$$\Phi(\theta) = \Phi_0 \left( \mathbf{I}_3 + \left[ \frac{\sin(\theta)}{\theta} \right] \boldsymbol{\Gamma}^\times + \left[ \frac{1 - \cos(\theta)}{\theta^2} \right] \boldsymbol{\Gamma}^\times \boldsymbol{\Gamma}^\times \right). \quad (1.13)$$

We wish to inform the reader that the closed-form solutions (1.11) and (1.12) represent the forward configuration kinematics of the backbone curve. To express the forward velocity kinematic, let  $\boldsymbol{\eta}(\sigma, \mathbf{q}, \dot{\mathbf{q}}) = (\boldsymbol{\omega}^\top, \mathbf{v}^\top)^\top \in \mathbb{R}^6 \cong \text{se}(3)$  be the aggregate of the angular velocity and linear velocity components relative to an inertial frame at  $\sigma$ , where the space  $\text{se}(3)$  denotes the Lie algebra of  $\text{SE}(3)$ . The velocity twist is computed by the following integration procedure:

$$\boldsymbol{\eta}(\sigma, \mathbf{q}, \dot{\mathbf{q}}) = \mathbf{Ad}_{\mathbf{g}(\sigma, \cdot)}^{-1} \int_0^\sigma \mathbf{Ad}_{\mathbf{g}(s, \cdot)} \mathbf{J}^* \dot{\mathbf{q}} \, ds =: \mathbf{J}(\mathbf{q}, \sigma) \dot{\mathbf{q}}, \quad (1.14)$$

where  $\mathbf{Ad}_g : \text{SE}(3) \rightarrow \mathbb{R}^{6 \times 6}$  denotes the adjoint transformation matrix regarding the rigid body transformation  $\mathbf{g} \in \text{SE}(3)$  that maps local velocities (i.e., twist) to a frame located at  $\sigma$ , and  $\mathbf{J}^* : \mathcal{Q} \rightarrow T_q \mathcal{Q}$  the joint-axis matrix that relates the DOFs to the geometric deformations. Let it be clear that the joint-axis matrix is naturally constant for a soft segment modeled with the Constant-Strain (CS) assumption. We will later relax this assumption in Chapter ???. Nevertheless here, the joint-axis matrix for an extensible and bendable CS segment parametrized by the Bishop parameters is given by

$$\mathbf{J}^* := \begin{pmatrix} \partial \boldsymbol{\Gamma}^\top & \partial \mathbf{U}^\top \\ \frac{\partial}{\partial \mathbf{q}} & \frac{\partial}{\partial \mathbf{q}} \end{pmatrix}^\top = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^\top. \quad (1.15)$$

Although we based the forward kinematics on the work of Mochiyama et al.[15], the derived expression for the velocity twist in (1.14) is analogous to the work of Renda et al. (2018, 2020; [20, 30]), and Boyer et al. (2010, 2021; [28, 29]). Please also note that (1.14) gives rise to the geometric manipulator Jacobian  $\mathbf{J}(\sigma, \mathbf{q})$  that defines the mapping from joint velocities to the velocity twist for a point  $\sigma$  on the elastic body.

In continuation, let us also introduce the acceleration twist [15, 20, 29] – obtained through time differentiation of (1.14):

$$\begin{aligned} \ddot{\boldsymbol{\eta}}(\sigma, \mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) &= \mathbf{J} \ddot{\mathbf{q}} + \mathbf{Ad}_{\mathbf{g}(\cdot, \sigma)}^{-1} \int_0^\sigma \mathbf{Ad}_{\mathbf{g}(s, \cdot)} \mathbf{ad}_{\boldsymbol{\eta}(s, \cdot, \cdot)} \mathbf{J}^* \dot{\mathbf{q}} \, ds \\ &:= \mathbf{J}(\sigma, \mathbf{q}) \ddot{\mathbf{q}} + \dot{\mathbf{J}}(\sigma, \mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}}, \end{aligned} \quad (1.16)$$

where  $\text{ad}_\eta : \mathbb{R}^6 \rightarrow \mathbb{R}^{6 \times 6}$  denotes the adjoint transformation regarding the velocity twist  $\eta^\wedge \in \text{se}(3)$ . The reader is referred to Appendix ?? for more detailed expressions on the adjoint transformations.

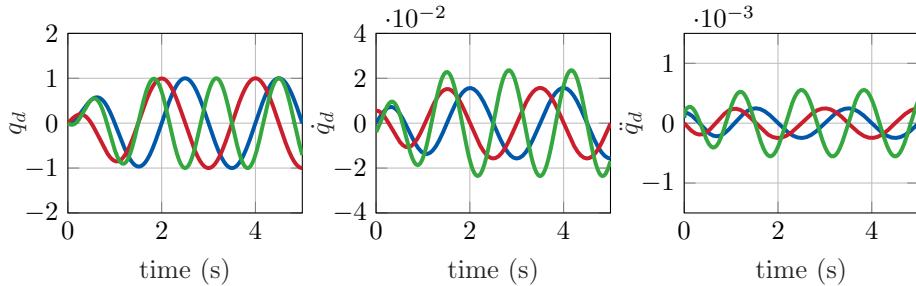
**Remark 1.2** (Numerical instability near zero-curvature). *In many of the PCC modeling literate there are mentions of a singularity point when the soft robot  $\kappa \rightarrow 0$ , stating that the linear velocities  $\mathbf{v} := [\eta]_3$  are not well-defined and thus may be unbounded. However, this is perhaps a common misconception in literate is mainly caused by numerical instability. To illustrate, consider the inextensible planar case:  $\varepsilon = \kappa_y = 0$  and  $\kappa = \kappa_x$ . Hence, by solving the forward kinematics for the position vector  $\gamma(\sigma, \kappa)$ , and taking the approaching zero from the positive domain  $\kappa^+ \rightarrow 0$ , we see that*

$$\lim_{\kappa \rightarrow 0^+} \gamma(\sigma, \kappa) = \left( \frac{1 - \cos(\sigma\kappa)}{\kappa}, \quad 0, \quad \frac{\sin(\sigma\kappa)}{\kappa} \right)^\top = (0 \quad 0 \quad \sigma)^\top, \quad (1.17)$$

which is clearly well-defined. Since the position vector  $\gamma$  is continuously differentiable when approaching the origin from both sides  $\kappa \rightarrow 0^+$  and  $\kappa \rightarrow 0^-$ , it follows that  $\dot{\gamma}$  must be bounded for all  $\kappa \in \mathcal{Q}$ . We can simply check this by investigating the behavior of the linear-velocity portion of the geometric Jacobian near zero-curvature, which yields

$$\begin{aligned} \lim_{\kappa \rightarrow 0^+} [\mathbf{J}]_3(\sigma, \kappa) &= \left( \frac{\sigma\kappa \sin(\sigma\kappa) + 1 - \cos(\sigma\kappa)}{\kappa^2}, \quad 0, \quad \frac{\sigma\kappa \cos(\sigma\kappa) - \sin(\sigma\kappa)}{\kappa^2} \right)^\top \\ &= (\sigma^2 \quad 0 \quad 0)^\top. \end{aligned} \quad (1.18)$$

Again, the expression is well-defined. Consequently, the magnitude of the linear velocity of the end-effector reads simply  $\|\dot{\gamma}(L, \dot{\kappa})\| = L^2 \dot{\kappa} = L\omega_1$  with  $\omega_1$  the angular velocity at the tip. This naturally poses an ambiguity on the origin of this kinematic singularity. The problem is, however, of numerical origin when considering the zero-division. To make matters worse, deriving analytical expressions for accelerations will contain similar expressions that are hard to stabilize numerically. To resolve this issue, we opt for a numerical approximation of the forward kinematics – namely, we employ an explicit forward integration scheme (i.e., trapezoidal integration) to solve (1.7) and (1.8).



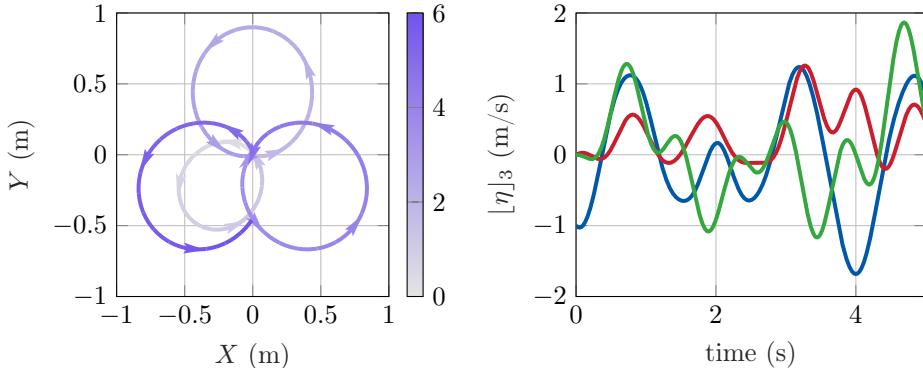
**Figure 1.3.** The time evolution of the predefined geometric strain parameters of the Piece-wise Constant curvature model  $\mathbf{q}_d \rightarrow (\varepsilon, \kappa_x, \kappa_y)^\top$  and their corresponding time-derivatives  $\dot{\mathbf{q}}_d$  and  $\ddot{\mathbf{q}}_d$ , given by the (constant) elongation  $\varepsilon$  (—), and the (constant) curvatures  $\kappa_x$  (—) and  $\kappa_y$  (—).

**Example 1.1** (Kinematic behavior of PCC segment). As an illustrative example, we provide simulation of the forward kinematics for a single PCC segment given predefined trajectory on the geometric joint variables  $\mathbf{q}(t) \equiv \mathbf{q}_d(t)$ ,  $\dot{\mathbf{q}}(t) \equiv \dot{\mathbf{q}}_d(t)$  and  $\ddot{\mathbf{q}}(t) \equiv \ddot{\mathbf{q}}_d(t)$ . Let us consider the following reference trajectory for the 3-DOF soft manipulator:

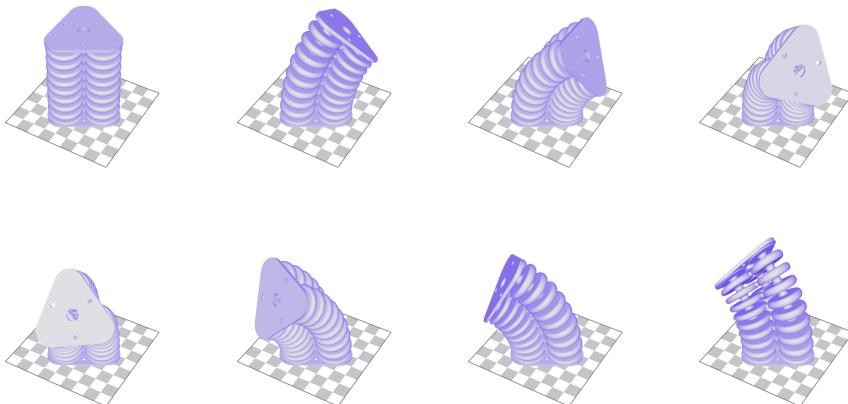
$$\mathbf{q}_d(t) = \text{erf}(t) \cdot \left( \varepsilon_0 \sin(\omega t) \quad \kappa_0 \cos(\omega t) \quad \kappa_0 \sin\left(\frac{3}{2}\omega t - \frac{\pi}{4}\right) \right)^\top,$$

where  $\text{erf}(t) := \frac{2}{\pi} \int_0^t \exp(-\tau^2) d\tau$  is referred to as the error function. Note that these are smooth functions such that reference velocity  $\dot{\mathbf{q}}_d$  and reference acceleration  $\ddot{\mathbf{q}}_d$  exist and are bounded. The reference signals for the geometric strain of the soft robot are shown in Figure 1.3. Let it be clear that the reference  $\mathbf{q}_d$  has been carefully selected to ensure it passes the line  $\kappa_x = \kappa_y = 0$  on the configuration space  $\mathcal{Q}$ , *i.e.*, the numerical instability point for near-zero curvature..

Then, by inject the reference into the kinematic relations described by the expressions (1.7), (1.8), (1.14), and (1.16), we obtain a (close) approximation of forward kinematics as shown in Figure 1.4. Furthermore, we provided a 3D-rendering of the soft robot subjected to the reference  $\mathbf{q}_d$  in Figure 1.5. Now, two key observations can be made. First, although a simple harmonic trajectory is used, the resulting trajectory of the end-effector as shown in Figure 1.4 is rather complex. This perhaps stresses the importance of inverse kinematic solver that can be used for task-space control. Second, although we pass the point of numerical instability for  $\kappa \rightarrow 0$ , we see that the velocity solutions are smooth and bounded at these instances. This result shows our approach does not suffer from the near-zero curvature instabilities that are notoriously mentioned in [1, 18], yet at the cost of providing an analytical expression.



**Figure 1.4.** The forward kinematics related to the prescribed reference  $\mathbf{q}_d(t)$ , where we show the planar displacements (—) and the linear velocities  $\mathbf{v} = [\boldsymbol{\eta}]_3 = (v_1, v_2, v_3)^\top$ , shown as (—), (—), (—), respectively. A key observation here, is that our numerical approach prevents the common numerical instability for near-zero curvatures, *i.e.*,  $\kappa \rightarrow 0$ , since the linear velocities are bounded and continuous even for  $\kappa(\mathbf{q}) = 0$ .



**Figure 1.5.** Three-dimensional deformation of the three-bellow soft robot manipulator using the PCC model subjected to the prescribed reference  $\mathbf{q}_d(t)$ .

### 1.3.2 Curve dynamics using Euler-Lagrange

Given the forward kinematics in (1.11), (1.12), (1.14) and (1.16), we can shift our attention to formulating the finite-dimensional dynamics of the soft robot. Our goal here is to write the spatio-temporal dynamics of the hyper-elastic soft robot as a second-order ODE into the Lagrangian form:

$$\frac{d}{dt} (\nabla_{\dot{\mathbf{q}}} \mathcal{L}) - \nabla_{\mathbf{q}} \mathcal{L} = \mathbf{Q}^{\text{nc}}, \quad (1.19)$$

where the mathematical operator  $\nabla_{\mathbf{x}}(\cdot) := \partial(\cdot)^{\top}/\partial\mathbf{x}$  denotes the gradient w.r.t. to a vector field  $\mathbf{x}$ ,  $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})^{\top} := \mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) - \mathcal{U}(\mathbf{q})$  the Lagrangian function,  $\mathcal{K} \in \mathbb{R}_{\geq 0}$  and  $\mathcal{U} \in \mathbb{R}$  the kinetic and potential energy, respectively; and  $\mathbf{Q}^{\text{nc}} \in \mathbb{R}^n$  a vector of generalized non-conservative forces. To apply the Lagrangian formalism to a continuum dynamical system, regard an infinitesimal slice of the continuum body for each material coordinate  $\sigma$  along the backbone curve. Given this notion, we embody this infinitesimal slice with an inertia tensor  $\mathbf{M} = \text{blkdiag}(\rho_0 \mathbf{I}, \mathcal{J}_0)$  with  $\rho_0 = m/L$  the line-density and  $\mathcal{J}_0 \in \text{so}^*(3) \times \text{so}(3)$  a symmetric tensor related to the second moment of inertia of infinitesimal slice at  $\sigma$ .

The kinetic energy can be obtained through spatial integration of its respective kinetic energy densities [15, 21, 28], i.e.,  $\mathcal{T} = \frac{1}{2} \boldsymbol{\eta}^{\top} \mathbf{M} \boldsymbol{\eta}$ :

$$\begin{aligned} \mathcal{T}(\mathbf{q}, \dot{\mathbf{q}}) &= \frac{1}{2} \int_{\mathbb{X}} \boldsymbol{\eta}(\sigma, \mathbf{q}, \dot{\mathbf{q}})^{\top} \mathbf{M} \boldsymbol{\eta}(\sigma, \mathbf{q}, \dot{\mathbf{q}}) d\sigma, \\ &= \frac{1}{2} \dot{\mathbf{q}}^{\top} \left[ \int_{\mathbb{X}} \mathbf{J}(\sigma, \mathbf{q})^{\top} \mathbf{M} \mathbf{J}(\sigma, \mathbf{q}) d\sigma \right] \dot{\mathbf{q}} := \frac{1}{2} \dot{\mathbf{q}}^{\top} \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}}. \end{aligned} \quad (1.20)$$

**Remark 1.3** (Generalized inertia for zero-curvature). *Given the form of the generalized inertia matrix (1.20), and assuming  $\mathbf{J}(\cdot, \mathbf{q})$  to be full-rank for all  $\mathbf{q} \in \mathcal{Q}$  we can easily show that  $\lambda^- \preceq \mathbf{M}(\mathbf{q}) \preceq \lambda^+ < \infty$  with  $\lambda^-$  and  $\lambda^+$  positive constants.*

Note that expression for the kinetic energy naturally gives rise to the generalized inertia matrix  $\mathbf{M}(\mathbf{q})$  of the Lagrangian model. By substitution of the kinetic energy into the Euler-Lagrange equation (1.19), we find  $\mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \mathbf{q}$  where  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$  denotes the Coriolis matrix. Instead of computing the Coriolis matrix through the conventional Christoffel symbols [45], we adopt a computational scheme by Garofalo et al. (2013, [46]) used for serial-chain rigid manipulators, in which we replaced the finite summation of  $N$  rigid bodies by a spatial integration over the continuum domain  $\mathbb{X}$ :

$$\begin{aligned} \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) &= \int_{\mathbb{X}} \mathbf{J}(\sigma, \mathbf{q})^{\top} \overbrace{\left[ \mathbf{M} \text{ad}_{\boldsymbol{\eta}} - \text{ad}_{\boldsymbol{\eta}}^{\top} \mathbf{M} \right]}^{c_{\boldsymbol{\eta}}} \mathbf{J}(\sigma, \mathbf{q}) + \dots \\ &\quad \dots + \mathbf{J}(\sigma, \mathbf{q})^{\top} \mathbf{M} \dot{\mathbf{J}}(\sigma, \mathbf{q}, \dot{\mathbf{q}}) d\sigma, \end{aligned} \quad (1.21)$$

where  $\mathcal{C}_\eta = -\mathcal{C}_\eta^\top$  a skew-symmetric matrix. The computation above is slight different from existing literature [29, 30] to ensure that the matrix  $\dot{\mathbf{M}} - 2\mathbf{C}$  is skew-symmetric – the so-called '*passivity condition*' (see Assumption 1.3). The importance of this property will become apparent later in the energy-based controller design. Lastly, the potential energy is given by sum of gravitational potential energy and internal elastic potential, *i.e.*,  $\mathcal{U}(\mathbf{q}) = \mathcal{U}_g(\mathbf{q}) + \mathcal{U}_e(\mathbf{q})$ . Since gravitational potential energy density is given by  $\mathfrak{U}_g = -\rho_0 \gamma(\sigma, \mathbf{q}) \mathbf{a}_g$  with  $\mathbf{a}_g \in \mathbb{R}^3$  is a vector of body accelerations, the potential energy related to gravity is obtained by spatial integration of their respective energy densities:

$$\mathcal{U}_g(\mathbf{q}) = \int_{\mathbb{X}} \mathfrak{U}_g(\sigma, \mathbf{q}) d\sigma = -\rho_0 \int_{\mathbb{X}} \gamma(\sigma, \mathbf{q})^\top \mathbf{a}_g d\sigma. \quad (1.22)$$

To model the hyper-elastic nature, lets introduce two nonlinear stiffness functions for both stretching and bending, denoted by  $k_e : \mathbb{R} \mapsto \mathbb{R}_{>0}$  and  $k_b : \mathbb{R} \mapsto \mathbb{R}_{>0}$ , respectively. These functions allow us to describe a collective elastic behavior imposed by the hyper-elastic materials and the continuum-bodied deformation. It shall be clear that these entities are unique to the soft robot's geometry and soft material choice, and thus finding a suitable candidate model requires further analysis. Later, we will sculpt these nonlinear stiffness functions through Finite Element Methods (FEM). For now, we assume that these analytical nonlinear stiffness functions are known, and thus the (hyper)-elastic potential energy takes the form

$$\mathcal{U}_e(\mathbf{q}) = \int_0^\varepsilon k_e(s) s ds + \int_0^{\beta(\mathbf{q})} k_b(s) s ds, \quad (1.23)$$

where  $\varepsilon$  is the elongation strain, and  $\beta(\mathbf{q}) := \kappa L(\varepsilon + 1)$  is the bending angle with the total curvature of the segment  $\kappa(\mathbf{q}) = \|\kappa_x, \kappa_y\|_2$  (see Figure 1.2).

### 1.3.3 Overall dynamic model

Finally, by combining (1.19), (1.20), (1.21), (1.22), and (1.23), the continuum dynamics of the soft robot can be casted into the familiar closed form [18, 20, 29] similar to aforementioned model (1):

$$\mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \mathbf{f}_e(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{f}_g(\mathbf{q}) = \boldsymbol{\tau}(\mathbf{u}, \boldsymbol{\delta}), \quad (1.24)$$

where  $\mathbf{f}_e = \nabla_{\mathbf{q}} \mathcal{U}_e + \mathbf{R} \dot{\mathbf{q}}$  is a vector of generalized forces imposed by the deformation of the soft materials with  $\mathbf{R} \succ 0$  the Rayleigh damping matrix,  $\mathbf{f}_g = \nabla_{\mathbf{q}} \mathcal{U}_g$  a vector of generalized gravitational forces, and  $\mathbf{u}(t)$  the control input with the index  $m$  the number of pressure inputs. The generalized input vector is chosen of the form:  $\boldsymbol{\tau}(\mathbf{u}, \boldsymbol{\delta}) = \mathbf{G}\mathbf{u} + \boldsymbol{\delta}$  with  $\mathbf{G}\mathbf{u} : \mathbb{R}^m \rightarrow \mathbb{R}^n$  a mapping from the input space to the joint actuation space, and  $\boldsymbol{\delta}(t)$  an external disturbance (*e.g.*, unmodelled material uncertainties).

**Remark 1.4.** Given the context of manipulators, a possible disturbance  $\delta(t)$  could be an external mass applied to the tip of the soft robot. Given the kinematic relations in (1.14) and (1.16), one can describe the disturbance (modeled here as a point-mass located at  $L$ ) by a state-dependent vector:

$$\delta_m = m_\delta [\mathbf{J}(\cdot, L)]_3^\top \left( \mathbf{A} \mathbf{d}_{\mathbf{g}(\cdot, L)}^{-1} \gamma_g + [\dot{\mathbf{q}}(\cdot, L)]_3 \right), \quad (1.25)$$

where  $[\cdot]_3$  extracts the last three rows of a matrix or vector, and  $m_\delta > 0$  the applied mass to the end-effector. It is worth recalling that the acceleration twist can be computed through the geometric Jacobian and its time derivative, i.e.,  $\dot{\eta} = \mathbf{J}\ddot{\mathbf{q}} + \mathbf{J}\dot{\mathbf{q}}$ . Indeed, the PCC condition for a soft body can only accurately describe the true dynamics if external forces produced by mass  $m_\delta$  do not excessively exceed the intrinsic elastic balancing forces  $\mathbf{f}_e(\mathbf{q})$ . Alternatively, a soft body can be modeled using multiple PCC curves of smaller size, similar to standard Finite Element discretization.

The actuation mapping  $\mathbf{G}$  depends on the geometry, placement, and orientation of the (pneumatic) soft actuators. Since the pneumatic chambers are aligned parallel to the backbone curve and are equally spaced along the circumference, we propose the following ansatz:

$$\mathbf{G} \cong \begin{pmatrix} \alpha_\varepsilon & \dots & \alpha_\varepsilon \\ -\alpha_\kappa \cos(\phi_1) & \dots & -\alpha_\kappa \cos(\phi_m) \\ \alpha_\kappa \sin(\phi_1) & \dots & \alpha_\kappa \sin(\phi_m) \end{pmatrix}, \quad (1.26)$$

where  $\alpha_\varepsilon, \alpha_\kappa > 0$  are system parameters representing the effective transferal of differential pressure to joint forces, and  $\phi_i = (i-1) \cdot \frac{2\pi}{m}$  the angular inter-distance between the  $m$ -number of pneumatic bellows. Please note that the parameters  $\alpha_\varepsilon$  and  $\alpha_\kappa$  are dependent on the bellow area and radius from the bellow to the backbone curve.

## 1.4 Extension to multi-link systems

We previously expressed the position and velocity kinematics as explicit functions of the generalized coordinates (i.e., Bishop parameters) and their time-derivatives. This explicit dependency stems from the PCC conditions inferring the curvature is non-varying along the spatial domain  $[0, L]$ , i.e.,  $\kappa(\sigma, \mathbf{q}) = \kappa(\mathbf{q})$ . Although sufficient for some cases, the condition is generally restrictive, and to some extent inconvenient, since the inclusion of multiple links demands piece-wise integration of the kinematics (1.12), (1.13), (1.14), and (1.16). Rather than separation of integration, we can extend this PCC description by using piece-wise continuous spatial function to distinguishes multiple soft-bodied links along the continuous

body of the soft robot. The idea of parametrization through shapes functions has been explored earlier by Chirikjian et al. (1994, [16, 47]), and later by Boyer et al. (2021, [29]), Della Santina et al. (2020, [18]). A similar discontinuous shape function series was used by Berthet-Rayne et al. (2021, [48]) to pursue multi-body dynamics for growing continuum robots; and proposed by Chirikjian (1992, [47]) for hyper-redundant robots earlier.

Following the aforementioned works, let us parameterize the geometric vectors  $\mathbf{\Gamma}$  and  $\mathbf{U}$  for a  $N$ -link soft robot through the product of a basis of orthonormal functions  $\{\theta_i\}_{i=1}^N$  and the Bishop parametrization as follows

$$\mathbf{\Gamma}(\sigma, \mathbf{q}) \cong \sum_{i=1}^N \theta_i(\sigma) [\mathbf{J}^*]_3 \mathbf{z}_i, \quad (1.27)$$

$$\mathbf{U}(\sigma, \mathbf{q}) \cong \sum_{i=1}^N \theta_i(\sigma) [\mathbf{J}^*]_3 \mathbf{z}_i + \mathbf{U}^\circ, \quad (1.28)$$

where  $\mathbf{J}^*$  is the joint-axis matrix as in (1.15), the mathematical operators  $[\cdot]_3$  and  $[\cdot]_3$  extract the first or last three rows of a matrix, respectively;  $\tilde{q}_i$  the joint variables of the  $i$ -th link, and  $\theta_i : [0, L] \mapsto \{0, 1\}$  is a piece-wise continuous shape function, whose purpose is to be non-zero for a given interval on  $\mathbb{X}$ . The new generalized coordinate vector becomes the aggregate of all joint variables of the multi-body soft robotic system  $\mathbf{q} = (\mathbf{z}_1^\top, \mathbf{z}_2^\top, \dots, \mathbf{z}_N^\top)^\top$  with the vector  $\mathbf{z}_i = (\varepsilon_i, \kappa_{x,i}, \kappa_{y,i})^\top$  relating to the Bishop parametrization of the  $i$ -th link.

Given (1.27) and (1.28), we may now rewrite the velocity-twist as

$$\boldsymbol{\eta}(\sigma, \mathbf{q}, \dot{\mathbf{q}}) = \left[ \mathbf{Ad}_g^{-1} \int_0^\sigma \mathbf{Ad}_g \mathbf{J}^* \boldsymbol{\Theta}(s) ds \right] \dot{\mathbf{q}} := \mathbf{J}(\sigma, \mathbf{q}) \dot{\mathbf{q}} \quad (1.29)$$

where  $\boldsymbol{\Theta}(\sigma) = (\theta_1, \theta_2, \dots, \theta_n) \otimes \mathbf{I}_n$  is an unitary selection matrix derived from the basis of piece-wise continuous shape functions  $\{\theta_i\}_{i=1}^n$ . Substitution of the discontinuous variation of the geometric Jacobian in (1.29) into (1.20) leads to the dynamic model of a  $N$ -link soft robot manipulator in the Lagrangian form similar to (1.24).

**Example 1.2** (Piece-wise selection for two-link system). To reduce ambiguity on the selection matrix  $\boldsymbol{\Theta}(\sigma)$ , lets consider a spatial coordinate  $\sigma_2 \in [L_1, L_1 + L_2]$  that lies on the spatial interval of the second link. Consequently, the operation  $\boldsymbol{\Theta}(\sigma_2)\mathbf{q} = \mathbf{z}_2$  returns the corresponding joint variable of the second link. This selection of generalized coordinates follows analogously for other links along the serial-chain of the soft manipulator.

## 1.5 Accelerated computation of PDE-like systems using Lagrangian formulism

Due to the partial differential nature of soft robots, obtaining a closed-form expression for the projected Lagrangian model in (1.24) can become notoriously long and complex (especially for multi-link systems). The origin of this problem stems from the integrands of inertia matrix  $\mathbf{M}(\mathbf{q})$  in (1.20) and Coriolis forces  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$  in (1.21); which become highly nonlinear and therefore difficult to calculate a-priori. As a result, solving the forward dynamics using traditional solvers often deteriorates the real-time performance, and in turn its usability for closed-loop control. Inspired by Boyer et al. (2021, [29]) and Godage et al. (2016, [4]), instead of finding an exact solution to the dynamic entries  $\mathbf{M}(\mathbf{q})$ ,  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$  and  $\mathbf{f}_g(\mathbf{q})$ , let us introduce a similar reduced-order integration scheme that produces an approximate of the dynamic model (1.24). Yet, instead of using an inverse Newton-Euler algorithm (i.e., Featherstone) in which the Lagrangian entries are built column-wise, we propose an explicit integration scheme that efficiently computes all Lagrangian entities in parallel through a so-called Matrix-Differential Equation (MDE).

The idea here is to replace all necessary spatial integrations required for the Lagrangian entities with an equivalent Matrix-Differential Equation of the form:

$$\frac{\partial \mathbf{Z}}{\partial \sigma} = \mathbf{F}(\mathbf{Z}, \sigma), \quad (1.30)$$

where  $\mathbf{Z}(\cdot, \sigma)$  is a matrix-valued function composed of the necessary elements for the forward kinematics and forward dynamics, and  $\mathbf{F}(\mathbf{Z}, \sigma)$  a matrix-valued flow function that describes the spatial evolution of  $\mathbf{Z}$ . Then, by choosing the appropriate initial condition for  $\mathbf{Z}(\cdot, \sigma = 0) = \mathbf{Z}_0$  and numerically solving (1.30) over a finite horizon  $\mathbb{X}$ , we can retrieve an approximate of the Lagrangian model in (1.24) by extracting the necessary elements from the solution  $\mathbf{Z}(\cdot, L)$ .

Before describing the MDE, let us first introduce two intermediate matrices related to the computation of the manipulator Jacobian and its time-derivative, namely:

$$\frac{\partial \mathbf{B}_1}{\partial \sigma} = \mathbf{Ad}_{\mathbf{g}(\cdot, \sigma)} \mathbf{J}^* \boldsymbol{\Theta}(\sigma) \quad (1.31)$$

$$\frac{\partial \mathbf{B}_2}{\partial \sigma} = \mathbf{Ad}_{\mathbf{g}(\cdot, \sigma)} \mathbf{ad}_{\boldsymbol{\eta}(\cdot, \sigma)} \mathbf{J}^* \boldsymbol{\Theta}(\sigma) \quad (1.32)$$

such that they satisfy  $\mathbf{J}\dot{\mathbf{q}} = \mathbf{Ad}_{\mathbf{g}}^{-1} \mathbf{B}_1 \dot{\mathbf{q}}$  and  $\dot{\mathbf{J}}\dot{\mathbf{q}} = \mathbf{Ad}_{\mathbf{g}}^{-1} \mathbf{B}_2 \dot{\mathbf{q}}$ . Given the expressions above, we can now include a partial computation Jacobians into the MDE. By collecting all the differential relation for the forward kinematics (1.7), (1.8), (1.14) and forward dynamics (1.20), (1.21) and (1.22), we can assign a flow

function  $\mathbf{F} := \text{blkdiag}(\mathbf{F}_1, \mathbf{F}_2)$  composed of two matrices:

$$\mathbf{F}_1 = \begin{pmatrix} \Phi\Gamma^\times & \Phi\mathbf{U} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_3 \end{pmatrix} \quad \left| \quad \begin{array}{c} \mathbf{Ad}_g J^* S \\ \mathbf{Ad}_g \mathbf{ad}_\eta J^* S \end{array} \right. , \quad (1.33)$$

$$\mathbf{F}_2 = \begin{pmatrix} \frac{\partial \mathbf{M}}{\partial \sigma} & \frac{\partial \mathbf{C}}{\partial \sigma} & \frac{\partial \mathbf{f}_g}{\partial \sigma} \end{pmatrix}, \quad (1.34)$$

in which the differential form of the dynamic entities  $\mathbf{M}(\mathbf{q})$ ,  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ , and  $\mathbf{f}_g(\mathbf{q})$  of the Lagrangian model are given by

$$\frac{\partial \mathbf{M}}{\partial \sigma} = (\mathbf{Ad}_g^{-1} \mathbf{B}_1)^\top \mathcal{M}(\mathbf{Ad}_g^{-1} \mathbf{B}_1), \quad (1.35)$$

$$\frac{\partial \mathbf{C}}{\partial \sigma} = (\mathbf{Ad}_g^{-1} \mathbf{B}_1)^\top [\mathcal{C}_\eta(\mathbf{Ad}_g^{-1} \mathbf{B}_1) + \mathcal{M}(\mathbf{Ad}_g^{-1} \mathbf{B}_2)], \quad (1.36)$$

$$\frac{\partial \mathbf{f}_g}{\partial \sigma} = ([\mathbf{B}_1]_3)^\top \rho_0 \mathbf{a}_g, \quad (1.37)$$

We wish to stress that  $\mathbf{F}_1$  collects all elements related to the forward kinematics, whereas  $\mathbf{F}_2$  contains the dynamic entities related to the Lagrangian model. Following the spatial Matrix-Differential equation in (1.30) above, its solution will be a matrix  $\mathbf{Z} := \text{blkdiag}(\mathbf{Z}_1, \mathbf{Z}_2)$  composed of two state matrices  $\mathbf{Z}_1$  and  $\mathbf{Z}_2$ :

$$\mathbf{Z}_1(\sigma, \mathbf{q}, \dot{\mathbf{q}}) := \begin{pmatrix} \Phi & \gamma \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_3 \end{pmatrix} \quad \left| \quad \begin{array}{c} \mathbf{B}_1 \\ \mathbf{B}_2 \end{array} \right. , \quad (1.38)$$

$$\mathbf{Z}_2(\sigma, \mathbf{q}, \dot{\mathbf{q}}) := (\mathbf{M} \quad \mathbf{C} \quad \mathbf{f}_g), \quad (1.39)$$

Such a set of Matrix-Differentials as in (1.30) are not supported natively by standard ODE solvers. Therefore, an explicit second-order Runge-Kutta solver for MDEs is developed such that efficiently computes the evolution of the state matrix  $\mathbf{Z}$  along  $\mathbb{X} = [0, L]$ . The numerical solver is written in MATLAB 2021a and can be found in the public repository of the SOROTOKI toolkit (see [40]).

As for state trajectories along the temporal regime  $\mathbb{T} = [0, T]$ , an implicit trapezoidal integration scheme is proposed to solve the approximated continuum dynamics, which are generally less conservative on discretization to preserve numerical stability. Here implicit schemes are favored over the explicit scheme since a coarser time integration can significantly increase real-time performance. In addition, to further boost the performance of the temporal integration, a cost-effective approximation of the Hessian is introduced. For more detail, see Appendix ?? for more detail.

## 1.6 System identification

In this section, the nonlinear stiffness function for elongation and bending,  $k_e : \mathbb{R} \rightarrow \mathbb{R}_{>0}$  and  $k_b : \mathbb{R} \rightarrow \mathbb{R}_{>0}$ , respectively, are solidified such that the elastic deformation aligns with the physical system (see Figure 1.1). Numerous studies consider these stiffnesses to be linear, however, the presence of exotic materials and complicated structures would justify the modeling of nonlinear elastic behavior. Here, we extend these conservative material models and explore the nonlinear and time-varying regime.

### 1.6.1 Finite element method and hyper-elasticity

Generally, soft robots are operated by (differential) pressure to air channels embedded in the elastic body. If the applied pressure is sufficiently larger than the ambient pressure, the elastic body deforms to counteract the external forces – the critical point at which the external force overcome the internal elastic forces is proportional to the Young's modulus of the material. To enable efficient mobility, soft robots often explore of materials (or material composites) with a low Young's moduli, e.g., silicone elastomers. Unfortunately, large deformations of these rubber-like materials inherently lead to state-dependency in the mechanical compliance, and thus Hookean elasticity is no longer accurate – rendering them hyper-elastic. These hyper-elastic materials branch a whole new subfield in continuum mechanics. Although analytic descriptions exist, hyper-elasticity is generally treated numerically through Finite Element techniques [23–25] paired with a (nonlinear) continuum mechanics framework.

Many variations of constitutive models for hyper-elastic materials are available, including Saint Venant-Kirchhoff, Neo-Hookean, Mooney-Rivlin, Ogden, and Yeoh [49–51]. In Mustaza et al. (2019, [33]), a Yeoh constitutive model is explored to describe hyper-elastic material characteristics of a silicone-composite actuator. Duriez et al.(2013, [23]) and related works [24, 25] employ Neo-Hookean material models to enrich the nonlinear deformations in FEM-driven models. There are many different constitutive models available, each better suited to describe specific nonlinear elastic behavior. Constitutive material models are mathematical functions used to express the (nonlinear) relationship between stress and strain in terms of deformation. The geometrical deformation of a solid is described by the deformation gradient tensor:

$$\mathbf{F}(\mathbf{x}) := \mathbf{I}_3 + \nabla_{\mathbf{x}} \mathbf{d}(\mathbf{x}) \quad (1.40)$$

where  $\mathbf{x} \subseteq \mathbb{V}$  is a material point in  $\mathbb{R}^3$ , and  $\mathbf{U}(\mathbf{x}, t) \in \mathbb{R}^3$  is the displacement vector. For hyper-elastic materials, it is postulated that there exists a potential energy function  $\mathbf{F} \mapsto \Psi(\mathbf{F})$  that is a function of the strain tensors. This potential

function  $\Psi(\mathbf{F})$  is also referred to as strain-energy density function, which depends exclusively on the material deformation.

In this work, we regard the Saint Venant-Kirchhoff constitutive model for hyper-elasticity. The Saint Venant-Kirchhoff model is in many ways similar to linear elastic materials (*i.e.*, Hooke's law), however, it is an extension from linear deformations into the nonlinear regime. The strain-energy density function for the Saint Venant-Kirchhoff model is defined as

$$\Psi^{\text{SV}} := \frac{\lambda}{2} \text{trace}(\mathbf{\mathcal{E}})^2 + \mu \text{trace}(\mathbf{\mathcal{E}}^2), \quad (1.41)$$

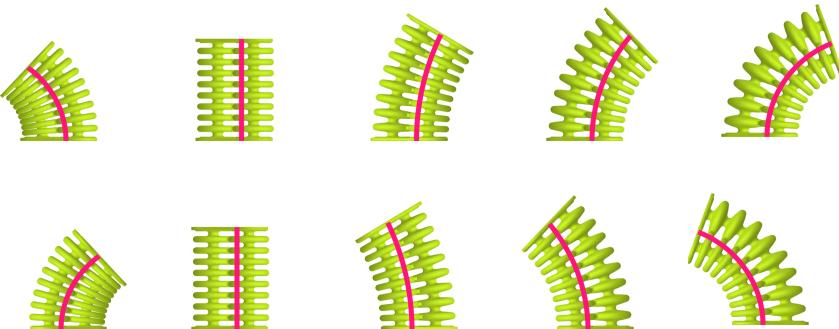
where  $\mathbf{\mathcal{E}} = \frac{1}{2}(\mathbf{F}^\top \mathbf{F} - \mathbf{I})$  the Green-Lagrange strain tensor,  $\text{trace}(\cdot)$  denotes the trace of a tensor, and  $\lambda > 0$  and  $\mu > 0$  are the Lamé parameters which arise from the strain-stress relationships of the elastic material. The relations for the Lamé parameters are given by

$$\lambda = \frac{\nu E_0}{(1 + \nu_0)(1 - 2\nu_0)} \text{ MPa}, \quad \mu = \frac{E_0}{2(1 + \nu_0)} \text{ MPa}; \quad (1.42)$$

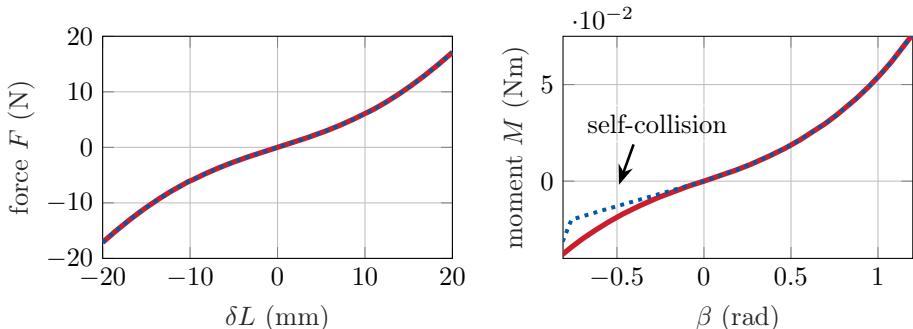
where  $E_0$  is the Young's modulus or elasticity modulus and  $\nu_0$  is a dimensionless constant denoting the Poisson ratio. It is worth mentioning that the Yeoh or Ogden model is more suitable for silicone elastomer materials that are conventional material models for soft robotics.

In order to invoke the constitutive material law (1.41) for three-dimensional solids, we explore the finite element method. We generated a finite element mesh of the soft robot manipulator in Figure 1.1. The finite element analysis has been carried out in Abaqus/CEA with variable time increments. Given preliminary uniaxial tension tests, the 3D-printed elastomer material is estimated to be linear isotropic described by the following Lamé parameters:  $E_0 = 80$  MPa,  $\nu_0 = 0.4$  (-). The Lamé parameters can be computed according to (1.42). Furthermore, tangential (frictionless) contact interaction is included in the numerical simulation to prevent self-intersection of the elastic body.

Two finite element simulations are performed to independently characterize the elongation and bending stiffness of the soft robot. Due to simplicity, we start with the elongation stiffness. Each embedded bellow is actuated simultaneously up to a quasi-static differential pressure  $-20 \text{ kPa} \leq u_1(t) = u_2(t) = u_3(t) \leq 30 \text{ kPa}$ . Due to the symmetry of soft actuators, the resulting deformation will be exclusively in axial-direction. The corresponding elongation strain of the soft robot can then be found by recovering the maximum vertical displacement of the nodal mesh, *i.e.*,  $\varepsilon = L^{-1} \max(U_z)$ . Secondly, the analysis of the bending stiffness is conducted by actuating a single bellow up to a quasi-static differential pressure  $-30 \text{ kPa} \leq u_1(t) \leq 80 \text{ kPa}$ , while  $u_2(t) = u_3(t) = 0 \text{ kPa}$ . To recover the bending angle of the elastic body, certain spatial coordinates of nodes close to the end effector



**Figure 1.6.** Highly-resolution finite element simulations of the soft robot manipulator (—) subjected to various input pressures  $-20 \text{ kPa} \leq \mathbf{u}_i(t) \leq 80 \text{ kPa}$ . To validate the PCC condition, an optimal backbone curve  $\gamma(\mathbf{q}^*)$  is shown (—) whose joint coordinates are recovered by solving the optimization problem (1.43).



**Figure 1.7.** Comparison of the (nonlinear) mechanical compliance between the proposed hyper-elastic model (—) and the finite element simulations (—). Notice the proposed stiffness model (1.45) does not capture the self-collision.

are tracked. Given their global coordinates, a constraint nonlinear optimization `fmincon.m` is used to recover optimal Bishop parameters  $\kappa$  and  $\varepsilon$  subjected to the kinematic relation in (1.13) and (1.12).

$$\mathbf{q}^* = \underset{\mathbf{q}}{\operatorname{argmin}} \left\| \log_{\text{SE}(3)} [\mathbf{g}^{-1}(L, \mathbf{q}) \mathbf{g}_L^{\text{FEM}}(\mathbf{u})] \right\|_2, \quad (1.43)$$

where  $\mathbf{g}_L^{\text{FEM}}$  is simply the corresponding end-effector configuration derived from the high-resolution finite element simulation given the input  $\mathbf{u}$ . To some extent, (1.43) can be viewed as an inverse kinematics optimization problem.

**Table 1.1.** Estimated hyper-elastic and visco-elastic material parameters for the study case soft robot

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$
$\alpha$	$2.23 \cdot 10^{+3}$	$1.74 \cdot 10^{+3}$	$-4.55 \cdot 10^{+2}$	$1.31 \cdot 10^{-3}$	$1.23 \cdot 10^{-2}$	$-2.29 \cdot 10^{-1}$
$\alpha_\lambda$	$3.21 \cdot 10^{+2}$	$5.22 \cdot 10^{-1}$	$26.4 \cdot 10^{+1}$	$1.82 \cdot 10^{-4}$	$26.4 \cdot 10^{+1}$	$1.82 \cdot 10^{-4}$

**Remark 1.5.** Although straightforward for constant curvature soft-links, the approach could be easily extended to non-constant curvature and multi-link cases. However, an additional term  $\mathcal{U}^* = \frac{1}{2}\mathbf{q}^\top \mathbf{K}\mathbf{q}$  with some positive definite matrix  $\mathbf{K} \succ 0$  is required to ensure the nonlinear regression inhibits over-fitting.

Following the PCC condition, the bending angle  $\beta$  can be calculated straightforwardly. Given these geometric curve parameters and the effective areas of the bellows, the applied elongation force and bending torque can be computed accordingly.

The numerical results are provided in Figure 1.7. In practice, these nonlinear strain relations can also be determined experimentally; however, the numerical methods have the beneficial convenience of several post-processing procedures and gravitation-free deformations. To support our previous claim concerning the consistency of the PCC condition, Figure 1.6 provides a few FEM snapshots results together with the optimal backbone curve from (1.12). It can be seen that the piece-wise constant curvature (PCC) condition, although a clear oversimplification of the true mechanics at hand, is remarkably consistent with the FEM simulations.

## 1.6.2 Hyper-elasticity on joint space

From the finite element results in Figure 1.6, the mathematical description for the nonlinear stiffness can be detailed further. However, a suitable candidate function must be chosen first to properly represent the hyper-elastic stress-strain relation. The stiffness function  $k_e(\xi)$  and  $k_b(\beta)$  have to satisfy the following properties.

**Assumption 1.4** (Elastic boundedness). There exists positive constants  $k^-$  and  $k^+$  such that  $k^- \leq k_e(\xi), k_b(\xi) \leq k^+$  for all possible strains  $\xi \in \mathbb{R}$ .

**Assumption 1.5** (Deformations reversibility). The stiffness functions  $k_e(\xi)$  and  $k_b(\xi)$  must have a global optimum (i.e., either a maximum or minimum) at their origin. As a result, the force produced by any deformation, given by  $\mathcal{F} = \int_0^\xi k(s)s ds$ , must be a monotonically increasing function where  $\mathcal{F} = 0$  for the origin.

Assumption 1.4 and 1.5 are necessary since they inhibit any elasto-plastic behavior, i.e., elastic bodies undergoing non-reversible deformation due to applied

forces. Then, consider the following elasticity models for the nonlinear (hyper-elastic) elongation and bending stiffness:

$$k_e(\mathbf{q}) = \alpha_1 + \alpha_2 (\tanh[\alpha_3 \varepsilon]^2 - 1), \quad (1.44)$$

$$k_b(\mathbf{q}) = \alpha_\phi(\mathbf{q}) \cdot [\alpha_4 + \alpha_5 (\tanh[\alpha_6 \beta]^2 - 1)], \quad (1.45)$$

where  $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6)^\top$  is a vector composed of the (possibly time-varying) stiffness parameters, and  $\alpha_\phi : \mathcal{Q} \rightarrow [1, \infty)$  a nonlinear correction term for asymmetry along the circumference of the radial-axis. Please note that these nonlinear functions possess a decomposable structure: a linear term and a nonlinear term that mimics strain-hardening or strain-softening. As for the asymmetric stiffness due to the layout of the pneumatic bellows, we propose the following ansatz:

**Assumption 1.6** (Stiffness variation under radial offset). Given the radial layout of the pneumatic bellows of the soft robot (see Figure 1.1), we assume that the nonlinear correction term for asymmetric radial stiffness along the circumference can be modeled by:

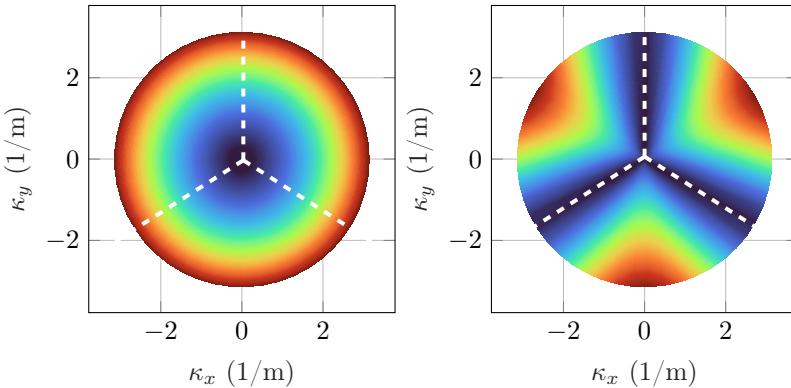
$$\alpha_\phi(\mathbf{q}) = \frac{1}{2}\beta [\sin(m\phi) + 1] + 1, \quad (1.46)$$

where  $m$  is the number of bellows, and  $\phi = \text{atan2}(\kappa_y, \kappa_x)$  the direction angle or heading. This stiffness correction term ensures the nonlinear bending stiffness becomes larger between bellows – as it causes simultaneous deformation of multiple bellows. Please note that  $\alpha_\phi(\mathbf{q}) \geq 1$  for all  $\mathbf{q} \in \mathcal{Q}$ .

To satisfy the aforementioned conditions, it should hold that  $\alpha_1 > \alpha_2$ ,  $\alpha_4 > \alpha_5$  and  $\alpha_{1,4} > 0$ . Using a weighted least-square optimization, the nonlinear stiffness parameter vector  $\boldsymbol{\alpha}$  can be identified. The estimated hyper-elastic material parameters are shown in Table 1.1. Furthermore, the weighted regression is biased towards positive strains, to better represent the deformation characteristics under positive pressurization. Theoretically, the self-contact interactions (as seen in Figure 1.6) can also be parameterized using a different set of (convex) non-zero polynomials, similar to the functions (1.44) and (1.45).

### 1.6.3 Visco-elastic creep

In material mechanics, the tendency of mechanical solids to move slowly under stress is called creep. Unlike ideal elastic materials, when polymeric materials are subjected to abrupt change in stress, the constitutive network of polymer chains reconfigure until the stress is evenly distributed. Therefore, let us introduce a new state vector  $\boldsymbol{\lambda}(t) \in \mathbb{R}^n$ , which contains the creep state variables. According to



**Figure 1.8.** Spatial representation of the nonlinear stiffness, *i.e.*, elongation stiffness (left) and bending stiffness (right), for an radially distributed sampling of the curvature joint space  $(\kappa_x, \kappa_y) \in (-\pi, \pi]$ . Note that the bending stiffness

Meyer et al. (2009, [49]), the Kelvin-Voigt model for creep is given by a first-order ordinary differential equation of the form

$$\dot{\boldsymbol{\lambda}} = - \begin{pmatrix} \alpha_{\lambda,1} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \alpha_{\lambda,2n-1} \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{pmatrix} - \begin{pmatrix} \alpha_{\lambda,2} \\ \vdots \\ \alpha_{\lambda,2n} \end{pmatrix} \dot{\boldsymbol{q}} \quad (1.47)$$

where the vector  $\boldsymbol{\alpha}_{\lambda} = (\alpha_{\lambda,1}, \alpha_{\lambda,2}, \dots, \alpha_{\lambda,2n})^T$  contains positive parameters that describe the visco-elastic material dynamics, and the state variables  $\lambda_i(t)$  with  $i \in \{1, 2, \dots, n\}$ , *i.e.*, the creep strains. Note that the state dimension of the creep strains is equivalent to the state dimension  $\dim(\boldsymbol{q})$ . Now, the dynamics of the visco-elastic creep can be intuitively included into (1.24) as an external disturbance  $\boldsymbol{\delta}(t)$ . Hence, consider the visco-elastic creep forces of the Kelvin-Voigt model to be characterized by

$$\boldsymbol{\delta}_c(\boldsymbol{\lambda}) = \boldsymbol{K}_{\lambda}^T \boldsymbol{\lambda}, \quad (1.48)$$

where  $\boldsymbol{K}_{\lambda}$  denotes the creep compliance matrix, which is a linear mapping from creep strains to creep forces. Since creeping strain are difficult, if not impossible, to distinguish from the elastic strains, the creeping parameters  $\boldsymbol{\alpha}_{\lambda}$  and the creep compliance  $\boldsymbol{K}_{\lambda}$  are empirically identified from experimental data (e.g., unforced oscillations) by using least-squares regressions. The estimated visco-elastic parameters are also found in Table 1.1. The linear damping parameters from the Rayleigh dampings matrix  $\boldsymbol{R}$  and the initial conditions for  $\boldsymbol{\lambda}(t_0)$  are identified similarly.

**Example 1.3** (Kelvin-Voigt creep dynamics). To highlight the dynamics of elastomer materials exhibiting creep, let us consider an rudimentary illustrative example of a 1-DOF mass-spring-damper system with a Kelvin-Voigt creep element. Let  $\mathbf{x} = (x_1, x_2, x_3)^\top = (\varepsilon, \dot{\varepsilon}, \lambda)^\top$  be the state vector composed of the elongation strain, elongation rate, and the creep strain, respectively. Then, the dynamics can be written in the familiar state space form as follows:

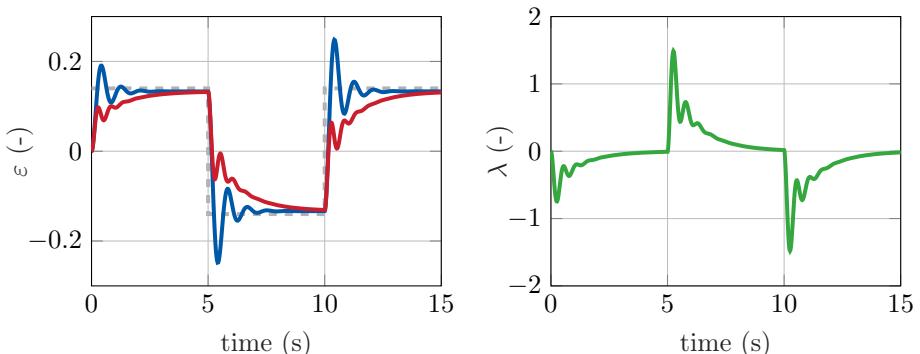
$$\dot{x}_1 = x_1, \quad (1.49)$$

$$\dot{x}_2 = \frac{1}{m} [-kx_1 - cx_2 + k_\lambda x_3 + u], \quad (1.50)$$

$$\dot{x}_3 = -\alpha_{\lambda,1}x_3 - \alpha_{\lambda,2}x_2 \quad (1.51)$$

where  $m, k, c > 0$  the mass, spring, damper coefficients; respectively,  $k_\lambda > 0$  the creep stiffness,  $\alpha_\lambda > 0$  the creep parameters, and  $u : \mathbb{T} \rightarrow \mathbb{R}$  an auxiliary input (*i.e.*, prescribed force). The following parameters are consider for this illustrative example:  $m = 0.1$  kg,  $k = 10$  Nm $^{-1}$ ,  $c = 0.1$  Nsm $^{-1}$ ,  $k_\lambda = \alpha_{\lambda,1} = 2$ , and  $\alpha_{\lambda,2} = 5$ . By choosing a smooth block signal as input  $u(t)$ , we obtained the simulation results as shown in Figure 1.9. Note that the simulation result presents an unmodified Hookean model (—), *i.e.*,  $k_\lambda = 0$ , and the Kelvin-Voigt variant (—).

Clearly, we see a difference between the two trajectories. The original Hookean model oscillates around the quasi-static equilibrium, whereas the Kelvin-Voigt variant slowly converges to the setpoint. Note that this is clearly different than over-damped response, as the Kelvin-Voigt model does allow for oscillations.



**Figure 1.9.** Simulation study of introducing visco-elastic Kelvin-Voigt dynamics to a mass-spring-damper system, where we show the evolution of the elongation  $\varepsilon$  subjected to an smooth block signal  $u(t)$  in (—) compared between the original Hookean model (—) and visco-elastic Hookean model (—). Also, the evolution of the visco-elastic creeping strain  $\lambda$  is shown in (—).

## 1.7 Adaptive control

As briefly discussed in the introduction, the dynamics model will be used as a control-oriented framework for model-based controllers applicable to soft robotics. In retrospect to previous model-based controllers, Della Santina et al. (2020, [18]) proposed a combination of feedforward and model-based feedback; yet, satisfying the passivity condition, more robustness approach could be acquired through energy-based controller (especially in the face of material uncertainties). Franco et al. (2020, [38]) proposed an adaptive energy-based controller but the underlying model (multi-link pendulum) is not rooted in a continuum description. Here, we wish to provide a mix of the control methodologies – an energy-based control approach for the continuous PCC model with an adaptive material law.

### 1.7.1 Passivity-based Adaptive Control

The continuous dynamics of the soft robotic manipulator are described by (1.24), where Lagrangian system matrices depend on physical parameters, e.g., mass, moments of inertia, stiffness, and viscosity. Within the context of robust control, these parameters often deviate from their true value. So merely an estimate of the system matrices  $\tilde{\mathbf{M}}(\mathbf{q})$ ,  $\tilde{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}})$ ,  $\tilde{\mathbf{f}}_g(\mathbf{q})$  and  $\tilde{\mathbf{f}}_e(\mathbf{q}, \dot{\mathbf{q}})$  can be acquired, where we denote  $\Delta(\cdot) = (\tilde{\cdot}) - (\cdot)$  as the difference between the true value and its estimate. The difference (or uncertainty) between true and estimated values is of particular relevance in soft robotics, where material properties play a significant role on both the statics and dynamics. Poor estimates of the material parameters could lead to instability in some model-based controllers if not considered carefully. Exploiting the passivity in Lagrangian models (see Appendix ??), we can derive passivity-based adaptive controller similar to the works of Slotine et al. (1988, [35]) and Ortega et al. (1998, [37]). The merit benefit of passivity-based control techniques is its robustness regarding parameter uncertainties and unmodelled dynamics. Passivity-based control is rooted in energy-shaping and damping injection techniques, leading to simple implementation yet effective means of stabilization.

Let  $\mathbf{q}_d(t) \in \mathcal{Q}$  be the desired trajectory of the soft robot together with its time-derivative  $\dot{\mathbf{q}}_d(t), \ddot{\mathbf{q}}_d(t) \in \mathbb{R}^n$ . Next, let  $\boldsymbol{\pi} \in \mathbb{R}^p$  be a vector containing all unknown values from a set of physical parameters, and the parametrization error  $\mathbf{e}_p := \tilde{\boldsymbol{\pi}} - \boldsymbol{\pi}$  in which the the vector  $\tilde{\boldsymbol{\pi}} \in \mathbb{R}^p$  denotes the parameter estimates. The control objective is given by finding an appropriate control input and update law such that  $\lim_{t \rightarrow \infty} \mathbf{q}(t) = \mathbf{q}_d(t)$  is achieved with good transient behavior. Assuming linearity in the parameters the linear parametrizability matrix of the soft robot's dynamics is given as follows

$$\mathbf{Y}(\cdot, \boldsymbol{\pi}) \mathbf{e}_p = \Delta \mathbf{M} \ddot{\mathbf{q}}_r + \Delta \mathbf{C} \dot{\mathbf{q}}_r + \Delta \mathbf{f}_g + \Delta \mathbf{f}_e + \Delta \boldsymbol{\delta}, \quad (1.52)$$

where  $\dot{\mathbf{q}}_r = \dot{\mathbf{q}}_d - \boldsymbol{\Lambda} \mathbf{e}$  is called the reference velocity vector,  $\boldsymbol{\Lambda} \in \mathbb{R}^{n \times n}$  a positive diagonal matrix, and  $\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}_r, \dot{\mathbf{q}}_r, \boldsymbol{\pi}) \in \mathbb{R}^{m \times n}$  is called the regressor matrix. Following the work of Slotine and Li (1988, [35]), the control law and adaptation law are given by

$$\tau = \widetilde{\mathbf{M}} \ddot{\mathbf{q}}_r + \widetilde{\mathbf{C}} \dot{\mathbf{q}}_r + \widetilde{\mathbf{f}}_g + \widetilde{\mathbf{f}}_e - \widetilde{\boldsymbol{\delta}} - \mathbf{K}_p \mathbf{e} - \mathbf{K}_d \mathbf{e}_r, \quad (1.53)$$

$$\dot{\boldsymbol{\pi}} = -\mathbf{K}_{\boldsymbol{\pi}} \mathbf{Y}^{\top} \mathbf{e}_r, \quad (1.54)$$

where  $\mathbf{e}_r := \dot{\mathbf{q}} - \dot{\mathbf{q}}_r = \dot{\mathbf{e}} + \boldsymbol{\Lambda} \mathbf{e}$ ,  $\mathbf{K}_p, \mathbf{K}_d \in \mathbb{R}^{n \times n}$  are controller gains, and  $\mathbf{K}_{\boldsymbol{\pi}} \in \mathbb{R}^{p \times p}$  is a positive definite matrix called the adaptation rate. It should be mentioned that the magnitude of adaptation rate does not affect the global stability of the system (if unmodelled dynamics are not excited); however, it sets the rate of adaptation, and accordingly the performance of the system.

**Remark 1.6** (Persistence of excitation). *Under the condition of persistent excitation, that is, for any instances  $t_1, t_2$  with  $t_1 \leq t_2$  there exists a positive constant  $\alpha$  such that  $\int_{t_1}^{t_2} \mathbf{Y}^{\top} \mathbf{Y} dt \preceq \alpha \mathbf{I}$ , it can be proven that the parameter estimates converge exponentially. The proof is provided by Morgan et al. (1977, [36]) through a fairly involved procedure.*

## 1.8 Numerical and experimental implementation

In this section, we will discuss the simulation results of the dynamic model (1.24), the passivity-based controller (1.53), and the adaptive law (1.54). To illustrate effectiveness and performance of the approach, we segregate our analysis into several study-cases of various complexity. First, focusing on the physical one-link soft robot in Figure 1.1 ( $N = 1$ ), we investigate the unforced system's equilibria and their corresponding stability. In continuation, we compare the simulated trajectories of the dynamical model with experimental data for natural oscillations, forced pneumatic inputs, and external loading conditions; where we also highlight contribution of the hyper-elastic FEM-driven material model. Second, to illustrate the flexibility and computational efficiency of the numerical framework, we extend the one-link model to a multi-link model with  $N = 6$  soft-bodied links.

The numerical solutions to the ordinary differential equations in (1.24) together with (1.53) and (1.54) are computed using the aforementioned MDE integration scheme which is developed in MATLAB, and the underlying code can be found at Caasenbrood et al. (2020, [52]). The software architecture is compactly written as Object-Oriented class labeled under `./src/Model.m` that enables a minimal programming interface to set-up various soft robotic simulation models easily. Additionally, all numerical examples that will be discussed in this section are made available under `./examples/paper` on the repository.

**Example 1.4** (Natural dynamics – One-link soft robot).

The following physical parameters are chosen for the soft robot: the mass  $m_0 = 17.3$  g, the relaxation length  $l_0 = 64.4$  mm. The material parameters for hyper-elasticity and visco-elasticity models are chosen identical to Table 1.1. For the additional viscous material behavior, the Rayleigh damping matrix and the creep compliance matrix are chosen as follow:

$$\begin{aligned} \mathbf{R} &= \begin{pmatrix} 0.01 & 0 & 0 \\ 0 & 1.05 \cdot 10^{-5} & 0 \\ 0 & 0 & 1.05 \cdot 10^{-5} \end{pmatrix}; \\ \mathbf{K}_\lambda &= \begin{pmatrix} 502.3 & 0 & 0 \\ 0 & 1.53 \cdot 10^{-2} & 0 \\ 0 & 0 & 1.53 \cdot 10^{-2} \end{pmatrix}. \end{aligned}$$

We stress that the values for the Rayleigh damping and creep compliance shown above are identified empirically through open-loop measurements, similar to the creep coefficient provided in Table 1.1.

First, we investigate the existence and the stability of the equilibria of the unforced system. If the system is at rest (i.e.,  $\dot{\mathbf{q}} = 0$ ,  $\ddot{\mathbf{q}} = 0$ ), then by definition there

are no conservative forces acting on the system. Thus, for any equilibrium point  $\mathbf{q}_0$  it holds that  $\nabla \mathcal{U}(\mathbf{q}^*) \equiv \mathbf{0}$ . If  $\mathcal{U}(\mathbf{q}^*) \equiv E_0$  is a local minimum, then the equilibrium is deemed stable. Any small disturbance will result in a new energy-state  $E_1$  and will consequently bring the system in motion. However, regarding  $E_0$  is a local minimum, the system will remain in a neighborhood of  $\mathbf{q}^*$  and eventually converge towards its nearest low-state energy  $E_0$ . If  $\mathcal{U}(\mathbf{q}^*) \equiv E_0$  is a local maximum, the equilibrium is deemed to be unstable, since there exist a configuration close to  $\mathbf{q}^*$  with a lower energy-state, *i.e.*,  $\mathcal{U}(\mathbf{q}^* + \delta\mathbf{q}) < E_1$ .

By analysis of the gradient of the potential energy function  $\nabla \mathcal{U}(\mathbf{q})$ , two unique equilibria can be found numerically. The potential function has a local maximum for  $\mathbf{q}_{\text{unstab}}^* = (-\frac{m_0 g}{L(\alpha_1 - \alpha_2)}, 0, 0)^\top$  which is unstable. To some extent, it is analogous to the unstable upwards equilibrium position related to the single-DOF pendulum system. For the stable equilibria, the bisection method was used to find the zero-crossing of  $\nabla \mathcal{U}(\mathbf{q})$ , where it was found that all stable solutions of the unforced system will tend to the following set:

$$\Omega_{\text{stab}} = \left\{ \mathbf{q} \in \mathcal{Q} : \varepsilon = \varepsilon^*, \kappa(\mathbf{q}) = \frac{\alpha^*}{\alpha_\phi(\mathbf{q})} \right\}$$

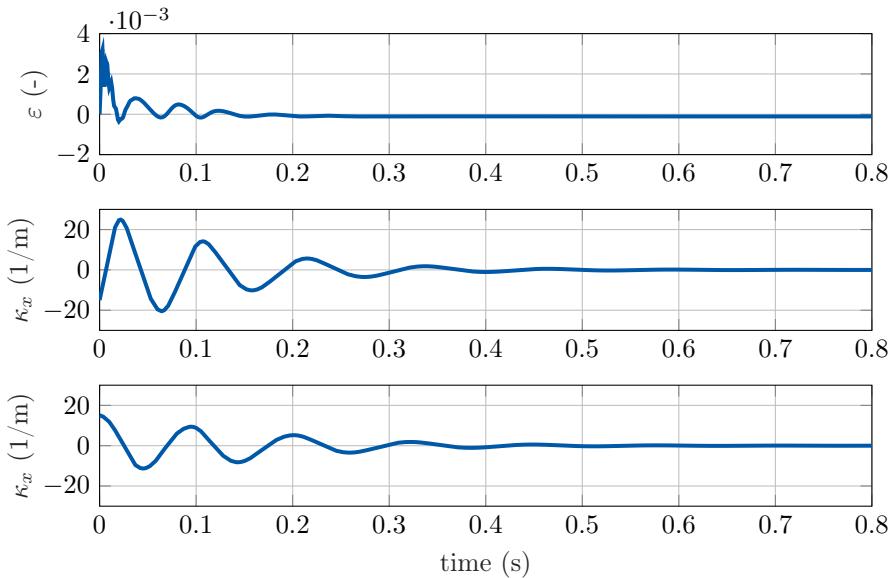
with  $\varepsilon^* = -0.0021$  and  $\alpha^* = 0.0174$ , which topologically equivalent to a ring. This set corresponds to the hanging position of the soft robot. It should be worth mentioning that the stable set of equilibria stems from the force balance between the internal elastic potential forces and the external gravitational potential forces, and thus any stiffness will lead to a stable set with a similar topology. By changing the base orientation of the soft manipulator (*i.e.*, by modifying  $\Phi_0$ ), both equilibria vanish and all state trajectories will tend to a global stable equilibrium. For fully reversing the orientation, this trivially leads to the stable equilibrium  $(\varepsilon, \kappa_x, \kappa_y) = (+\frac{m_0 g}{L(\alpha_1 - \alpha_2)}, 0, 0)$ . This phenomenon is referred to as local bifurcation, in which the change of parameter values alters the existence and stability of equilibria. This property might be interesting for soft robot manipulators with multiple soft-bodied links, as they are likely to be subjected to different gravitational loads.

To illustrate the unforced dynamics and the existence of stable equilibria, time-domain simulations of the dynamical model with nonzero initial conditions:

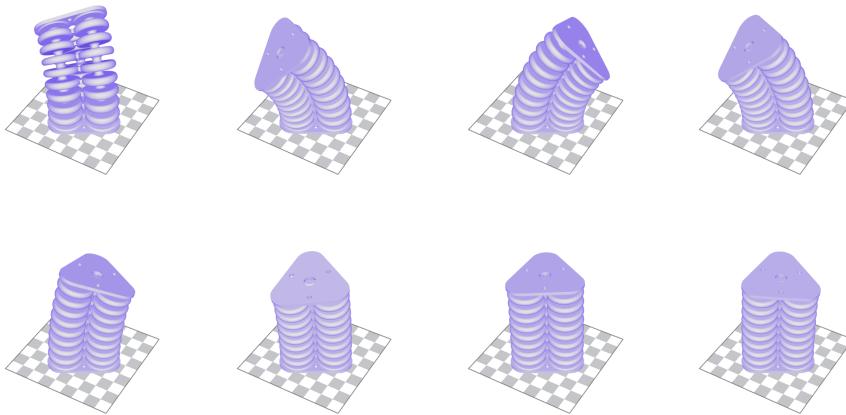
$$\mathbf{q}_0 = (0, -15, 15)^\top,$$

$$\dot{\mathbf{q}}_0 = (0, 2500, 0)^\top$$

Figure. 1.14 shows the state trajectories of the soft robot; whereas Figure ??b is provided to better illustrate the underlying dynamics and the trajectory of the end-effector. Besides the existence of stable solutions, the numerical simulations perfectly illustrate the coupled dynamics between the elongation and bending of the soft robot. Due to the difference in mechanical stiffness for elongation and



**Figure 1.10.** Three-dimensional volumetric evolution of the one-link soft robot model with initial conditions  $\mathbf{q}_0 = (0, -15, 15)^\top$  and  $\dot{\mathbf{q}}_0 = (0, 2500, 0)^\top$ . Notice that the soft robot oscillates about the set of stable equilibria  $\Omega_{\text{stab}}$ .



**Figure 1.11.** State trajectories of one-link soft robot model with initial conditions  $\mathbf{q}_0 = (0, -15, 15)^\top$  and  $\dot{\mathbf{q}}_0 = (0, 2500, 0)^\top$ . The figure shows the elongation strain  $\varepsilon$  and the curvatures  $\kappa_x$ ,  $\kappa_y$  in the  $xz$ -plane and  $yz$ -plane, respectively.

bending, we observe high-frequency and low-frequency oscillation for the elongation strain  $\varepsilon(t)$ , and we observe low-frequent oscillations for the curvatures  $\kappa_x(t)$

and  $\kappa_y(t)$ . Interestingly, the low-frequency oscillations are passed from the curvature dynamics to elongation dynamics; conversely, the dynamics of the elongation barely affect the curvatures. After sufficient time passes, the trajectories indeed tend to the set of stable equilibria  $\Omega$ .

### 1.8.1 Experimental platform

Before validating the one-link dynamic model, we detail the experimental setup and control platform of the soft robotic system. We would like to briefly emphasize that sensing is a challenging topic soft robotics, due large continuous deformations and the presence of sensors that might affect the system's dynamics. As such, a combination of integrated sensors were used to recover an estimate of the states  $\mathbf{q} \in \mathcal{Q}$ . First of all, we employed a 9-DOF inertial measurement unit (Bosch, BNO055) measures the angular displacement of the soft robot's end-effector (*i.e.*,  $\sigma = L$ ). Through on-board sensor fusion, the bending angle of the soft robot can be recovered, *i.e.*,  $\beta = \kappa l$ . Since the bending angle alone is not sufficient to decouple the curvature and elongation, additional sensing is required. Consequently, we used a vision sensor (CreateLab, PixyCam2) with an embedded processor (NXP LPC44330, 200 MHz) for optical tracking. A spherical optical marker is attached to the end-effector of the soft robot. Through trigonometry and the measured bending angle  $\beta$ , an estimate of the position vector  $\boldsymbol{\gamma}(L, \mathbf{q})$  can be recovered. Given the analytic expressions for the orientation and position in (1.11) and (??), an inverse Jacobian kinematic solver is employed to recover an estimate of the state vector  $\tilde{\mathbf{q}}_d$ . During each experimental trail, it was made sure the soft robotic body does not occlude the optical marker.

As for the pneumatic actuation, an array of proportional-pressure regulators (Festo, VEAB-B-D16) was used with an active pressure range of  $-0.1 \text{ MPa} < \mathbf{u}(t) \leq 0.1 \text{ MPa}$ , which simultaneously allow for pressure measurements. These measurements are fed into the (quasi-static) model to also recover a quasi-static estimate of the states  $\hat{\mathbf{q}}_s$ . Then, the dynamic estimates  $\tilde{\mathbf{q}}_{\text{dyn}}$  and the quasi-static pressure-based estimates  $\tilde{\mathbf{q}}_{\text{qs}}$  are fused using a complementary filter. The control and data acquisition are done using a Raspberry Pi 4 (2GB). The full experimental setup can be seen in Figure ??.

**Example 1.5** (Model validation – unforced, forced, and external loads). To validate the dynamic model, the solutions of the model are compared with measurements of the physical system in unforced, forced, and tip-load conditions. As such, the model validation is separated into three parts: *i*) unforced, *ii*) forced conditions, and *ii*) external tip-loads applied on the end-effector. We start with the unforced scenario, *i.e.*, no input is considered  $u_i(t) \equiv 0$ . For the unforced analysis, two experimental trials are performed for the unforced validation. First, the soft robot is deformed slightly and then released from rest, which corresponds

**Figure 1.12.** (a) Validation results of the dynamic model in unforced conditions, where the dashed lines represent the experimental measurements and the solid lines are the simulated trajectories. (b) Validation results of the dynamic model in forced conditions, where the dashed lines represent the experimental data from the inertial sensor and the solid lines simulated. In addition, the figure also illustrates the model for Hookean elasticity and applied pressure inputs  $u(t)$  to the dynamical system. It is evident from these comparison results that FEM-driven elasticity models can significantly improve accuracy. (c) Experimental validation of the one-link soft robot subjected to various end-effector payloads of different mass  $m_\delta = \{0.05, 0.1, 0.15\}$  kg. (d) The quasi-static deformation produced by the hyper-elastic model under various end-effector loads. In both figures, the solid lines represent the dynamic model and the dashed lines are the experimental measurements.

to the initial conditions  $\mathbf{q}_0 = (0.015, 4.75, 0)^\top$  and  $\dot{\mathbf{q}}_0 = 0_3$ . Since the mechanical deformations are relatively small here, the presence of hyper-elastic and visco-elastic material behavior are less dominant. Secondly, the soft robot is moderately deformed such that the initial configuration (or shortly after) lies within the hyper-elastic and visco-elastic regime. In this scenario, the nonlinear and time-dependent material effects may not be neglected. These initial conditions correspond to  $\mathbf{q}_0 = (0.046, 11.25, 0)^\top$  and  $\dot{\mathbf{q}}_0 = 0_3$ . It is worth mentioning that the creep strains  $\lambda$  are difficult to distinguish from the true strain, and thus the initial conditions for  $\lambda(t_0)$  are determined empirically. The validation results for both unforced scenario are shown in Figure 1.12a. The associated code for the unforced validation simulations can be found under `./valid_one_link_open.m`.

As can be seen, the state trajectories of the end-effector closely match the ground truth trajectories, even for significant nonlinear deformation. For the first validation run (inside linear elastic regime), the RMS error and the maximum error are  $\pm 0.19$  and  $\pm 0.50$  degrees, respectively. For the second case (outside the linear elastic regime), the RMS error and the maximum error is  $\pm 0.78$  and  $\pm 2.33$  degrees, respectively.

Second, we consider a forced scenario in which a regulated pressure input is applied to the pneumatic bellows. Since the pneumatic mapping  $H$  in (??) plays an important role here, the actuator coefficients are recomputed to match the experimental data better. To be more specific, by considering a pre-defined set of excitation signals  $u(t)$  of various amplitudes and frequencies, a least-squares optimization routine is employed that minimizes the difference between the measured states  $\hat{\mathbf{q}}$  with the simulated states  $\mathbf{q}$  by tuning the coefficients  $\alpha_\varepsilon$  and  $\alpha_\kappa$ . This leads to the following values:  $\alpha_\varepsilon = 2.34 \cdot 10^{-7}$  and  $\alpha_\kappa = 1.61 \cdot 10^{-8}$ . As for the

excitation signal, we have chosen the following input:

$$u_i(t) = P_0 + P_A \left[ \frac{1}{2} + \frac{1}{2} \sin(t + \phi_i) \right] \cdot \max(0.05t, 1), \quad (1.55)$$

with a static offset  $P_0 = 10$  kPa, an amplitude  $P_A = 25$  kPa, and a phase offset  $\phi_i = (i - 1)\frac{2\pi}{m}$  rad. To highlight the significance of the proposed hyper-elastic modeling approach, we also compare the results using an optimized Hookean material model with  $k_e = 50.6$  N/m and  $k_b = 5.8 \cdot 10^{-4}$  Nm/rad. The initial conditions are set to zero. The validations results for both the FEM-driven hyper-elasticity model and linear model in the forced setting are shown in Figure 1.12b. The figure also shows the measured outputs from the pneumatic VEAB regulators  $u_i(t)$ , which are directly fed into both linear and hyper-elastic models. The associated code for the forced validation can be found under `./valid_one_link_closed.m`.

Given these results, two key observations can be made. First, both the linear Hookean and hyper-elastic models provide reasonable accuracy for small deformations  $0 \leq \kappa(t) \leq 3$  with a RMS error of  $\pm 0.13$  and  $\pm 0.16$  in curvature, respectively. However, as deformations exceed the linear regime, the hyper-elastic model significantly outperforms the Hookean model. Focusing on the hyper-elastic model, both the asymmetric stiffness in radial direction and the strain-hardening are captured well, where the linear models is not sufficiently rich to capture the material effects. The overall RMS errors for the linear and hyper-elastic model are  $\pm 1.79$  and  $\pm 0.21$  in curvature, respectively. Regarding the end-effector accuracy, the overall RMS errors for the linear and hyper-elastic model are  $\pm 2.58$  and  $\pm 0.65$  mm, respectively; which translates to an arc-length normalized error of  $\pm 4.09\%$  and  $\pm 1.03\%$ . These validations results show that introducing nonlinear elastic effects driven by FEM-data can further improve the accuracy for a larger region of the soft robot's workspace.

For the last validation case, we subject the soft robot to an external payload of mass  $\delta_m$  located at the end-effector. To model the disturbance, we use the expression for the external payload disturbance model  $\delta_m$  in (1.25). The goal here is *i*) to demonstrate the accuracy of the proposed payload model and quasi-static behavior of the dynamic model, and *ii*) to highlight the limitations of the PCC assumptions under certain conditions. In this analysis, we consider three different payloads  $m_\delta = \{0.05, 0.1, 0.15\}$  kg. The experimental results of the payloads deformations are shown in Figure 1.12c, and the resulting quasi-static deformations of the dynamic model are shown in Figure 1.12d. The associated code for the external payload simulations can be found under `./valid_one_link_tipload.m`.

As can be seen, the quasi-static behavior of the dynamic model matches the experimental results relatively well for smaller payloads. For the mass  $m_\delta = 0.05$  kg, the Euclidean error between the model and the measurement are  $\pm 1.31$

mm. Increasing the payload to  $m_\delta = 0.1$  kg leads to an error of  $\pm 2.15$  mm. We can also clearly observe that the estimate of the backbone curve subject to the PCC condition is beginning to deviate from the ground truth yet the overall shape still matches the experimental data. Lastly, by further increase the payload  $m_\delta = 0.15$  kg, we clearly observe the limitations of the PCC condition under external loads – with an end-effector error of 3.98 mm. Also, there is a clear discrepancy in the backbone curve of the model and the ground truth, which might imply the PCC condition is no longer valid here as the payload could induce non-constant curvatures along the backbone. A possible solution might be to introduce a different shape parametrization, similar to the works [16, 18, 29, 30].

**Example 1.6** (Model benchmark – Multi-link soft manipulator case). In this section, we benchmark the proposed numerical integration scheme for a dynamic model of a six-link soft robot manipulator ( $N = 6$ ). Here, we want to highlight that sufficient numerical speed can be obtained while preserving sufficient numerical precision. This is an important criteria for model-based control, as slow numerical models lack transferability from theory to application. As mentioned earlier, the numerical integration of the Lagrangian entities is the computational bottleneck. Although the MDE solver does aid with numerical performance; ultimately, using a balanced spatial and temporal discretization impacts real-time performance the most. Trivially, using larger stepsize – both in space and time – lead to a decrease in numerical precision; and in some cases numerical instability. In this benchmark, we investigate these effects by varying two solver parameters: the spatial stepsize of the explicit MDE solver denoted by  $\Delta\sigma$  and the temporal stepsize of the implicit trapezoidal solver denoted by  $\Delta t$ . For convenience, we represent these stepsizes as standardized parameters: the number of finite elements  $N_s = L/\Delta\sigma$  and the implicit solver frequency  $f_s = \Delta t^{-1}$ . For the benchmark, we choose a total length of  $L = 0.15$  m and simulation time of  $T = 10$  s.

The extension to the multi-link soft robot ( $N = 6$ ) can be described by a following generalized coordinates with the following structure:

$$\mathbf{q} = (\varepsilon_1, \kappa_{x,1}, \kappa_{y,1}, \dots, \varepsilon_6, \kappa_{x,6}, \kappa_{y,6})^\top \in \mathcal{Q} \quad (1.56)$$

To ensure the soft manipulator is self-supporting, we introduce slight variations to the hyper-elastic stiffness, link lengths, and the inertial properties of the dynamical system. Considering homogeneity, all links are chosen identical in length and mass: intrinsic link length  $L_i = 0.025$  m and mass  $m_i = 0.05$  kg. Next, the bending stiffness is slightly altered where we choose  $\alpha_3 = 0.425$  Nm/rad and  $\alpha_4 = 0.4$  Nm/rad. Please note that the material domain is now given by  $\mathbb{X} = [0, \sum_{i=1}^N L_i]$ . To introduce some interesting dynamics for the benchmark, we purely excite the

**Figure 1.13.** (a) State trajectories of six-link soft robot model under dynamic excitation. The figure shows the extensible elongation strains and the total curvature  $\kappa = \sqrt{\kappa_x^2 + \kappa_y^2}$  (as the dynamics are planar). (b) The three-dimensional representation of the end-effector trajectory of the six-link soft robot. Notice that the dynamics of the six-link robot converge to a periodic solution.

first link of the serial-chain soft robot manipulator with a harmonic input:

$$u_i(t) = \begin{cases} P_a \cos(t) & \text{for } i = 1, \\ 0 & \text{otherwise,} \end{cases} \quad (1.57)$$

where  $P_a = 125$  kPa is the pressure amplitude. As for the pneumatic mapping that converts pressure to joint torques, we choose  $\tau = (H \otimes I_6) u$  as the corresponding pneumatic map for the six-link soft manipulator. In total 36 benchmark simulations with different solver settings were performed and tested for their precision relative to a high-precision model ( $f_s = 500$  Hz and  $N_s = 500$ ). The state trajectories of the high-precision model are shown in Figure 1.13a, whereas Figure 1.13b is provided to highlight the underlying dynamics and the trajectory of the end-effector. The results for all benchmark simulations are shown in Table 1.2. The associated code for the six-link simulations can be found under `./benchmark_six_link.m` on the repository.

Let us first discuss the dynamics of the six-link soft manipulator subjected to a harmonic input. Given this relatively straightforward harmonic excitation, some interesting (stable) nonlinear dynamics appear. Although we excite the system using one harmonic, the dynamics of the soft robot show a rich collection of harmonic oscillations – highlighting its nonlinear nature. Furthermore, after a short transient time (i.e.,  $t < 5$ ), the solutions of the multi-link soft robotic system tend to a so-called '*periodic solution*'. Here, a solution is called periodic if there exists a period time  $T_c > 0$  such that  $\mathbf{q}(t) = \mathbf{q}(t + T_c)$  for all time  $t$ . Similar observations of the existence of periodic solutions (and control of such oscillations) were reported by Della Santina et al. [?] for articulated soft robots. Given the harmonic excitation in (1.57), the period time here is  $T_c = 1$ .

Now, we can exploit these periodic solutions for benchmarking the numerical solver in which we compare the solution of a high-resolution model (i.e., ground truth) to the benchmark solution. Let us define the benchmark error as the Eu-

clidean distance between the two periodic solutions:

$$\begin{aligned} e &= \int_0^{T_c} \left\| {}^0\hat{p}(q(t), L) - {}^0p(q(t), L) \right\| dt, \\ &\approx \frac{1}{W} \sum_{i=1}^W \left\| {}^0\hat{p}(q(t_i), L) - {}^0p(q(t_i), L) \right\|, \end{aligned} \quad (1.58)$$

where  $W$  is the number of time samples of the benchmark model inside the time interval of the periodic solution, and  ${}^0\hat{p}(\cdot, L)$  and  ${}^0p(\cdot, L)$  the tip position of the benchmark and the ground truth, respectively. The index  $W$  naturally depends on the sampling frequency of the implicit solver. Table 2 shows the normalized errors (i.e., the tracking error  $e$  normalized with the manipulator length  $L$ ) together with the effective computation times of the numerical solver.

**Table 1.2.** Benchmark results of the six-link soft robot manipulator for various temporal and spatial discretizations ( $T = 10$  s). The tables shows the mean tracking error of the end-effector relative to a ground truth ( $f_s = 500$  Hz and  $N_s = 500$  elements). The RMS errors are normalized with the total length  $L = 0.15$  (i.e., the errors are presented in %). The CPU times are also given; where the light-gray entries achieve real-time computation (consistently).

$N = 6$ s	$N_s = 24$	$N_s = 36$	$N_s = 60$	$N_s = 90$	$N_s = 120$	$N_s = 180$
$f_s = 25$ Hz	<b>19.0 / 1.77s</b>	<b>9.37 / 2.54s</b>	<b>5.36 / 4.06s</b>	<b>4.63 / 5.98s</b>	<b>4.62 / 7.93s</b>	<b>4.43 / 11.75s</b>
$f_s = 50$ Hz	17.5 / 3.50s	7.04 / 5.04s	4.33 / 6.73s	3.69 / 11.60s	3.60 / 15.3s	3.69 / 22.71s
$f_s = 75$ Hz	13.77 / 5.31s	5.80 / 7.56s	4.51 / 12.41s	4.22 / 17.49s	3.68 / 22.35s	3.72 / 33.33s
$f_s = 100$ Hz	14.73 / 6.48s	6.12 / 9.10s	4.42 / 14.54s	3.71 / 21.46s	3.53 / 28.35s	2.89 / 42.03s
$f_s = 150$ Hz	15.22 / 7.26s	5.73 / 9.78s	4.32 / 15.58s	3.46 / 23.07s	3.16 / 30.38s	2.67 / 45.06s
$f_s = 250$ Hz	14.61 / 11.98s	5.32 / 16.71s	4.14 / 27.11s	3.35 / 40.19s	2.86 / 53.06s	2.21 / 78.61s

The benchmark table above gives some useful insight into which settings benefit numerical precision and speed the most. First, regarding the coarser models  $N = 24$  (i.e., 4 elements per link), we observe large numerical errors of  $> 10\%$  independent of solver frequency. Although one could argue these settings grantee real-time performance (2s of simulation time for 1s computation at  $N_s = 24$ ,  $f_s = 75$  Hz) and might suffice for model-based control in slow settings (i.e., set-point stabilization), they are most likely unsuited for dynamic tracking objectives. Moving towards  $N_s = 60$  (i.e., 10 elements per link), we observe a significant improvement in numerical precision  $\pm 4\%$  and while still retaining real-time performance (1.6s of simulation time for 1s computation time at  $f_s = 50$  Hz). As such, these settings show that both numerical precision and real-time computation can be achieved for more complicated multi-links soft robots. However, regarding the high-accuracy models  $N_s > 100$ , we see only a slight increase in numerical precision  $\pm 3\%$  yet lose real-time capabilities. Consequently, these models might be suited for offline simulations, but lack transferability to online model-based control. However, a possible solution here might be to convert the MATLAB model into a C or C++ equivalent model to further increase numerical speed [32, 34].

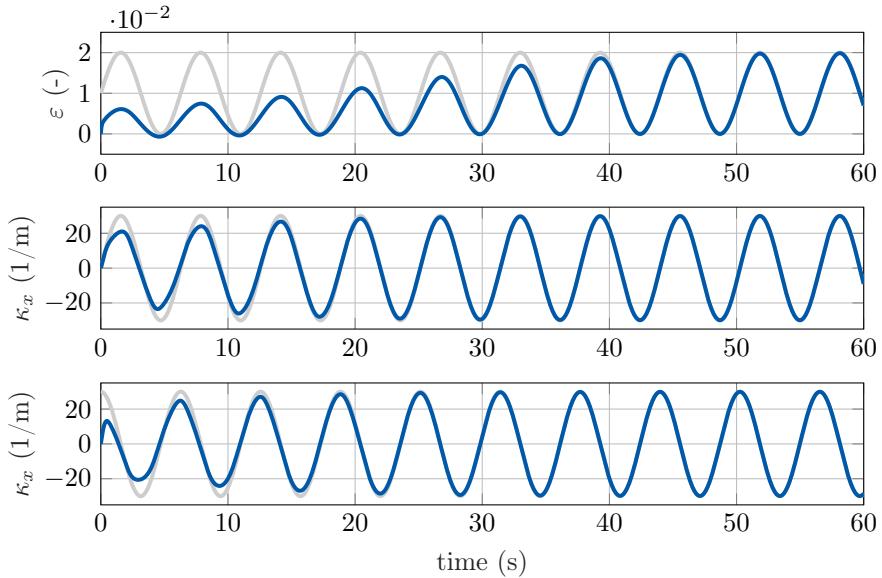
**Example 1.7** (Closed-loop control – Passivity-based controller under material certainty and mass disturbance). In the last numerical analysis, we demonstrate the results of the proposed passivity-based adaptive scheme. Due to its compactness and computational speed, we again consider the model of the one-link soft robot as seen in previous simulations. To illustrate the robustness of the controller, we purposely introduce some uncertainties. First, we introduce uncertainties in the hyper-elastic material parameters that deviate moderately from their true values in Table 1. Second, we introduce a payload to the end-effector of mass  $m_\delta = 0.1$  kg. We again model this external disturbance using the relation in (1.25). Following the linear parametrizability of the uncertainties, the parameter estimation vector yields:

$$\tilde{\boldsymbol{\pi}}(t) = (\tilde{\alpha}_1, \tilde{\alpha}_2, \tilde{\alpha}_4, \tilde{\alpha}_5, \tilde{m}_\delta)^\top, \quad (1.59)$$

with initial conditions  $\tilde{\boldsymbol{\pi}}(t_0) = \tilde{\boldsymbol{\pi}}_0 = (0.75\alpha_1, 0.75\alpha_2, 0.45\alpha_4, 0.45\alpha_5, 0.65m_\delta)^\top$ . Furthermore, the feedback gains and the adaptation rate are chosen as diagonal matrices as follows  $K_p = \text{diag}(5, 5e-5, 5e-5)$ ,  $K_d = \text{diag}(1, 1e-5, 1e-5)$ ,  $\Lambda = I_3$ , and  $K_{\Pi} = \text{diag}(5e3, 5e3, 2e-6, 2e-6, 2e-2)$ . These values were found to yield the best performance while avoiding noticeable oscillations in the closed-loop dynamics. Lastly, the following reference trajectory is considered:

$$\boldsymbol{q}_d(t) = (0.01 + 0.01 \sin(t), 30 \sin(t), 30 \cos(t))^\top, \quad (1.60)$$

Since the reference trajectory above satisfies persistence of excitation (see Remark 2), it should grantee the all convergence of the hyper-elastic material estimation



**Figure 1.14.** State trajectories of soft robot with the passivity-based adaptive controller in (1.53). The figure shows the elongation strains and the curvatures in the  $xz$ -plane and  $yz$ -plane, respectively.

**Figure 1.15.** The evolution of the elongation stiffness estimate and bending stiffness estimate,  $k_e(q, \hat{\Pi})$  and  $k_b(q, \hat{\Pi})$  respectively; and the estimate of the payload  $\hat{m}_\delta$ . For illustration, the estimates are normalized with their true value, where we can see they slowly approach one after sufficient time passes.

and the unknown mass contribution. Figure ??a shows the state trajectories of the soft robot, Figure ??b is provided to better illustrate the underlying dynamics of the controller, and Figure 1.15 shows the evolution of the nonlinear stiffnesses estimates,  $\hat{k}_e(q, \hat{\Pi})$  and  $\hat{k}_b(q, \hat{\Pi})$ , respectively; and the payload estimate  $\hat{m}_\delta(\hat{\Pi})$ .

As can be seen in Fig. ?? and Fig. 1.15, the passivity-based controller offers good performance in the face of material uncertainties and external disturbances. The RMS tracking error in steady-state (i.e.,  $t \geq 30$ s) between the desired end-effector trajectory and true trajectory is  $\pm 0.77$  mm; which translates to an arc-length normalized error of  $\pm 1.22\%$ . Despite the presence of uncertainties, the passivity-based controller also ensures the states converges to the desired trajectory with a smooth transient. Regarding the results of Figure 9, the (nonlinear) stiffness estimates  $k_e(q, \hat{\Pi})$  and  $k_b(q, \hat{\Pi})$ , and the unknown payload mass  $\hat{m}_\delta$  slowly converge to the their true values. It should be mentioned that increasing the adaption rate

leads to undesired (but bounded) oscillations of the estimates rather than faster convergences, therefore negatively affects the controller's performance.

## 1.9 Summary

In this chapter, we aimed to reduce the gap between modeling and control-oriented research in soft robotics. To promote the use of model-based controller, two issues must be addressed. First, the dynamic models that describe the continuum-bodied motions need to be sufficiently accurate, and second the model must retain real-time performance to be applicable in control. By building upon the existing Piece-Wise Constant Curvature models, we express the continuum deformation using a minimal set of coordinates related to the differential geometry of spatial curves; and explored Finite-Element driven data to model the hyper-elasticity. To retain numerical performance, a reduced-order integration scheme is developed that efficiently computes the entries of the Lagrangian model through a Matrix-Differential equation. Combined with an implicit time integration, the continuum dynamics of an hyper-elastic soft robot can be computed in real-time with a small lost in numerical precision.

The dynamic model has been extensively corroborated through simulations and experimental results. Not only does the dynamic models allow for real-time simulations with systems with various degrees of motion, it show good correspondence with the true physical soft robot. Furthermore, a passivity-based adaptive controller is proposed that provides good tracking performance even in the face of parameter uncertainties. The adaptive controller enables online estimation of the hyper-elastic stiffness and external loads, which further enhances the robustness toward modeling uncertainty undoubtedly present in soft robotics. In future work, we wish to further explore FEM-driven data for the parametrization of the spatial shape functions – extending beyond constant-curvature, and employ the proposed controller methods to multi-link soft robots.

# Bibliography

- [1] V. Falkenhahn, T. Mahl, A. Hildebrandt, R. Neumann, and O. Sawodny, “Dynamic Modeling of Bellows-Actuated Continuum Robots Using the Euler-Lagrange Formalism,” *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1483–1496, 2015.
- [2] K. Suzumori, S. Iikura, and H. Tanaka, “Development of flexible microactuator and its applications to robotic mechanisms,” in *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1622–1627, 1991.
- [3] I. S. Godage and I. D. Walker, “Dual Quaternion based modal kinematics for multi-section continuum arms,” *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015-June, no. June, pp. 1416–1422, 2015.
- [4] I. S. Godage, G. A. Medrano-Cerda, D. T. Branson, E. Guglielmino, and D. G. Caldwell, “Dynamics for variable length multisection continuum arms,” *Int. J. Rob. Res.*, vol. 35, pp. 695–722, May 2016.
- [5] A. D. Marchese, C. D. Onal, and D. Rus, “Autonomous Soft Robotic Fish Capable of Escape Maneuvers Using Fluidic Elastomer Actuators,” *Soft Robotics*, vol. 1, no. 1, pp. 75–87, 2014.
- [6] S. Kriegman, D. Blackiston, M. Levin, and J. Bongard, “A scalable pipeline for designing reconfigurable organisms,” *Proceedings of the National Academy of Sciences*, vol. 117, no. 4, pp. 1853–1859, 2020.
- [7] K. C. Galloway, K. P. Becker, B. Phillips, J. Kirby, S. Licht, D. Tchernov, R. J. Wood, and D. F. Gruber, “Soft Robotic Grippers for Biological Sampling on Deep Reefs,” *Soft Robotics*, vol. 3, no. 1, pp. 23–33, 2016.
- [8] W. Choi, G. M. Whitesides, M. Wang, X. Chen, R. F. Shepherd, A. D. Mazzeo, S. A. Morin, A. A. Stokes, and F. Ilievski, “Multigait soft robot,” *Proceedings of the National Academy of Sciences*, vol. 108, no. 51, pp. 20400–20403, 2011.
- [9] M. Pilz da Cunha, S. Amberg, M. G. Debije, E. F. Homburg, J. M. den Toonder, and A. P. Schenning, “A Soft Transporter Robot Fueled by Light,” *Advanced Science*, vol. 7, no. 5, pp. 1–7, 2020.
- [10] 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.

- [11] Z. Li, L. Wu, H. Ren, and H. Yu, "Kinematic comparison of surgical tendon-driven manipulators and concentric tube manipulators," *Mechanism and Machine Theory*, vol. 107, no. September 2016, pp. 148–165, 2017.
- [12] M. Cianchetti, T. Ranzani, G. Gerboni, T. Nanayakkara, K. Althoefer, P. Dasgupta, and A. Menciassi, "Soft Robotics Technologies to Address Shortcomings in Today's Minimally Invasive Surgery: The STIFF-FLOP Approach," *Soft Robotics*, vol. 1, no. 2, pp. 122–131, 2014.
- [13] B. A. Jones and I. D. Walker, "Kinematics for multisection continuum robots," *IEEE Transaction on Robotics*, vol. 22, pp. 43–55, Feb. 2006.
- [14] H. Mochiyama, E. Shimemura, and H. Kobayashi, "Shape Control of Manipulators with Hyper Degrees of Freedom," *The International Journal of Robotics Research*, vol. 18, no. 6, pp. 584–600, 1999.
- [15] H. Mochiyama and T. Suzuki, "Kinematics and dynamics of a cable-like hyperflexible manipulator," *IEEE International Conference on Robotics and Automation*, pp. 3672–3677, 2003.
- [16] G. S. Chirikjian and J. W. Burdick, "A hyper-redundant manipulator," *IEEE Robotics Automation Magazine*, vol. 1, pp. 22–29, 1994.
- [17] A. D. Marchese and D. Rus, "Design, kinematics, and control of a soft spatial fluidic elastomer manipulator," *The International Journal of Robotics Research*, vol. 35, no. 7, pp. 840–869, 2016.
- [18] C. Della Santina and D. Rus, "Control oriented modeling of soft robots: The polynomial curvature case," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 290–298, 2020.
- [19] R. K. Katzschnmann, C. D. Santina, Y. Toshimitsu, A. Bicchi, and D. Rus, "Dynamic motion control of multi-segment soft robots using piecewise constant curvature matched with an augmented rigid body model," *RoboSoft 2019 - 2019 IEEE International Conference on Soft Robotics*, no. February, pp. 454–461, 2019.
- [20] F. Renda, F. Boyer, J. Dias, and L. Seneviratne, "Discrete Cosserat Approach for Multisection Soft Manipulator Dynamics," *IEEE Transactions on Robotics*, vol. 34, no. 6, pp. 1518–1533, 2018.
- [21] E. Tatlicioğlu, I. D. Walker, and D. M. Dawson, "New dynamic models for planar extensible continuum robot manipulators," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1485–1490, IEEE, Oct. 2007.
- [22] E. Tatlicioğlu, I. D. Walker, and D. M. Dawson, "Dynamic Modelling for Planar Extensible Continuum Robot Manipulators," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 1357–1362, IEEE, Apr. 2007.
- [23] C. Duriez, "Control of elastic soft robots based on real-time finite element method," pp. 3982–3987, Proceedings - IEEE International Conference on Robotics and Automation, 2013.

- [24] E. Coevoet, T. Morales-Bieze, F. Largilliere, Z. Zhang, M. Thieffry, M. Sanz-Lopez, B. Carrez, D. Marchal, O. Goury, J. Dequidt, and C. Duriez, "Software toolkit for modeling, simulation, and control of soft robots," *Advanced Robotics*, vol. 31, no. 22, pp. 1208–1224, 2017.
- [25] F. Largilliere, V. Verona, E. Coevoet, M. Sanz-Lopez, J. Dequidt, and C. Duriez, "Real-time control of soft-robots using asynchronous finite element modeling," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015-June, no. June, pp. 2550–2555, 2015.
- [26] O. Goury and C. Duriez, "Fast, Generic, and Reliable Control and Simulation of Soft Robots Using Model Order Reduction," *IEEE Transactions on Robotics*, vol. 34, no. 6, pp. 1565–1576, 2018.
- [27] 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.
- [28] F. Boyer, M. Porez, and A. Leroyer, "Poincaré–Cosserat Equations for the Lighthill Three-dimensional Large Amplitude Elongated Body Theory: Application to Robotics," *Journal of Nonlinear Science*, vol. 20, pp. 47–79, Feb. 2010.
- [29] F. Boyer, V. Lebastard, F. Candelier, and F. Renda, "Dynamics of Continuum and Soft Robots: A Strain Parameterization Based Approach," *IEEE Transactions on Robotics*, vol. 37, no. 3, pp. 847–863, 2021.
- [30] F. Renda, C. Armanini, V. Lebastard, F. Candelier, and F. Boyer, "A Geometric Variable-Strain Approach for Static Modeling of Soft Manipulators with Tendon and Fluidic Actuation," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4006–4013, 2020.
- [31] J. Till, V. Alois, and C. Rucker, "Real-time dynamics of soft and continuum robots based on Cosserat rod models," *International Journal of Robotics Research*, vol. 38, no. 6, pp. 723–746, 2019.
- [32] S. Grazioso, G. Di Gironimo, and B. Siciliano, "A Geometrically Exact Model for Soft Continuum Robots: The Finite Element Deformation Space Formulation," *Soft Robotics*, vol. 6, no. 6, pp. 790–811, 2019.
- [33] S. M. Mustaza, Y. Elsayed, C. Lekakou, C. Saaj, and J. Fras, "Dynamic Modeling of Fiber-Reinforced Soft Manipulator: A Visco-Hyperelastic Material-Based Continuum Mechanics Approach," *Soft Robotics*, vol. 6, pp. 305–317, June 2019.
- [34] S. M. H. Sadati, S. E. Naghibi, A. Shiva, B. Michael, L. Renson, M. Howard, C. D. Rucker, K. Althoefer, T. Nanayakkara, S. Zschaler, C. Bergeles, H. Hauser, and I. D. Walker, "TMTDyn: A Matlab package for modeling and control of hybrid rigid–continuum robots based on discretized lumped systems and reduced-order models," *International Journal on Robotics Research*, vol. 40, pp. 296–347, Jan. 2020.
- [35] J. J. E. Slotine and L. Weiping, "Adaptive manipulator control: A case study," *IEEE Transactions on Automatic Control*, vol. 33, no. 11, pp. 995–1003, 1988.

- [36] A. P. Morgan and K. S. Narendra, “On the Uniform Asymptotic Stability of Certain Linear Nonautonomous Differential Equations,” *SIAM Journal on Control and Optimization*, July 1977.
- [37] R. Ortega, J. A. L. Perez, P. J. Nicklasson, and H. Sira-Ramirez, *Passivity-based Control of Euler-Lagrange Systems: Mechanical, Electrical and Electromechanical Applications*. London, England, UK: Springer, Sept. 1998.
- [38] E. Franco and A. Garriga-Casanovas, “Energy-shaping control of soft continuum manipulators with in-plane disturbances,” *International Journal of Robotics Research*, 2020.
- [39] B. Caasenbrood, A. Pogromsky, and H. Nijmeijer, “Dynamic modeling of hyper-elastic soft robots using spatial curves,” *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 2020.
- [40] B. Caasenbrood, “SOROTOKI - an open-source soft robotics toolkit for matlab,” GitHub, 2020.
- [41] C. Della Santina, R. K. Katzschmann, A. Bicchi, and D. Rus, “Model-based dynamic feedback control of a planar soft robot: trajectory tracking and interaction with the environment,” *International Journal of Robotics Research*, vol. 39, no. 4, pp. 490–513, 2020.
- [42] C. Della Santina, C. Duriez, and D. Rus, “Model Based Control of Soft Robots: A Survey of the State of the Art and Open Challenges,” pp. 1–69, 2021.
- [43] R. L. Bishop, “There is More than One Way to Frame a Curve,” *Am. Math. Mon.*, vol. 82, pp. 246–251, Mar. 1975.
- [44] K. M. Lynch and F. C. Park, *Modern Robotics: Mechanics, Planning, and Control*. Cambridge, England, UK: Cambridge University Press, 2017.
- [45] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, vol. 29. 1994.
- [46] G. Garofalo, C. Ott, and A. Albu-Schaffer, “On the closed form computation of the dynamic matrices and their differentiations,” *IEEE International Conference on Intelligent Robots and Systems*, pp. 2364–2369, 2013.
- [47] G. S. Chirikjian, *Theory and applications of hyper-redundant robotic manipulators*. PhD thesis, 1992.
- [48] P. Berthet-Rayne, S. M. H. Sadati, G. Petrou, N. Patel, S. Giannarou, D. R. Leff, and C. Bergeles, “MAMMOBOT: A Miniature Steerable Soft Growing Robot for Early Breast Cancer Detection,” *IEEE Rob. Autom. Lett.*, vol. 6, pp. 5056–5063, Mar. 2021.
- [49] M. A. Meyers and K. K. Chawla, *Mechanical Behavior of Materials*. Cambridge, England, UK: Cambridge University Press, Dec. 2009.
- [50] C. Renaud, J. M. Cros, Z. Q. Feng, and B. Yang, “The Yeoh model applied to the modeling of large deformation contact/impact problems,” *International Journal of Impact Engineering*, vol. 36, no. 5, pp. 659–666, 2009.

- [51] N.-h. Kim, *Introduction Analysis Finite Element to Nonlinear*. Springer, 2018.
- [52] B. Caasenbrood, A. Y. Pogromsky, and H. Nijmeijer, “Energy-based control for soft manipulators using cosserat-beam models,” in *International Conference on Informatics in Control, Automation and Robotics - ICINCO*, pp. 311–319, 2021.