Universidad Rey Juan Carlos

Escuela Técnica Superior de Ingeniería Informática

Programa de Doctorado en Tecnologías de la Información y las Comunicaciones

Computational Design of Flexible Structures

Jesús Pérez Rodríguez

Director: Miguel Ángel Otaduy

Tesis Doctoral

Noviembre, 2017

Abstract

Computational fabrication technologies have revolutionized manufacturing by offering unprecedented control over the shape and material of the fabricated objects at accessible costs. These technologies allow users to design and create objects with arbitrary properties of motion, appearance or deformation. This rich environment spurs the creativity of designers and produce an increasing demand for computer-aided design tools that alleviate design complexity even for non-expert users.

Motivated by this fact, in this thesis, we address the computational design and automatic fabrication of flexible structures, assemblies of interrelated elements that exhibit elastic behavior. We build upon mechanical simulation and numerical optimization to create innovative computational tools that model the attributes of the fabricated objects, predict their static deformation behavior, and automatically infer design attributes from user-specified goals.

With this purpose, we propose a novel mechanical model for the efficient simulation of flexible rod meshes that avoid using numerical constraints. Then, we devise compact and expressive parameterizations of flexible structures, that naturally produce coherent designs. Our tools implement inverse design functionalities based on a sensitivity-based optimization algorithm, which we further extend to deal with local minimum solutions and highly constrained problems. Additionally, we propose interaction approaches that guide the user through the design process. Finally, we validate all these contributions by developing computer-aided design solutions that facilitate the creation of *exible rod meshes* and *Kirchho -Plateau surfaces*.

In the first part of this work, we overview the relevant foundations of mechanical simulation, analyze the optimization problem that arises from inverse elastic design and discuss alternative solutions. Then, in the second part, we propose a computational method for the design of flexible rod meshes that automatically computes a fabricable design from user-defined deformation examples. Finally, in the last part, we study the design and fabrication of Kirchhoff-Plateau surfaces and present a tool for interactively exploring the space of fabricable solutions.

Acknowledgements

I should thank many people for their care and support during the development of this thesis, but it is late night and, as usual, just a few hours ahead of the deadline. Instead, I will thank a few chosen ones, without whom it wouldn't have been possible to get to this point sane. To all those not in the list, I thank you too!

Miguel, for teaching me to be a better researcher. For his hard work and absolute devotion to his students. For being an advisor to look up to, and for his inspiration during hard times. Apparently, there is no problem that cannot be solved.

The members of the MSLab and GMRV, for the laughs during work hours and discussions at lunch time. For kindly teaching me whatever they were learning, and sharing with me the burden and joy of research. I would like to thank in particular to Rosell, Sergio, Vane, Lobo, Lillo, Rosa, Hector, Valentn, Ines, Alex, Alvaro and their loved ones, for lling all these years with incredibly good memories.

The collaborators that I have had, for inspiring me to achieve much more than I thought I could. Thanks for sharing with me your deep knowledge, and for working hard to make our projects successful. My special thanks to Bernhard, for his continued support, trust and insightful teachings.

My supervisors and fellows during the internships at Disney Zurich, for letting me be part of their amazing group. It was truly an honor to learn from them. Special thanks to Gaurav, for his companionship and humor in a strange place.

My hometown friends, for being still there and not forgetting about me. My parents, Carmen and Francisco, for their unconditional love, care and education, for which I will never be thankful enough. And my sister Teresa, who never ceases to amaze me for her wisdom, her strength, and her contagious energy.

And nally Valeria. Thank you for your in nite support, love and patience. For sharing the laughs and the drama. For helping me to be brave, to stay healthy, to be patient, to be proud, and to be happy. Hers is one half of this thesis, and she deserves all forms of acknowledgements.

And with that, I think it is time to get back to work.

Contents

Abstract iii

1 Introduction 1
1.1 Computational design 1
1.2 Designing exible structures 4
1.2.1 Mechanical simulation 5
1.2.2 Inverse elastic design 6
1.3 Flexible rod structures 7
1.3.1 Flexible rod meshes 7
1.3.2 Kirchho -Plateau surfaces 8
1.4 Contributions and publications 9
1.5 Outline 10

2 Related work 13
2.1 Mechanical simulation 13
2.1.1 Elastic rod simulation 14
2.1.2 Thin-shell simulation 15
2.1.3 Model coupling 16
2.1.4 Numerical solving 17
2.2 Computational design 18
2.2.1 Surface design 19
2.2.2 Animation control 20
2.2.3 Material characterization 21
2.2.4 Exploration of constrained spaces 22
2.2.5 Fabrication-oriented design 23

3 Mechanical simulation 27
3.1 Foundations 27
3.2 Rod mesh model 29
3.2.1 Discrete rod model 29
3.2.2 Rod network model 35
3.3 Thin-shell model 40
3.3.1 Orthotropic membrane 41
3.3.2 Discrete shells bending 44
3.4 Static equilibrium 45
3.4.1 Root- nding solver 45
3.4.2 Optimization solver 48

4 Inverse elastic shape design 49
4.1 Problem characterization 49
4.1.1 Problem formulation 50
4.1.2 The solution space 53
4.2 Solution methodologies 54
4.2.1 Design space selection 55
4.2.2 Target shape de nition 56
4.2.3 Constrained optimization solution 57
4.2.4 Constraint exploration solution 61
4.3 Sensitivity-based optimization 61
4.3.1 Implicit equilibrium 61
4.3.2 Sensitivity-based SQP 64
4.3.3 Quadratic subproblem 66
4.3.4 Equilibrium projection 72
4.3.5 Discussion 77

5 Flexible rod meshes 79
5.1 Introduction 79
5.2 Computational model 81

5.2.1 Mechanical model 81
5.2.2 Design space 82
5.3 Optimization problem 85
5.3.1 Numerical optimization 85
5.3.2 Optimization scheme 87
5.4 Experimental results 87
5.4.1 Performance 88
5.4.2 Printed examples 90
5.4.3 Discussion and future work 96

6 Kirchho -Plateau surfaces 97
6.1 Introduction 98
6.2 KPS shape space 100
6.3 Computational model 102
6.3.1 Mechanical model 102
6.3.2 Design space 103
6.4 Computational design 105
6.4.1 Forward design 106
6.4.2 Sensitivity exploration 109
6.4.3 Inverse design 111
6.5 Experimental results 114
6.5.1 Printed examples 114
6.5.2 Fabrication process 119
6.5.3 Discussion and future work 120

7 Conclusions 121
7.1 General conclusions 121
7.1.1 Flexible rod meshes 122
7.1.2 Kirchho -Plateau surfaces 122
7.1.3 Mechanical simulation 123
7.1.4 Inverse elastic design 123

7.2 Discussion and future work 124
7.2.1 Rod structures applications 124
7.2.2 Inverse elastic design problem 125
7.2.3 Design guidance and topology 127

Bibliography 129

A Resumen 149
A.1 Antecedentes 150
A.2 Objetivos 151
A.3 Metodologa 154
A.4 Resultados 157
A.5 Conclusiones 159

Chapter 1

Introduction

This thesis deals with the computational design of exible structures. We build upon mechanical simulation and numerical optimization to develop computer-aided design tools that facilitate the creation of exible rod meshes and Kirchho -Plateau surfaces For this purpose, we de ne compact parameterizations that determine the geometry and material of the structures; propose novel mechanical models that allow us to accurately and e ciently predict their deformed shape in static equilibrium; and devise optimization methods for the automatic inference of design attributes from user-de ned functional goals. This chapter provides a brief introduction on these topics, focusing on the challenges that have been faced during the development of the thesis, and the contributions that resulted from its completion.

1.1 Computational design

Computational fabrication technologies like 3D printing o er unprecedented control over the shape and material attributes of the fabricated objects, with generally little additional cost dependent on object complexity. This enables an extremely fast transition from virtual prototypes to physical realizations. For these reasons, these technologies have already reached an enormous impact on industrial engineering, where rapid prototyping of all kinds of products, from furniture to vehicle parts, is already leading to cost reductions and e ciency gains. In addition, computational fabrication has also \democratized" production tools and brought them closer to designers, due to their relatively low purchase and material costs. Having full control over the geometry and material complexity enables the production of objects with arbitrary properties of motion, appearance or deformation, among others. Designers are ultimately responsible for creating the \design" or pattern for the construction of an object considering aesthetic, functional or economic goals.

However, humans are limited in their ability to handle design complexity. For instance, they may lack the understanding of the underlying process that drives the deformation of an object; or it may be too costly to analytically predict the e ect of a texture on the re ectance of the material. In general, regardless of the level of expertise of the design team, a typical product goes through an iterative process of analysis, design, prototyping and evaluation that is extremely time consuming. With computational fabrication technologies being now accessible to the public and massively adopted in industry, there is an increasing demand of computational tools that alleviate the burden of design complexity, coming from potential users with a broad range of expertise levels.

Over the last decade, a large part of the computer graphics community has engaged in responding to this demand, taking advantage of the community's scienti c heritage. The pursuit of realistic real-world representations has led to data structures for geometry and attribute representation; algorithms for image analysis, synthesis and simulation; and human-computer interaction techniques. All this interdisciplinary knowledge has been combined into computer-aided design (CAD) solutions with the purpose of facilitating the design of fabricable objects.

Modern computational design solutions o er four major functionalities (Fig. 1.1):

1. Modeling design attributes. This implies the de nition of a design space, i.e., nding a computational representation of the possibly complex geometry and material attributes that determine the fabricated object. In this sense, there is a preference for exhaustive but compact models that naturally produce \good" designs yet facilitate user control.

2. Predicting the properties of the fabricated object for a given design. This has been called *forward design* and it replaces the costly and time consuming design cycle. A user knows in advance the mapping from design attributes to resulting properties, which allows extremely fast iterations without the need for physical realizations. Forward design involves building a computational model of the underlying process governing the properties of the object for simulation (e.g., mechanical elasticity for elastic deformation).

3. Inferring the design of the object that produces some target user-de ned properties. This is the opposite functionality to forward design and has been called *inverse design*. This feature aims to replace {or at least reduce the need for{ expert knowledge. The user is not required to know the details of the design, but only to specify the properties that the fabricated object should ful ll (e.g., balance, stability or buoyancy). Through numerical optimization of the computational model built for prediction, the computer infers the corresponding geometry and material attributes that give rise to user speci cation.

4. Guiding the design process. In some design problems, it is not even clear which is the solution space, i.e., the range of properties that can be achieved by fabricable objects. Understanding of the solution space may be achieved by finding high-level descriptors of fabricable properties, so that the user can easily control them to define targets for inference. Nevertheless, more sophisticated implementations may directly allow designers to virtually navigate only the space of fabricable solutions, and even suggest alternative paths to spur the user's creativity.

Figure 1.1: Modern computational design solutions provide computational models to represent design attributes and fabricated object properties, and implement numerical methods allowing the user to navigate both spaces through forward and inverse design.

Fabrication-oriented design tools proposed during the last few years feature some or all of these functionalities. We have witnessed computational methods for designing objects that can stand on their own [1], spin stably [2] or sound in a specific manner [3]. It is also possible to create mechanical characters capable of interesting motions [4] and produce 3D-printable robotic creatures [5] or telescoping structures [6]. And the list goes on. This rich environment creates fertile ground for creativity which will go even further in the future, when computational fabrication technologies become progressively more accessible to the public.

Overview

In this thesis, we deal with the computational design and automatic fabrication of flexible structures: assemblies of {potentially heterogeneous{ interrelated elements that exhibit elastic behavior. With the development of these methods, we contribute on each of the functionalities mentioned above: i) we build compact parameterizations of complex structures composed of elastic rods and fabrics; ii) we devise mechanical models that allow the prediction of their deformation behavior; iii) we create optimization methods that automatically infer fabricable designs from user-defined functional goals; and iv) we propose interaction approaches that empower the user with further understanding and facilitate navigation of the solution space.

In the following sections, we give a brief overview on these topics, introducing the scienti c and technical challenges that we have addressed, and reviewing the contributions that we have developed.

1.2 Designing exible structures

The design and fabrication of deformable objects has been an integral part of human history since the origins of clothing. The adaptability of deformable materials allows compliant tools to remain functional in dynamic and constrained environments, and to safely interact with life forms and fragile objects. Soft materials are being used increasingly more often for the production of toys, apparel, furniture, architecture or robots, among others.

Originally limited to rigid materials, computational fabrication methods have recently opened the door for fast prototyping of deformable objects. While rigid designs merely require the speci cation of the geometry, the behavior of a deformable object is also heavily in uenced by the spatial distribution of material attributes. This poses a grand challenge on computer-aided design as it is non-trivial to model the continuum mechanics that drive the relationship between the local combination of geometry and materials, and the corresponding global deformation behavior.

This topic has received increased attention from the computer graphics community over the last few years. Most of the works di erentiate from each other in how they de ne the computational model that represents the design space. This characteristic is typically determined by the fabrication technology that is being employed. For instance, some works use a discrete combination of template materials with di erent deformation behaviors [7]. Other works focus on changing the overall shape of the objects to determine their deformation properties [8, 9]. Multi-material printing technologies allow us to consider an heterogeneous spatial distribution of material attributes [10, 11]. Nevertheless, several recent works are based on de ning small-scale geometry attributes to overcome the use of a single base material [12, 13]. Alternatively, other works have focused on creating interactive solutions that allow the user to rapidly navigate the space of fabricable designs [14, 15, 16]

The design and fabrication methods developed in this thesis are in line with the two latter groups of works. We aim for reducing fabrication complexity by using a single material, and we rely on local geometry changes to determine the deformation behavior of the exible structures. The interrelation between the components of the assembly poses further challenges for the de nition of the design space, which must be compact, tractable, expressive, and it must naturally produce coherent designs. At the same time, we favor approaches that enable the designers to de ne the properties of the fabricated objects either using high-level descriptors or interactive

techniques. The latter imposes and additional challenge in terms of performance.

Forward and inverse design functionalities heavily rely on mechanical simulation and elastic shape design through numerical optimization. These are the two foremost technical challenges that must be overcome for the computational design of exible structures.

1.2.1 Mechanical simulation

The computational design of deformable objects relies on mechanical simulation to accurately predict which will be the static deformation properties of a given design once fabricated. The creation of analytical models to explain the behavior of real-world phenomena has been traditionally addressed by classical physics. Engineering elds have been devoted to the development of computational model and numerical methods that allow computers to predict the behavior of materials and structures. In the last few decades, since the pioneering work by Terzopoulos et al. [17], computer graphics has heavily contributed to the innovation in this area motivated by the creation of realistic virtual representations of real-world phenomena.

Our work deals with the design and fabrication of exible structures composed of elastic rods and fabrics. The accurate simulation of the elastic behavior of fabric has been successfully tackled in the past [18, 19]. Similarly, several discrete elastic models have been suggested for the simulation of individual rods [20, 21]. However, modeling the deformation behavior of an assembly of elements, possibly of a diverse nature, has not been addressed so often: the structure should remain consistent and the di erent components should interact with each other adequately, i.e., point forces and rotational torques are transferred. In this thesis, we develop simple yet accurate computational models for the simulation of elastic rod meshes [22], in chapter 5, and tensile structures [23], in chapter 6. In both cases, we aim for creating solutions that satisfy two main requirements:

> Given some material characterization, simulations are experimentally validated to ensure that models produce accurate predictions of the elastic behavior of the structure, more precisely, its shape in static equilibrium.
>
> Simulations are e cient and avoid using numerical constraints for the coupling between the di erent elements. This is preferable so that the computational model lends itself well to numerical optimization, as this is necessary for inverse design.

Although developed in the context of a computer-aided design tool, these computational models might be of great use in other scenarios where an e cient yet accurate simulation is needed, e.g., animation or video games. In chapter 3, we

review the foundations of mechanical simulation and provide a detailed explanation of all the models used for the development of this thesis, including our own contributions.

1.2.2 Inverse elastic design

Our tools feature inverse design operators that automatically determine attributes of the elastic structures such that their deformed shapes in static equilibrium meet some user-speci ed goal. Solving this problem constitutes a very hard challenge for two reasons:

1. Any feasible deformed shape of the structure must be in static equilibrium. This imposes a hard constraint on the solution space and limits the satisfaction of user-speci ed goals, which will be infeasible in most cases.
2. Similarly, the fabrication technology imposes bounds on the design space, as often it is not possible to use arbitrary fabrication materials or arbitrarily complex geometry.

The associated constrained optimization problem heavily relies on mechanical simulation and has been repeatedly formulated throughout the extensive literature on computational design of deformable objects [7, 14, 10, 16, 24, 25]. Standard numerical optimization methods could be used to solve such problems, but they often do not guarantee nding a good solution. Together with an appropriate de nition of the design and solution spaces, challenges come with the selection of the particular optimization approach. In this thesis, we adopt an approach based on the iterative navigation of the equilibrium constraint manifold, similar to [14, 26]. Built upon this idea, we make additional contributions to deal with two problems:

1. Local minimum solutions . In the context of exible rod meshes (chapter 5), we identify the attributes of the design space having the largest e ect on the reduction of the bulk tting error, and we propose a multi-resolution algorithm that incrementally considers additional design parameters using interpolation.
2. Highly constrained problems . In the context of Kirchho -Plateau surfaces (chapter 6), we propose a two-step algorithm to interactively explore the solution space of highly constrained problems, for which user-speci ed goals are generally too far from being feasible.

Optimization challenges similar to ours are common across many other design problems, and other authors might nd inspiration in our solutions to face their own challenges. In chapter 4, we formally characterize our shape optimization problem, and we discuss other alternative solving approaches.

1.3 Flexible rod structures

Rods are curve-like elastic bodies that have one dimension ("length") much larger than the others ("section"). Assemblies of rods are extensively used in structure engineering, where rigid truss structures are specially valued for their light weight, high sti ness and low cost in terms of transport, storage and assembly time [27, 28]. Their exible counterpart are called active-bending structures, and appear in combination with elastic membranes for the creation of tensile structures that provide more design freedom and reduce the amount of external supports needed [29, 30]. The use of rod-like structures has been mainly restricted so far to architectural geometry applications, and has just started to receive attention from the computer graphics community for more general design applications like minimizing printing material [31] or producing wire meshes [32], reciprocal frames [33] and tensegrities [34], among others.

Figure 1.2: In this thesis, we explore the computational design and fabrication of two instances of elastic rod assemblies: exible rod meshes (left) and Kirchho -Plateau surfaces (right).

Nevertheless, these structures are still underused and, as we demonstrate in this thesis, might nd many other potential applications. We validate our computational tools with the design and fabrication of two di erent instances of exible rod structures: exible rod meshes and Kirchho -Plateau surfaces (Fig. 1.2).

1.3.1 Flexible rod meshes

Flexible rod meshes are assemblies of elastic rods. These structures are lightweight, relatively low-cost and can also form the support structure of solid objects if lled or dressed with other materials. But most importantly, the global deformation properties of a rod mesh can be adjusted simply by locally varying the cross-sectional

properties of the rods, i.e., the radii. Consequently, a heterogeneous deformable object can be fabricated in one piece and from a single base material. This constitutes a key feature in the current scenario of low-cost consumer level printing machines.

In chapter 5, we consider exible rod meshes as an implementation of deformable surfaces and push the limits of their expressive power. Our computational tool takes as input several poses of a deformable surface with known boundary conditions and automatically computes the rod mesh that best approximates the desired shapes. From several deformation instances with di erent elasticity properties, we seek a unique design that generalizes all the behaviors. From the design perspective, this facilitates producing complex deformation features like anisotropy, heterogeneity and model merging. In order to solve the associated optimization problem, we propose a simulation-based approach where the selection of the design space and the de nition of a proper optimization strategy are essential.

In this thesis, we have explored the potential of exible rod meshes for toy and apparel design, but they might also nd potential application in other elds like furniture design, soft-robotics, orthotics or wearable assistance.

1.3.2 Kirchho -Plateau surfaces

Kirchho -Plateau surfaces (KPS) are planar rod meshes embedded in pre-stretched fabric that deploy into complex three-dimensional shapes. In their deformed state, these structures consist of a combination of piece-wise minimal surface patches, i.e., patches shaped as the surface of minimum area that spans a given boundary. Furthermore, such boundaries can only assume shapes corresponding to the static equilibrium of a planar rod mesh under membrane tension. These seemingly complex shapes may attract the attention from the design and research communities for two main reasons:

> From an application point of view, minimal surfaces are appreciated for their smooth aesthetic appearance and inherent material e ciency and structural stability. Minimal surfaces are widely used for light-weight and cost-e cient structures, ranging from large-scale roofs, canopies and shade systems, to acoustic de ectors, light di users and decorative elements for interior design.
>
> From a theoretical point of view, at small scale, these structures can be easily manufactured by 3D-printing planar rods onto stretched fabric, as recently demonstrated by [35]. Being able to easily design and manufacture instances of the Kirchho -Plateau problem may intrigue theorists and designers that have struggled with producing physical realizations.

From a design perspective, considering the highly constrained solution space, we

cannot expect that there exists a design that closely approximates a user-de ned target shape, but nevertheless KPS provides ample room for shape abstraction, interpretation and creativity.

In chapter 6, we turn away from fully-automated solutions in favor of a user-guided but computer-assisted design paradigm. Here the user is responsible for creating the topology of the rod mesh and for transforming it into the desired shape using a set of modeling tools that implement editing operations directly on the equilibrium state of the surface. We highlight a two-step optimization scheme which is essential for the implementation of the inverse design functionality, which allows the designer to e ectively navigate the space of feasible solutions interactively. We demonstrate our method by designing a diverse set of complex-shaped KPS, each validated by a physically fabricated prototype.

1.4 Contributions and publications

The main contributions of this thesis are the following:

A mechanical model for the accurate and e cient simulation of exible rod meshes that implicitly handles coupling between rods without requiring numerical constraints or any additional degrees-of-freedom. (Chapter 3)

A computational model for the representation of the design space of exible rod meshes that is compact, expressive and naturally produces good quality designs through Hermite interpolation. (Chapter 5)

A multi-objective sensitivity-based optimization method for the inverse elastic design of exible rod meshes. In combination with the previously mentioned mechanical model and design space de nition, it allows us to infer the design attributes of a rod mesh such that its static equilibrium matches an arbitrary number of user-de ned goals with speci c boundary conditions. (Chapter 5)

A multi-resolution optimization strategy for improving the convergence of inverse elastic design problems, which adaptively increments the detail level of the design space parameterization when needed. (Chapter 5)

A computational model for the compact representation of the design space of Kirchho -Plateau surfaces that naturally handles coupling between rods and fabric using collocation and Laplacian interpolation. (Chapter 6)

A complete computational tool for the design of Kirchho -Plateau surfaces that features multiple editing operations a ecting both the design and solution spaces. Such editing operations include: i) forward geometry and topology

editing, with real-time feedback on the resulting deformation, ii) sensitivity-based modal exploration of the solution space, iii) interactive inverse modelling of the equilibrium shape with design constraint preservation, and iv) visualization of mechanical properties for decision-making. (Chapter 6)

A two-step optimization strategy based on incrementally exploring an approximation to the solution space in order to de ne close-to-feasible target deformations for the inverse elastic design problem. (Chapter 6)

The results corresponding to the contributions of this thesis are compiled in the following two works published as a rst author in ACM Transactions on Graphics, and presented at the ACM SIGGRAPH conference:

Jesus Perez, Bernhard Thomaszewski, Stelian Coros, Bernd Bickel, Jose A. Canabal, Robert Sumner, and Miguel A. Otaduy. 2015. Design and fabrication of exible rod meshes.ACM Trans. Graph. 34, 4, Article 138 (July 2015).

Jesus Perez, Miguel A. Otaduy, and Bernhard Thomaszewski. 2017. Computational design and automated fabrication of Kirchho -Plateau surfacesACM Trans. Graph. 36, 4, Article 62 (July 2017).

1.5 Outline

The rest of the thesis if organized as follows:

Related work . Chapter 2 reviews the main works that have inspired and in uenced the development of this thesis, grouping the extensive literature in two general themes: mechanical simulation and computational design.

Mechanical simulation . Chapter 3 overviews the relevant foundations of mechanical simulation and describes in detail the mechanical models and numerical solvers that have been developed in this thesis, focusing on rod-mesh and thin-shell mechanics and the static equilibrium computation.

Inverse elastic design . Chapter 4 presents a formal characterization of the inverse elastic design problem, describes some numerical solving methods, and analyzes the potential problems of their practical implementation, focusing on sensitivity-based constraint exploration.

Flexible rod meshes . Chapter 5 focuses on the computational design and fabrication of exible rod meshes and describes the computational model and optimization methods that facilitate the solution of the inverse design problem.

Kirchho -Plateau surfaces . Chapter 6 explores the computational design and fabrication of Kirchho -Plateau surfaces and proposes a user-guided but computer-assisted tool that allows a user to interactively design such structures.

Conclusions . Chapter 7 contains a discussion on the limitations of the methods presented in this thesis, analyzing their potential impact and suggesting possible future work.

Chapter 2

Related work

The design and fabrication of flexible structures is an interdisciplinary subject integrating knowledge across several related fields. This chapter reviews the main works that have inspired and influenced the development of this thesis. We group this extensive literature in two general themes: mechanical simulation and computational design.

The first section 2.1 reviews works dealing with the creation of computational methods for the simulation of mechanical systems. We focus on mechanical modeling of elastic rods and thin shells, model coupling and numerical solving.

The second section 2.2 groups many heterogeneous works related with computational design. We first categorize the literature according to several criteria and briefly review a few topics that are tangentially related with this thesis: surface design, animation control, material characterization and exploration of constrained spaces. We then focus on fabrication-oriented design problems with special emphasis on rod structures and physical surfaces.

2.1 Mechanical simulation

Classical physics has traditionally addressed the creation of analytical models to explain the behavior of real-world phenomena. From an engineering perspective, mechanical simulation comprises the development of computational models and numerical methods that allow computers to predict the behavior of materials and structures. In the last few decades, since the pioneering work by Terzopoulos et al. [17], computer graphics has heavily contributed to the innovation in this area due to its never-ending pursuit of more realistic virtual representations of the real world.

This pursuit is motivated by di erent purposes. Fabrication-oriented computational design, for example, often relies on mechanical simulation to accurately predict how a given prototype will behave once fabricated. Our work deals with the design and fabrication of deformable structures composed of elastic rods and fabric. In the following, we will summarize the main computational models that have been used for the simulation of elastic rods and thin shells and review the state of the art. Then, we brie y go over algorithms currently used to solve numerical integration problems derived from the partial di erential equations of mechanics. Finally, we will introduce a few more works dealing with the combination of potentially heterogeneous models for the simulation of complex structure assemblies.

2.1.1 Elastic rod simulation

Rods are curve-like elastic bodies that have one dimension ("length") much larger than the others ("section"). A great variety of rod simulation methods have been presented in the literature. Many, including the earliest approaches, deal with the e cient simulation of hair, highly demanded in the VFX and video games industries. These solutions are mostly based on mass-spring systems which explicitly represent the curve as a sequence of nodes joined together by elastic springs [36, 37, 38, 39]. While these methods are capable of modeling resistance to stretch and bending, their formulations do not come from the discretization of a continuous rod elasticity model. Plus, they are limited in their ability to model twist, anisotropy and curls.

Alternatively, framed representations describe the con guration of a rod by an adapted framed curve. The assignment of a material frame to each point on the centerline contains the requisite information for measuring the orientation of the rod section and hence enables advanced features. The corresponding governing equations {a set of partial di erential equations together with boundary conditions{ were rst developed by Kirchho and Clebsch [40] in their theory of thin elastic rods under nite displacements. Some works following this idea solve the statics and dynamics of Kirchho rods using an explicit centerline representation based on the discretization of the Cosserat rod geometry model [41, 42, 20]. Other works however use an implicit centerline representation and introduce reduced-coordinates models based on a minimal parameterization needed to account for the exact kinematics for the rods. Examples of these works are the solutions based on articulated rigid body systems [43] or the super-helix model [44]. Relying on the Bishop frame, discrete elastic rods (DER) and discrete viscous threads (DVT) by Bergou et al. [45, 46] use a curve-angle parameterization to reduce the number of redundant parameters and guarantee that the orientation frame naturally remains adapted to the centerline. The latter approach constitutes the best starting point for our work, as it is experimentally validated, and it lends itself well to numerical optimization, as we demonstrate later in this thesis.

Rod meshes are networks of elastic rods that are linked by joints. Not many works before our own have addressed the simulation of this kind of structures. In general, previous approaches are based on the use of numerical constraints during simulation to ensure rods remain connected. Such is the solution by Bergou et al. [45], which uses bodies rigidly attached to rods as joints. Elastic joints were considered in Cosserat Nets by Spillman and Teschner [47], which extends the Cosserat model to branched and looped topologies albeit at the expense of also using constraints. More recent works have taken the position-based dynamics (PBD) approach introduced in [48] and adapted it to the simulation of elastic rod networks [49]. However, the lack of a formal connection between the positional constraints used in PBD and elasticity theory restricts the use of this solution to applications where a predictive representation is not needed. As it will be further explained in this thesis, the work presented in chapter 5 extends the DER model introducing a physically-based elastic energy at connections. This allows the accurate simulation of rod mesh mechanics without the need of numerical constraints, which reduce the complexity of our optimization problem. A similar formulation was suggested later in the work by Zehnder et al. [50] by adding rotational DoFs at each joint.

2.1.2 Thin-shell simulation

Thin shells are thin exible structures with a high ratio of width to thickness (e.g. > 100). Thin shells have been also extensively studied by the computer graphics community, focusing mainly on the e cient and accurate modeling and simulation of cloth mechanics. Cloth has exempli ed advances in physics-based animation dealing with a broad range of problems such as mechanical modeling, contact handling, friction, plasticity, fracture and numerical integration, among others. In the following, we will roughly review the most important works and mention a few current hot topics. For a broad survey on cloth simulation we refer the reader to [18, 19].

Since the pioneering work on elastically deformable models by [51], the most widely used simulation techniques were initially mass-spring systems [52] and more general particle systems [53, 54]. The popularity of mass-spring systems is due to the ease of implementation and low computational cost, but the accuracy o ered by this method is rather limited. As an alternative, continuum-based approaches lead to a set of partial di erential equations which have to be discretized in space, generally using the nite element method (FEM). Most of the existing FE-approaches are based on the geometrically exact thin-shells formulation by Simo et al. [55]. With this basic continuum formulation, simple isotropic materials can be simulated consistently with reduced dependency on the speci c discretization. Further works incorporated novelties allowing to model increasingly more complex behaviors that are observable on real-world textile materials such as anisotropy [56, 57, 58] and nonlinear deformations [59, 60, 58]. As happened to rod simulation, many current

works are based on position-based approaches such as [61, 62] but are not accurate enough for fabrication-oriented design applications. Our work relies in the widely known discrete shellsformulation [63] for the exural energy and the classiœr-thotropic St. Venant-Kirchho constitutive model for the membrane energy (see e.g. [64]).

Apart from the mechanical model, during the last few years, there have been contributions in many other aspects of thin-shell simulation. To mention just a few examples, some works have tried to overcome the limitations of spatial discretizations using remeshing methods [65], which have been successfully applied to model complex phenomena such as folding [66] and fracture [67]. Other works have addressed the problem of modeling and estimating cloth hysteresis as an e ect of the internal friction between yarns [68]. The augmentation of coarse cloth simulations with realistic-looking wrinkles have also attracted considerable attention and clustered a variety of solutions like subspace simulation using adaptive bases [69] or procedural wrinkle creation based on the coarse strain tensor [70, 71]. Finally, a promising research line related to both thin shells and rod mechanics proposes the simulation of cloth at the yarn level [72, 73, 74, 75], which allows showing new interesting e ects and unprecedented levels of agreement to real-world materials.

2.1.3 Model coupling

This thesis deals with the computational design of structures composed by an assembly of objects, possibly of a diverse nature. To accurately predict their deformation behavior, it would be theoretically possible to use a single volumetric mechanical model. However, to obtain precise enough results, that would require to employing a very ne spatial discretization to e ectively capture su cient geometry detail at each individual object scale. The resulting computational problem would be intractable in most of the cases. As an alternative solution, computer graphics research has often considered the use of di erent mechanical models, conveniently selected to take advantage of the speci c kinematic and mechanical properties of each individual object. For example, a rod mesh can be simulated as an assembly of elastic rod models, as in chapter 5, or use a cloth model coupled to a rod mesh model at speci c locations to represent a tensile structure, as in chapter 6.

To guarantee that the structure remains consistent and the di erent objects interact with each other adequately, several coupling approaches have been explored over the years. The simplest solution is the collocation of DoFs: all objects attached to a given coupling area share their corresponding discretization variables. This ensures point forces are shared and the structure remains consistent in any case. However, rotational torques are not transferred and so this approach does not o er a complete solution for the interaction between parts. In the real world, structural

joints are not just single points but volumetric components and, consequently, rigid bodies are often placed at connections to mimic this fact. In some computer graphics works, these joints have been explicitly represented with rotational DoFs [21, 50, 23], while in others, the rotation of the joint is implicitly derived from the state of all the attached elements [47, 22]. To ensure rotational torques are transferred, the attached components must remain aligned with the rigid body at the connection point. The straightforward solution is to employ soft or hard numerical constraints to enforce this condition, as in e.g. [45, 47, 49]. In general, this approach has been extensively used for modeling many phenomena in which any kind of continued attachment between objects takes place, for instance, in frictional [76] and adhesive [77] contact. An alternative to using constraints is the de nition of physically-based energies, whose derivation leads to elastic coupling forces [78, 22, 50, 23]. The resulting joints are compliant, what, with an adequate material characterization, produces a better agreement with the behavior of real-world structural joints, even under coarse discretizations.

Nevertheless, model coupling is not only restricted to simple inertia transfer. Many interesting e ects have been achieved over the last few years by studying how objects in a heterogeneous assembly interact with each other. For instance, a few works have developed coupling methods between thin shells and deformable solids that allow the simulation of skin sliding on top of the muscles [79], facilitate adding high resolution wrinkles to coarse volumetric models [80], or model friction and air e ects between cloth and bodies [81]. In the context of model coupling, contact and sliding have been an interesting convergence point for several works built around the Eulerian-on-Lagrangian idea [82]. This method proposes the combination of the two simulation methodologies in mechanics, i.e. Eulerian and Lagrangian. This approach has been applied to formulate frictional contact models between rigid bodies and strands [83] or cloth [84] that preserve contact contours independently of the discretization level. A particularly interesting research line proposes the simulation of cloth at the yarn level [74, 85, 75]. This approach assumes all yarns share Lagrangian DoFs at their crossing points but are allowed to slide with respect to each other using Eulerian coordinates. Consequently, there is no need to explicitly handle contacts between yarns, and hence the simulation of large garments at the yarn level is more tractable. In our case, the same methodology might be applied in the future for the design of tensile structures with sliding components.

2.1.4 Numerical solving

Since the seminal work of Bara et al [86], implicit methods have been the predominant choice in computer graphics for rod and thin-shell dynamics simulation {and in general for most physically-based animation problems. For a detailed overview and comparison of existing integration schemes and their e ciency {applied to cloth

simulation{, the reader is referred to [87]. In our work, we have adapted the variational form of implicit Euler integration in [88] to solve dynamics, as well as the corresponding Lagrangian mechanics formulation of the static equilibrium problem. Recent approaches however have shifted towards position-based methods inspired by works like the previously mentioned PBD [48] and projective dynamics by Bouaziz et al. [89]. Initial implementations used positional constraints with no connection to classical elasticity theory, e.g. [61, 90, 91, 92]. However, more recent solutions like the one by Liu et al. [93] have formalized the connection between projective dynamics and quasic-Newton methods, allowing to leverage ideas from numerical optimization and thus support real-time simulation of many hyperelastic materials.

2.2 Computational design

From a general perspective, computational design deals with the creation of software that facilitate design tasks. It roughly comprises the development of a parameterized computational model of the designed entity, which allows computers to predict how this entity would perform considering aesthetics, functionality, e ciency and many other criteria.

This thesis mainly focuses on the design of exible structures, i.e., nding the parameters of a mechanical model such that the resulting fabricated object behaves and/or looks in a desired manner. However, computer graphics has explored many other related design problems which have heavily in uenced our own work. To give the reader an overall view of the diversity in the literature, we will categorize these works according to several criteria:

The nature of the underlying mathematical formulation . Our own work together with many others [7, 14, 8, 16, 22, 24, 23, 94, 95, 25] relies on classical elasticity theory to predict the behavior of the fabricated objects. However, over the years, many works have also addressed kinematic [96, 97, 98, 5, 99], geometric [100, 101, 102, 103, 104, 105, 106] or light-interaction [107, 108, 109, 110, 111] design problems.

The de nition of the design goal . Our work belongs to a family that focuses on functionality and fabricability [14, 88, 97, 16, 22, 24, 23, 94] but many other works address questions like appearance and aesthetics [108, 112, 113, 50, 114, 111, 32] or manufacturing and material cost-e ciency [31, 115, 116, 117].

The solution approach . Some works on computational design consider mostly forward solutions [14, 118, 30, 29, 119, 50]: the user iteratively modi- es the parameters of the model while the computer provides a prediction of

the resulting performance. More recent solutions have also explored inverse approaches [120, 1, 10, 2, 12, 22]: the user speci es the desired result and a numerical optimization algorithm outputs a valid solution such that the corresponding behavior is as close as possible to the goal. Most current works however propose a combination of the two [97, 8, 16, 5, 23, 25].

Interactivity and user interaction . Methods can be classi ed into those that do require user interaction [14, 8, 16, 97, 16, 5, 25, 23], and those that do not [88, 1, 10, 2, 12, 22]. It is also important to highlight those methods that provide output feedback at interactive rates [26, 121, 31, 122, 15, 23, 123, 124].

Recently, Bermano et al. [125] presented a very thorough review of fabrication-aware design where works are also classi ed attending to several criteria such as object representation (e.g., surface, volume), designed attributes (e.g., shape, material) and goals (e.g., appearance, deformation, motion). We refer the reader to this work for further details. In the following pages we will focus on the the extensive literature on computational design that is most relevant to the work developed in this thesis. We will start going through some geometric problems focusing on surface design. Then, we explain several works dealing with animation control, material characterization and constrained space exploration, and how they are tangentially related to our problem. Finally, we will focus on fabrication-oriented design, grouping the extensive body of work according to how it is related to ours.

2.2.1 Surface design

Our work on KPS presented in chapter 6 targets the design of a class of physical surfaces whose shapes are governed by a particular set of equilibrium constraints. Surface design has been in the focus of computer graphics ever since its beginnings. Numerical problems associated with this works are similar to our formulation, although not all of this problems originate from mechanical simulation and they do not explicitly consider the fabricability of the designed surface.

For instance, developable surfaces have attracted a lot of attention from computer graphics [101, 102, 126, 127, 128]. They arise naturally when creating 3D surfaces from at, inextensible material such as plastic, paper, or sti fabric and so are closely related to fabrication-oriented design. As an example, based on the principle of auxetic materials, Konakovic et al. [129] were able to create doubly-curved surfaces by structuring planar sheets of quasi-inextensible material. This problem is also closely related with surface parameterization, which has many applications in computer graphics, for instance, for texture mapping. A good survey on the topic by Floater et al. can be found in [130]. While developable surfaces are characterized by having zero Gaussian curvature, requiring vanishing mean curvature leads to minimal surfaces

Minimal surfaces have been intensively studied in mathematics [131, 100] and also computer graphics [103, 104, 105]. In the context of architecture and engineering, minimal surfaces appear naturally when designing tensile membrane structures; see, e.g., [132]. Beyond minimal surfaces and the related Plateau problem, there has been an increasing interest in the generalized Plateau problem which, instead of assuming a rigid boundary, considers the case of Euler elastica [133] and Kirchho rods [134]. While these works focus primarily on questions of existence and uniqueness, in our work [23], we investigate the problem of modeling such surfaces for the purpose of fabrication.

Apart from digital surfaces made for virtual worlds, one important physical application domain is architectural geometry [106]. Surfaces from this category are often subject to constraints relating, e.g., to the planarity of polygonal faces [135, 136] or to compression-only self-supporting structures [27, 137]. In addition to enforcing such constraints numerically, exploring the resulting design spaces is a challenging problem as well [26, 15]. One particular line of recent work [30, 29] has studied the forward design of membrane structures coupled with bending-active elements. However, to the best of our knowledge, the inverse problem of automatically determining parameters such that the resulting equilibrium shape approximates given design goals has not been investigated so far.

2.2.2 Animation control

Most current animation systems rely on physically-based methods for the realistic depiction of real-world materials such as hair, cloth or uids. Experts working on video games and the VFX industry are responsible for tuning the corresponding simulation parameters so that the nal behavior responds to some artistic intention. This might result in a very slow trial and error process. As a consequence, the problem of optimizing external forces, material parameters and rest shapes to achieve a certain mechanical behavior has also been explored in the context of animation.

As an example of early work, Kondo et al. [138] controlled the deformation of elastic objects by keyframing rest shapes. Very similar methods have been used for the same purpose focusing on other mechanical models. For cloth simulation, for instance, Twigg and Kacic-Alesic [139] computed rest lengths for mass-spring systems in order to achieve desired garment drapes under gravity. Mainly applied to hair simulation, the same problem has been investigated for a number of di erent curve and rod models, including 2D elastic curves [140], articulated rigid body chains [43] and the super helices rod model [141]. In the context of character animation, Coros et al. [142] show that it is possible to create autonomous characters modeled as elastic objects by optimizing their rest shapes as a function of high-level motion goals. More generally, the methods described by Martin et al. [88] and Schumacher et

al. [143] are also related to the problem of modulating rest shapes in order to e ect the behavior of dynamic simulations, but solve the problem by adding a potential energy term speci ed relative to a set of input example shapes. One particular line of work has focused on the interactive editing of deformable simulations using techniques such as space-time methods or rotation-strain reduced coordinates to efficiently handle the high computational load [121, 122]. Finally, as an alternative to rest-shape design, di erent material optimization solutions have also been proposed with the same purpose. These works employ a variety of techniques such as model reduction [144] and optimization of principal stretches [145].

2.2.3 Material characterization

Correctly predicting object deformation requires i) a computational model that is capable of representing the material behavior, and ii) an accurate estimation of the parameters of the fabrication material. For the development of this thesis, we had to estimate material parameters from data. Additionally, the mathematical formulation associated with this task is equivalent to the one of a design problem. Basically, we look for object attributes (in this case, material parameters) such that the predicted deformation matches a target observed behavior.

Some works in computer graphics have addressed the data-driven estimation of material parameters specially in the context of computational cloth. Early ones [53, 54] were based on tting parameterized functions of the deformation to measured data, extracted from comprehensive sets of experiments with their corresponding machinery {e.g., theKawabata evaluation system[146]. Later, a few works improved previous results by measuring complex 3D deformations instead of just considering one-dimensional force-displacement curves. For instance, in the work by Wang et al. [147], a piecewise linear elastic cloth model is proposed and its parameters are tted to experimentally acquired data using a simple setup. On the contrary, Miguel et al. [148] proposed a more powerful and sophisticated solution where nonlinear stress-strain curves are numerically optimized considering also the loading forces over boundary conditions.

In an alternative line of research, some works have proposed more inexpensive acquisition processes, like extracting parameters from casually captured videos [149, 150, 151]. Their aim was to avoid the need for controlled conditions, but as a consequence, it was not possible to separate internal (i.e., material-speci c) from external (e.g., friction, air drag) parameters. More recent works have precisely focused on modeling such phenomena and trying to estimate their corresponding parameters. A pair of examples are the work by Miguel et al. [68] on internal friction and the one by Xu et al. [152] on damping.

2.2.4 Exploration of constrained spaces

Many of the previous works feature interactive applications and require the user to participate {at least{ in part of the design process, e.g. [97, 8, 153, 154, 124, 123]. However, part of this literature that deals with highly constrained problems has focused more on speci cally providing methods for the exploration of such spaces.

Many works on constrained geometric modeling are based on using optimization methods to minimize the distance from a constrained design to a target shape. This is the case of the work by Bouaziz et al [155], where a shape proximity function and projection operators are de ned to perform constraint-preserving changes on arbitrary geometric data sets such as curves, polygons or volume elements. Another case is [26], which explored such constrained manifolds through rst- and second-order approximations and applied this method for the design of planar quad meshes. In addition to allowing manual speci cation of target shapes, authors often leverage modal analysis to directly explore feasible solutions parametrically close to a given design. Freeform architectural design has made good use of early constrained mesh exploration approaches. For instance, Deng et al. [15] proposed a method where the numerical optimization is split into a sequence of simple subproblems that can be solved e ciently and accurately. Our own work on KPS (chapter 6) is greatly inspired by all these contributions, as we also propose a two-step optimization procedure based on a rst-order approximation to the constraint, and we also feature modal analysis.

In general, all these methods assume that the user is responsible for the creation of the design topology and focus on the continuous optimization of low-level local geometry. However, there are some applications for which high-level structural features are also essential. Structure-aware shape processing addresses the problem by dealing with the global inter- and intra-semantic relations among the parts of a shape rather than on their local geometry. For an extensive review on this topic, we refer the reader to the work by Mitra et al. [156]. Structural relationships have been also studied in the context of di erent design problems. For instance, Guerrero et al. [119], propose a method to explore the variability of 2D geometric patterns, using di erent interpretations of their regularity that correspond to different design variations. The work by Umetani et al. [118] proposes an interactive solution for the guided exploration of physically valid shapes in furniture design. While the user focuses on the aesthetics, the framework helps to achieve physical realizability. This is done by generating multiple suggestions involving both discrete and continuous changes to restore validity when one or more constraints are violated. Generation of plausible furniture variations was further explored by Zheng et al. [157], where symmetric functional arrangements of substructures are identi ed, allowing the combination of parts across di erent model families. More recently, similar ideas have been applied to the creation of mechanical objects. In the work

by [158], for instance, the authors present an interactive system that {using 3D models and high-level functional rules (e.g., A ts in B){ is able to optimize the shape geometry to produce a working design. In our works, the topology of the designs is either automatically generated, chapter 5, or delegated to the user, chapter 6. However, the implementation of a variation of some of these methods has been considered for future work.

2.2.5 Fabrication-oriented design

Designing fabricable objects whose properties can be intuitively speci ed and controlled is an important research challenge that is quickly gaining interest in the computer graphics community. As mentioned above, a great variety of performance criteria have been considered.

In the following, we will focus only on functionality goals in the form of motion and deformation behavior, as they are closer to our particular work. A variety of methods that investigate these aspects of design for rigid objects have been proposed. For example, there are computational design methods for creating objects that can stand on their own [1], spin stably [2] or sound under contact in a speci c manner [3]. Thanks to recent works, it is also possible to create articulated 3D-printable representations of virtual characters [96, 159], design mechanical characters capable of interesting motions [97, 160, 4, 99] and produce 3D-printable robotic creatures [5] or telescoping structures [6]. Our work is closer to methods that control the deformation behavior of elastic objects {a problem that has received increased attention from the research community over the last few years. As also happened with animation control, some of these works dealt speci cally with *material* design while others have focused more on the *shape* of the object.

Material design. Current fabrication methods allow the use of multiple materials within a single printed object. Many works have taken this technology as the basis of their approach. For instance, Vidimce et al. [161] proposed a framework for the fabrication of objects composed of multiple materials and Bickel et al. [7] used a small set of template materials with di erent deformation behaviors to fabricate objects whose force-deformation response matches measurements of real objects. Similarly, Skouras et al. [10] described an optimization method for computing an inhomogeneous distribution of material parameters to control the way in which fabricated elastic objects deform under the in uence of external forces.

Shape design. The rest con guration of elastic objects can also be computed using automated methods. For instance, Skouras et al. [8, 16] showed that the shape of in atable balloons and more complex structures can be controlled to match input targets by optimizing the rest state of membrane-based

models. Similarly, the rest state of volumetric objects can be modi ed to control the deformation behavior of skin for robotic faces [120], or to specify the way in which elastic objects deform under gravity [9]. Many design problems based on single-material fabrication technologies require changing the shape of the object at a small scale to control deformation properties. Such is the case of the works developed in this thesis, where the sti ness of the rod structure is controlled by modifying rod radii. Microstructures is a related research line that has attracted increased attention over the last few years, see, e.g., [12, 162, 163, 13]. These works propose methods for fabricating deformable objects with spatially varying elasticity using 3D printing and a single relatively sti material. They employ assemblies of small-scale structures with varied geometries that have an e ect on global material compliance.

In either case, both material design and shape design problems reduce to identical mathematical formulations and can be treated in a generic manner. For instance, Chen et al. [164] presented a uni ed framework to fabricate objects with controllable deformation properties and Musialski et al. [165] proposed a series of numerical improvements that can be applied independently of the optimized features.

In general, the deformed shape of these elastic structures is governed by equilibrium conditions, thus requiring a balance between internal {elastic{ forces and external forces such as self-weight and applied loads. The problem of designing deformable objects that assume desired equilibrium shapes under gravity has been extensively studied in computer graphics, e.g. for hair [140] and cloth [139] animation. In the context of fabrication-oriented design, similar problems have been addressed applied, for instance, to self-supporting surfaces [166] and custom-shaped elastic solids [9]. Our work shares many aspects of these inverse problems. However, our KPS design, chapter 6, di ers in the sense that the driving force is also membrane stretch, which, unlike external loads, depends on the state of the system.

For completeness, let us brie y mention other performance criteria that have been considered in the literature. Some of these works have focused on appearance goals. For instance, Shuller et al. [114] proposed a uni ed framework to create surfaces that depict certain 3D shapes from prescribed view points. Consideration for aesthetics has also been seen in works dealing with the design of furniture [123] or ornamented decorative pieces [50]. Appearance goals are usually also combined with some high-level functionality. For instance, the work by Dumas et al [113] deals with the synthesis of structurally sound patterns from sample data, while Martinez et al. [112] use topology optimization to combine structural and appearance objectives into the same shape design. A completely di erent research line focuses on digital fabrication and deals instead with fabricability and material e ciency design goals. Some example works are the cost-e ective printing of 3D objects with skin-frame structures by Wang et al. [31], or the more recent design of lightweight structures

under force location uncertainty by Ulu et al. [117].

The following section nally reviews a few works dealing with two speci c representations of fabricable objects which are specially related to the work developed in this thesis: physical surfaces and rod structures.

Physical Surface Design

Most works in computer graphics related with physical surface design have been restricted to at panels, with applications to plush [167, 154], garment design [14, 168], rubber balloons [8], in atable structures [16], thermo-formed models [114], surfaces composed of interlocking elements [153] and tensile structures [30, 29], among others. Most of them use a similar approach to the works developed in this thesis, with slight di erences depending on the speci c application.

For instance, rather than automatically generating a rod network for an input surface as we do in chapter 5, Skouras et al. [16] leave the topology problem to the user and instead optimize performance such as to enable fast design iterations. Our approach in chapter 6 follows the same spirit, but instead of optimizing for a xed target shape, our method allows the user to explore the space of feasible designs using a set of editing tools that leverage simulation and optimization. This same paradigm of computer-assisted but user-guided design is implemented by several previous approaches, including the work by Umetani et al. [14], who use rst-order sensitivity analysis in order to quickly predict the impact of parameter changes on the equilibrium shape of clothing. As one di erence, our method extends this forward design approach with inverse modeling tools that, for user-speci ed editing objectives, automatically compute rst-order optimal directions in parameter space. Allowing the user to directly edit the 3D equilibrium state was also the driving motivation for the work of Bartle et al. [168]. However, while their method uses a heuristics-based, gradient-free approach speci cally tailored to the problem of pattern optimization for garment modeling, we capitalize on derivative information in order to implement fast forward and inverse design tools. Finally, we want to highlight the contemporary work by Guseinov et al. [94] on tension-actuated at plates, which shares many similarities on the functionality goal to our own KPS design. While their solution allows to obtain doubly curved surfaces at the price of a more complex fabrication method, our results can be easily 3D printed but are restricted to piece-wise minimal surfaces.

Non planar surfaces mostly arise in architectural geometry, which is a very important application domain of computational design, specially in the context of freeform surface modelling. Although these approaches are not always directly applicable to real-world scales, most of these works feature fabricated samples to demonstrate the validity of their solutions. As a consequence, they share similar challenges to

the ones described in this thesis. Generally, structures are mostly made of rigid materials and their design consider a combination of performance criteria including appearance, functionality and fabrication e ciency. For instance, the creation of surfaces composed of an assembly of polygonal faces has been extensively studied for their bene ts in terms of material transport and storage [135, 136, 15]. Similarly, many works have dealt with the design of masonry structures, in which case, the low exural strength of building mortar is countered using compression-only {self-supporting{ designs [27, 137, 169]. For a more complete survey on this topic, we refer the reader to the review by Pottmann et al [106].

Rod Structure Design

Our rod meshes share some similarities with truss structures, which have been addressed in the context of digital fabrication. Wang et al. [31], for instance, proposed a computational design method that optimizes a truss structure to minimize the amount of printing material used in the fabrication of rigid objects. Constructions based on the same principle are known in architectural geometry literature as space structures and has also received attention from the computer graphics community, e.g., [28]. Apart from truss structures, more speci c fabrication-oriented design problems have been considered over the last few years. For instance, Song et al. [33] proposed an interactive design tool for creating stable networks of interleaved rods (known as reciprocal frames), while Gauge et al. [170] addressed the design of physical characters using tensegrities {networks of rigid rods and elastic springs that are in static equilibrium.

Objects created by connecting and bending wires are also very common in furniture design, metal sculpting or wire jewelry, and have attracted increased attention over the last few years. For instance, Liu et al. [32] proposed a method for the image-based reconstruction of such structures, while Garg et al. [171] developed techniques for designing wire meshes with prescribed shapes. Wire meshes can be considered a special case of rod networks, but rather than physics-based optimization, their particular structures motivate a geometric approach. The approach by Miguel et al. [24] targets physical surface representations using bent wires that securely interlock by virtue of deformation. Our approach on KPS design shares the two-dimensional nature of their design space, but the underlying mechanics are very di erent. Instead of designing 3D networks that balance applied loads, our method computes a 2D layout optimized with respect to membrane forces.

While most of the above methods do not consider automatic topology generation, Zimmer et al. [172] focus exactly on this discrete problem, albeit in a context that does not involve equilibrium constraints. Similarly, the work by Zehnder et al. [50] faced the same problem and presented a computational tool for designing ornamental curve networks {structurally-sound surfaces with user-controlled aesthetics.

Chapter 3

Mechanical simulation

Our design tools build on physics-based animation in order to predict the deformation behavior of flexible structures in response to applied forces. In this chapter, we overview the relevant foundations of mechanical simulation and describe in detail the mechanical models and numerical solvers that we have developed in this thesis. Our main contribution here is the creation of a novel mechanical model for the simulation of flexible rod meshes in 3.2.2.

The first section 3.1 briefly reviews the foundations of mechanical simulation, formalize the mathematical problem and establish a common framework for the definition of mechanical models.

The second section 3.2 explains in detail the model that we have devised for the simulation of rod meshes, preceded by a description of the discrete elastic rods model upon which it is based.

The third section 3.3 describes the discrete shells model that we use for the simulation of the tensile fabric, with special emphasis on the orthotropic constitutive model employed for the elastic membrane.

The fourth section 3.4 concludes by explaining in detail the numerical methods that have been used for the computation of static equilibria and analyze potential problems of their practical implementation.

3.1 Foundations

Let us assume there exists a continuum mechanical model $\mathcal{M}(\mathcal{O})$ that determines the elastic behavior of a deformable object $\mathcal{O} \subset \mathbb{R}^3$ through a conservative energy

potential $V(O) \in \mathbb{R}$. Simulating the statics of this mechanical system implies solving the nonlinear system of differential equations defined by net force equilibrium $f = 0$ for any applied external forces and boundary conditions. Finding a computational solution to these equations using simulation requires defining discrete expressions for both the kinematics of the object and the potential energy function.

Discrete kinematics We consider a generic Lagrangian discretization in the form of a vector $q = \{q_1, \ldots, q_n\}$ in some known generalized coordinates system. This vector contains n independent variables that spatially approximate the state of the object in different configurations:

> We denote the *deformed* configuration of the object as q and refer to the space of all possible deformed configurations $q \in \mathbb{R}^n$ as *deformed* or *world space*, Q.
>
> We denote the *undeformed* or *rest* configuration as q and refer to the space of all undeformed configurations $q \in \mathbb{R}^n$ as *undeformed* or *material space*, U

Discrete energy Given a spatial discretization, an approximation of the potential energy must then be formulated, discretely parameterized by the deformed configuration q, $V(q) : \mathbb{R}^n \to \mathbb{R}$. Note that we are omitting here other magnitudes that might be also affecting the value of the energy potential (e.g. gravity constant, undeformed configuration material properties, etc.) but whose values do not change during the simulation. The discrete energy formulation allows deriving the expression of the system *forces* as the negative gradient of the potential energy V w.r.t the deformed configuration q, $f(q) \in \mathbb{R}^n$:

$$f(q) = \nabla_q V^T = \left[\frac{\partial V}{\partial q_1}, \ldots, \frac{\partial V}{\partial q_n}\right]^T. \tag{3.1}$$

Similarly, system forces Jacobian can be calculated as the matrix of negative second partial derivatives of the potential energy V w.r.t the deformed configuration q, $J(q) \in \mathbb{R}^{n \times n}$:

$$J(q) = \nabla_q^2 V = \begin{bmatrix} \frac{\partial^2 V}{\partial q_1 \partial q_1} & \cdots & \frac{\partial^2 V}{\partial q_1 \partial q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 V}{\partial q_n \partial q_1} & & \frac{\partial^2 V}{\partial q_n \partial q_n} \end{bmatrix}. \tag{3.2}$$

Note that, in our convention, we consider the partial derivatives of a scalar function to be a row-vector. With a close expression for both the force and the Jacobian of the mechanical model, the expression of the force can be linearly approximated at any point q_0 using Taylor's expansion:

$$f(q) \approx f(q_0) + J(q - q_0) \tag{3.3}$$

Mechanical model With these ingredients, standard numerical methods can be used to solve the nonlinear systems of equations resulting from the static equilibrium. In the following sections, we will describe the different mechanical models we have used in this thesis in terms of these two components. First, we will overview the continuum mechanics formulation associated with the model. Then, a spatial discretization approach will be introduced followed by a reformulation of the smooth energies into discrete expressions.

3.2 Rod mesh model

Rods are curve-like elastic bodies that have one dimension ("length") much larger than the others ("section"). As mentioned in chapter 2, among the many existing computational approaches for the simulation of rods, we choose the discrete elastic rod (DER) model by Bergou et al. [45, 46], due to its compact curve-angle representation with explicit centerline. We note, however, that other approaches based on reduced coordinates, e.g., [42], or full coordinates with constraints, e.g., [20], would be possible as well.

In the following section 3.2.1, we briefly summarize the most relevant part of the theory regarding Kirchhoff rods and discrete elastic rods. Since the model by Bergou et al. does not account for coupling among rods, in section 3.2.2, we propose a model for rod connections, formulate elastic energy terms, and describe how to correctly transmit bending and twist forces across connections.

3.2.1 Discrete rod model

In this section, we briefly review classic Kirchhoff rods theory as well as the reduced coordinate formulation of discrete Kirchhoff rods developed by Bergou et al. [21, 46]. We refer to the original works for a more thorough explanation.

Smooth Kirchhoff rods

Classic Kirchhoff rods are conveniently modeled as adapted framed curves with an arc-length parameterized centerline $c(s) \in \mathbb{R}^3$ and an orthonormal material frame $m(s) = [t(s)\ b(s)\ n(s)] \in SO(3)$, where t is the tangent, b, is the binormal and n is the normal. We assume that s is an arc-length material parameterization of

the undeformed con guration of the adapted framed curve. Recall that we will denote quantities associated to the undeformed con guration of the material with an overline, e.g. $\bar{c}$. The orthonormal material frame assigned to each point of the centerline spans the plane normal to the centerline's tangent called the cross-section, and contains the requisite information for measuring twist. Since the material frame is adapted to the centerline, the following condition must hold $t(s) = c^0 = \bar{c}^0$, where the prime indicates di erentiation with respect to the s coordinate.

The Kirchho theory of elastic rods de nes scalar functions that measure the strain of any adapted framed curve {given by the change of the orthonormal frame expressed in its own coordinates and the deformation gradient w.r.t. a naturally unitary straight rod:

$$= k\bar{c}^0 k; \quad !_b = \quad b; \quad !_n = \quad n; \quad m = b^0 \quad n; \tag{3.4}$$

where $= t^0$ is the centerline's normal curvature vector. Here, the rst term, , corresponds to the relative axial strain. The second and third terms $!_b$ and $!_n$, represent the rod's curvature vector expressed in material coordinates and measure the bending of the material frame. And the last term, m, refers to the twist of the material frame around the tangent. Accordingly, the total elastic energy contains stretching, bending and twisting contributions, $V = V_s + V_b + V_t$.

The stretch energy is de ned as

$$V_s = \frac{1}{2} \int k_s(\quad)^2 ds; \tag{3.5}$$

where k_s is the stretch sti ness.

The bending energy term is de ned as

$$V_b = \frac{1}{2} \int (! \quad !)^T B (! \quad !) ds; \tag{3.6}$$

where $! = (!_b; !_n)$ represents the centerline curvature vector and B is a symmetric positive de nite $2 \quad 2$ matrix representing the bending sti ness. This formulation allows for an anisotropic bending response in the two main directions of the cross-section.

Finally the twist energy term is de ned as

$$V_t = \frac{1}{2} \int (m \quad m)^2 ds; \tag{3.7}$$

where is the twist sti ness.

Note that this model generalizes to naturally non-unitary, curved or twisted rods by subtracting away the undeformed centerline axial strain, curvatures $!$ and twist m. The particular case of an isotropic, naturally straight rod is obtained by taking $B = k_b I_{2\ 2}$, $! = 0$ and $m = 0$.

Curve-angle formulation

The main contribution of Bergou et al. [21, 173] is the development of a reduced coordinate formulation of thin Kirchho rods, which is based on a previous analysis introduced by Langer and Singer [174]. In the curve-angle representation, the material frame is expressed through a rotation roll (s) from an adapted orthonormal reference frame $\underline{m}(s) = [\underline{t}(s)\ \underline{b}(s)\ \underline{n}(s)]\ 2\ SO(3)$ which will be further explained below. More precisely, material frame can be explicitly computed as $\underline{m} = R(\underline{t};\ )\ \underline{m}$, where R is the rotation around the vector $\underline{t}$ by an angle . When applied to an orthonormal frame, this operation reduces to:

$$\begin{aligned} b &= \cos\ \ \underline{b} + \sin\ \ \underline{n}; \\ n &= \ \ \sin\ \ \underline{n} + \cos\ \ \underline{b}; \end{aligned} \tag{3.8}$$

with $t = \underline{t}$. Like other reduced coordinate models, this avoids the need for sti constraints that couple the material frame to the centerline, while at the same time, the explicit centerline representation facilitates collision handling and rendering.

Discrete rod kinematics

In the discrete setting, the kinematic state of the rod results as follows: the centerline of the rod is represented by the centerline as a piece-wise linear curve de ned by a set of n nodes $c = \{c_i\}$ and material frames are represented by a set of roll angles $\{_i\}$ w.r.t. the reference frames that pertain to the edges of the centerline $e_i = (c_{i+1}\ \ c_i)$ (Fig. 3.1).

Whenever there is a change in the kinematic state of the rod, the reference frame $\underline{m}$ should be updated so that the material frame m remains adapted. In discrete elastic rods, this operation is performed by parallel transport of a frame initialized to the material frame in the rest state. Parallel transport is an important concept in the discrete elastic rods model, which serves to transform a frame adapted to a source edge into a destination edge. Given two unit vectors v_1 and v_2, the parallel transport is de ned as the minimum rotation that aligns v_1 with v_2:

$$P(v_1; v_2) = R(v_1\ \ v_2 = \|v_1\ \ v_2\|;\ \ (v_1; v_2)); \tag{3.9}$$

where $R(v;\ )$ is the rotation about the unit vector v by an angle , and $\ (v_1; v_2)$

Figure 3.1: Discretization of the centerline as a piece-wise linear curve de ned by a set of nodes c_i and edges e_i. In the curve-angle representation, the material frame m_i (red) is de ned as a rotation about the tangent of a reference frame $\underline{m}_i$ (blue) by a roll angle $_i$.

is the angle between the vectors v_1 and v_2. Depending on whether reference frames are parallel transported in space or in time, the resulting formulation o ers di erent computational advantages:

Parallel transport in space Initially, Bergou et al. [21], suggested using the geometrically most natural (and physically most relaxed) frame adapted to a curve: the Bishop frame. For a given centerline, this is an adapted frame with zero twist uniformly, and its evolution along the centerline corresponds in nitesimally to a rotation about the curvature binormal. In the discrete setting, this implies parallel transporting frames between adjacent edges $\underline{m}_k = P(e^t_{k\ 1}; e^t_k)\,\underline{m}_{k\ 1}$. During the simulation, the reference frame is subsequently updated starting from the frame at the beginning of the rod $\underline{m}_0$. The state of the k-th material frame of the rod is dependent on the kinematics of the (k 1) previous centerline edges. Consequently, the force stencil is not local and the resulting Jacobian is dense. Although this limits the applicability of this method, authors claim that, under speci c conditions {i.e. isotropic bending response, straight undeformed con guration, quasistatic material frame and explicit integration{ there is no computationally more e cient solution.

Parallel transport in time Posteriorly, Bergou et al. [46], suggested that the reference frame would also evolve naturally in time, rotating by the minimum amount needed to keep itself adapted; this corresponds to a frame having zero tangential angular velocity. In the context of static equilibrium solving, \time" will refer to the subsequent iterations of the quasistatic solver. In the discrete setting, this is equivalent to parallel transporting the reference frame

between two time-instances of the same edge $\underline{m}_k^{t+1} = P(e_k^t; e_k^{t+1})\, \underline{m}_k^t$. During a simulation, this evolution can be done incrementally {only the reference frame at the current time step must be known to advance it to the next time step. The resulting force stencil is local and the Jacobian is sparse, enabling the use of implicit integration. Given the particular needs of our problem, we adopt the second approach.

With this reduced-coordinates formulation, the twist of the rod depends only explicitly on the set of twist angle coordinates. However, parallel transport establishes an implicit link between the kinematic state of the centerline and the twist angle, which must be considered during force and Jacobian derivation.

Discrete elastic energies

With the discretization, the elastic energies introduced before result:

1. Stretch . For an edge between nodes c_1 and c_2, with rest-length l_0, and stretch sti ness k, the stretch energy depends only on centerline variables and can be computed as follows:

$$V_s = \frac{1}{2} k_s l_0 \quad 1 \quad \frac{kc_2 \quad c_1k}{l_0}^{\;2} : \tag{3.10}$$

2. Bending . Recall that the elastic bending energy in the smooth case was dependent on the curvature of the centerline. Since each edge is straight, it follows that discrete curvature is associated with vertices. For two adjacent edges with material frames $m_1 = [t_1; b_1; n_1]$ and $m_2 = [t_2; b_2; n_2]$, the discrete curvature binormal at the shared vertex is a vector orthogonal to the osculating plane passing through the edges and can be computed as:

$$(\quad b) = \frac{2t_1 \quad t_2}{1 + t_1^T t_2} : \tag{3.11}$$

It can be shown that the magnitude of this vector is $2\tan(_i/2)$, with $_i$ the turning angle between two consecutive edges. Note that this quantity is well-de ned in the case of collinear edges even though the binormal is not. Given this, the discrete material curvature at the shared vertex is de ned as

$$= \quad (\quad b)^T \frac{n_1 + n_2}{2}; \quad (\quad b)^T \frac{b_1 + b_2}{2}^{\;T} : \tag{3.12}$$

With a rest-state curvature $_0$ and anisotropic bending sti ness B (see material parameterization below), the resulting discrete bending energy between the

edges is

$$V_b = \frac{1}{2l_0}(\quad {}_0)^T B(\quad {}_0): \tag{3.13}$$

It should be noted that is de ned per vertex as a function of the incident edges and their material frames. Hence, depends on both, centerline positions and twist angles.

3. Twist . For the same two adjacent edges mentioned above, twist can be computed as the di erence in the material-reference roll angle between the two frames, i.e., ${}_2 \quad {}_1$, plus a reference twist introduced when parallel transporting the reference frame over time. For two subsequent steps in time $k \quad 1$ and k and two adjacent reference frames $\underline{m}_1$, $\underline{m}_2$, this computation results in:

$$= \backslash \quad P(\underline{t}_1^k; \underline{t}_2^k)\, P(\underline{t}_1^{k \quad 1}; \underline{t}_1^k)\, \underline{b}_1^{k \quad 1}; \underline{b}_2^k; \quad : \tag{3.14}$$

With a twist sti ness , the nal twist energy between the edges results in

$$V_t = \frac{1}{2l_0} \; (\; {}_2 \quad {}_1 + \quad)^2: \tag{3.15}$$

Note that twist forces acting on the centerline are non-zero, due to the in uence of the centerline on the reference twist . Similarly, bending forces acting on roll angles are non-zero, due to the in uence of these angles on the orientation of material normals and binormals.

Material parameterization

The elliptical cross-section of the rod is discretized at edges with normal and binormal radii r_n and r_b. The area of the resulting cross-section is $A = \; r_n r_b$. Knowing the Young modulus E and shear modulus G of the material, the stretch, bending, and twist sti ness can be computed, respectively, as [46]:

$$k_s = E A; \quad B = \frac{E A}{4} \begin{matrix} r_n^2 & 0 \\ 0 & r_b^2 \end{matrix} \; ; \quad = \frac{G A(r_n^2 + r_b^2)}{4}: \tag{3.16}$$

The mass of the edges is lumped at nodes $\{c_i\}$. For a material with mass density , the lumped mass on a node with incident edge volumes V_1 and V_2 is $m = \; (V_1 + V_2)/2$. Additionally, the model also considers the cross-sectional inertia of each twist angle $\{ \; {}_i\}$, which is $I = \; V r_n r_b/2$. Alternatively, $\{ \; {}_i\}$ variables might be assigned with zero inertia to ensure the material frames stay always in quasistatic equilibrium.

Simulation scheme

Independently of the nature of the simulation problem {whether we are solving dynamic deformation or static equilibrium{, generally each solver iteration provides a new guess for $\{c_i\}$ and $\{\ _i\}$. Once the new centerline is computed, we recompute the reference frame on every edge through parallel transport over time, and apply the roll angles to compute the new material frames. As noted before, the reference frame accumulates twist that must be updated after each iteration. For more details on the dynamic simulation algorithm, we refer the readers to the original work [173].

3.2.2 Rod network model

We propose an extension to discrete elastic rods model [21, 173] to handle connections between multiple rods. With the goal of avoiding the use of constraints, we devise an elastic energy that captures each rod deformation w.r.t. a rigid-body representation of the connection. Depending on how we keep track of connection kinematics, this framework leads to two alternative formulations:

> A co-rotational approach , that we de ned in [22], where the orientation of each connection is estimated kinematically from incident rods. We formulate their implicit relation and show how the elastic coupling energy must be derived to correctly transfer bending and twist forces between connected rods.
>
> An explicit approach , later used in [50, 23], where each connection keeps track of its own orientation with 3 additional DoFs in the form of Euler angles.

In the following sections, we will rst describe the co-rotational model, which bene ts from a reduced number of DoF. Then, we will brie y explain the changes that are necessary to implement the explicit approach.

Coupling energy

For each rod incident in a given connection, we seek to de ne an energy that captures its deformation relative to all other incident rods. We follow a co-rotational approach, measuring deformation potentials w.r.t. rigidly transformed connections. Speci cally, for each rod incident in a connection, we condense the e ect of the remaining rods into a single connection edge whose orientation is determined kinematically using best- t rigid transformations. Another possibility would be to de ne pairwise energies for the segments incident in a connection, as done in the Cosserat nets framework [47], but we found it more di cult to weight pair energy contributions to ensure a correct integration volume.

Let us consider a set of rod edges incident in a connection, with rest-shape material frames $\{m_i\}$. In the deformed setting, we estimate a rotation R of the rest-shape connection that best matches the deformed connection. Then, for each incident edge, we de ne a connection edge with material and reference frame $m_{c;i} = R\, m_i$, i.e., the rigidly rotated rest-shape material frame (Fig. 3.2). Note that, as material frames are already determined by the rotation of the connection, there is not need for a curve-angle discretization {i.e. reference and material frames are equal and hence roll angles at connection edges are trivially $\psi_{c;i} = 0$. However, the reference frame twist $_{c;i}$ between each incident rod edge and its corresponding connection edge must be considered as for individual rods.

Figure 3.2: At any moment, the rigid state of the connection is determined by a rotation matrix R which is computed using shape matching. For each incident edge reference frame $\underline{m}_i$ (blue) and corresponding material frame m_i (red), the rotation de nes a connection edge with material frame $m_{c;i} = R\, m_i$ (green).

Given material and reference frames of incident edges and their connection edges, we can compute bending and twist elastic energies at the connections based on the deviations between incident edges and their corresponding connection edges. For doing so, we use the very same formula in Eqs. (3.13)-(3.15) with a slight di erence: for each incident edge we use as integration domain half the rest length of the edge. Consequently the total potential elastic energy of the connection results:

$$V_c = \frac{1}{2} \sum_{i=0}^{k} V_b(m_i; Rm_{\ i}) + V_t(m_i; Rm_{\ i}); \tag{3.17}$$

where k is the number of incident rods in the connection. Note that the total domain of integration considered is half the sum of rest lengths of all edges incident in the connection. The deformation of each incident rod in the connection is a ected by its own material properties and the anisotropic edge radii. This allows geometrically

complex connections with heterogeneously thick incident rods to deform realistically according to the material, even for coarse discretizations (see Fig. 3.3 and Fig. 3.4).

Rotation estimation

In practice, we estimate the rotation R of a connection by minimizing the deformation of material frames of all incident edges. To consider the e ect of both the bending and twist deformation of incident rods on the connection rotation, we measure the deviation of the material tangent t and one of its orthonormal material axes n. Similarly to the shape matching approach by Mueller et al. [175], we rst estimate the linear transformation that minimizes the least-squares deformation energy:

$$A = \arg\min \sum_{i=0}^{k} \mathrm{Tr}(B_i)\|t_i - A\,t_i\|^2 + {}_i\|n_i - A\,n_i\|^2; \tag{3.18}$$

and then extract the rotation through polar decomposition $A = R\,S$. Weighting the deformations of tangents and their orthonormal axes with each edge's bending and twist sti ness respectively, we favor the alignment of thicker and sti er edges. Note that, for simplicity, both bending directions are considered together for the alignment of the tangent, and hence weights are not a ected by the anisotropy of the rod and are just proportional to its thickness, Young modulus and shear modulus. A more precise alternative might have been considered by taking into account also the binormal of the incident edge albeit at the expense of a much more complex derivation of coupling forces.

Figure 3.3: This example depicts force transfer at rod connections with our connection model. The axis of a windmill is twisted, and upon release its twist transfers into bending of the blades. A thinner axis (left) produces less momentum than a thicker one (right), as one would expect. The centerline of the axis was kept xed.

Coupling Forces

To compute the bending and twist forces at connections, the respective energy terms need to be differentiated w.r.t. the DoFs of all incident rod edges. The difficulty stems from the fact that the material frames are not explicit DoFs, they are kinematically defined by the best-fit rigid transformation as explained above.

As material frames depend through the rotation of the connection on all DoFs of incident edges, the bending and twist energies of an edge pair formed by an incident edge and its connection edge need to be differentiated w.r.t. centerline coordinates and roll angles of all the incident edges of a connection, producing non-zero bending and twist forces between all pairs of incident edges. This is actually the expected behavior, because bending of one incident edge may produce twist on other incident edges and viceversa. Fig. 3.3 shows how our model succeeds to represent the coupling between rods incident in a connection.

Even though the number of force terms and required derivative calculations grows w.r.t. the regular discrete elastic rods method, these derivatives are mostly built from the same basic blocks. The only exception is the need to compute partial derivatives of rotated rest-shape material frames $m_c = R\,m$, w.r.t. centerline coordinates and roll angles of all incident edges in the connection, as discussed in the paragraph above. By the chain rule, the differentials of rotated rest-shape material frames can be generally computed as:

$$d(R\,m) = \frac{\partial R}{\partial A} \sum_{i=0}^{k} \frac{\partial A}{\partial t_i} dt_i + \frac{\partial A}{\partial n_i} dn_i \quad m_i : \tag{3.19}$$

$\frac{\partial R}{\partial A}$ is the derivative of polar decomposition which can be found in the work by Barbic and Zhao [78], and $\frac{\partial A}{\partial t_i}$ and $\frac{\partial A}{\partial n_i}$ are constant and easily computed from the linear expression that defines A [175]. Derivatives dt_i and dn_i w.r.t. incident edges centerline and roll angles can be easily derived from the discrete rod kinematics formulation introduced in the previous section 3.2.1.

Finally, static equilibrium solving requires the computation of energy Hessians, i.e., force Jacobians. Other than derivatives already present in the original discrete elastic rod formulation, our rod networks require the computation of the second derivatives of polar decomposition. Barbic and Zhao [78] also derived the scalar second derivative of polar decomposition, but we require a matrix of second derivatives with mixed terms and follow a very similar derivation.

We denote with $sk(\omega)$ the $3 \quad 3$ skew-symmetric matrix of a vector $\omega \quad 2 \; R^3$, i.e., $sk(\omega)\,x = \omega \quad x; 8x \; 2 \; R^3$. Similarly, we denote with $sk^{1}(A)$ the unique skew-vector $\omega \quad 2 \; R^3$, such that $sk(\omega) = \frac{1}{2}(A \quad A^T)$.

Given a polar decomposition $A = R\,S$, its first derivative is:

$$\frac{\partial R}{\partial u} = \mathrm{sk}(\omega(u))\, R; \qquad \frac{\partial S}{\partial u} = R^T \frac{\partial A}{\partial u} \frac{\partial R}{\partial u} S \quad ; \tag{3.20}$$

$$\text{with } \omega(u) = 2\, G^{-1} \mathrm{sk}^{-1}\left(R^T \frac{\partial A}{\partial u} \right) ;$$

$$\text{and } G = (\mathrm{Tr}(S) I - S) R^T :$$

The mixed second derivative of the rotation matrix is:

$$\frac{\partial^2 R}{\partial u \partial v} = \mathrm{sk}\left(\frac{\partial \omega(u)}{\partial v} \right) R + \mathrm{sk}(\omega(u))\, \mathrm{sk}(\omega(v))\, R; \tag{3.21}$$

$$\text{with } \frac{\partial \omega(u)}{\partial v} = 2\, G^{-1} \mathrm{sk}^{-1}\left(R^T \frac{\partial^2 A}{\partial u \partial v} - \mathrm{sk}(\omega(v)) \frac{\partial A}{\partial u} \right)$$

$$- G^{-1} \left(\mathrm{Tr}\left(\frac{\partial S}{\partial v} \right) I - \frac{\partial S}{\partial v} \right) R^T \omega(u) + \mathrm{sk}(\omega(v))\, \omega(u):$$

Fig. 3.4 shows the application of our connection model to the dynamic simulation of a sphere-shaped rod mesh. The sphere moves down straight during compression, but then it twists up due to the mesh topology and the highly anisotropic bending behavior which favors tangent plane deformations.

Figure 3.4: Dynamics of a sphere with complex interplay of compression and twist. The mesh topology and the highly anisotropic bending behavior favor tangent plane over out-of-plane deformations.

Explicit formulation

A formulation extending our model was proposed by Zehnder et al [50]. In this approach, instead of implicitly inferring connection states using best- t transformations to incident rods, they explicitly modeled connections as mass-less rigid bodies with rotational DoFs. More precisely, the material frame at connection edges is computed as follows:

$$m = R_e(e_1; e_2; e_3)\, R_c\, m: \tag{3.22}$$

Here, $e = (e_1; e_2; e_3)$ are Euler angles which are used as rotational DoFs and R_e is the corresponding transformation. To avoid gimbal lock, after each step, the rotation from the Euler angles is transferred into a cumulative rotation R_c which is treated as a constant value. The total orientation is then the composition of the Euler angles and the cumulative rotation.

Despite the need of additional DoFs, derivatives of rigid-body rotations are computationally less expensive than calculating the expressions in Eqs. (3.19)-(3.21), and also more robust to extremely deformed con gurations. In general, this leads to better conditioned optimization problems. We adopt this alternative formulation in our own work in chapter 6, because the designed rod meshes are sparsely connected, and so the computational overload due to the additional DoFs is negligible.

3.3 Thin-shell model

Thin shells are thin exible structures with a high ratio of width to thickness (e.g. > 100). Thin-shell models are usually governed by *membrane* (stretch) and *ex-ural* (bending) energies formulated based on nonlinear continuum mechanics and discretized using FEM. In the following sections, we will explain in detail the two mechanical formulations that we have used to model the membrane and exural components of the energy:

Stretch (section 3.3.1). We use a *constant-strain triangle* FEM discretization with a St. Venant-Kirchho (StVK) constitutive model. Although the material that we employ for fabrication is almost isotropic, our implementation considers also orthotropic e ects [64] in order to accommodate other types of fabrics, if desired.

Bending (section 3.3.2). We adopt the *discrete shells* model by Grinspun et al [63]. In our setting of cloth simulation, internal forces are strongly dominated by membrane stress and exural stress is almost negligible. However, to avoid numerical problems when the fabric sags, we add weak bending forces.

3.3.1 Orthotropic membrane

A material is considered orthotropic if there exist three planes {orthogonal to each other{ by re ections with respect to which material properties remain invariant [176]. The axes normal to these planes are called principal material directions. This is the case for many fabric materials, as principal axes are usually in correspondence with warp and weft directions in woven fabrics, and with course and wale directions in knitted fabrics.

Smooth formulation

Elastic surfaces resist stretching (local change in area) and shearing (local change in length but not area). In the classical thin plates and shells theory, see e.g. [177], the orthotropic material behavior is described by the generalized Hooke's law assuming a linear relationship between the strain and the stress tensor.

We consider an elastic surface a 2-manifold embedded in the 3D space with planar undeformed con guration. Extending the notation introduced in section 3.1, the deformation of an object in the continuum is often characterized by a time dependent deformation map $: R^2 ! R^3$, from the material con guration $x \ 2 \ R^2$ to the deformed con guration $x \ 2 \ R^2$.

Continuum elasticity is generally built upon the deformation gradient $F = r_x$, i.e. the derivative of the deformation map w.r.t. material coordinates. In the case of the considered elastic surface, this is a $3 \ 2$ tensor. Based on the deformation gradient, rotation invariant measures are generally calculated to quantify the deformation, since pure rotation should not induce any stress in a deformable body. The most popular choice is the Green deformation tensor, which is de ned as $_{2 \ 2} = (F^T F \quad I)$. The material constitutive model then de nes the relation between such strain and the Cauchy stress $_{2 \ 2}$, which provides the elastic forces per length in a unit material direction n as $f_n = \quad n$. The St. Venant-Kirchho model states that the strain and the stress are linearly related through a fourth order compliance tensor C like $= C$. This relation can be written in matrix form referred to the principal material axes:

$$\begin{bmatrix} {}_{11} \\ {}_{22} \\ {}_{12} \end{bmatrix} = \begin{bmatrix} \frac{1}{E_1} & \frac{{}_{21}}{E_2} & 0 \\ \frac{{}_{12}}{E_1} & \frac{1}{E_1} & 0 \\ 0 & 0 & \frac{1}{2G} \end{bmatrix} \begin{bmatrix} {}_{11} \\ {}_{22} \\ {}_{12} \end{bmatrix}; \tag{3.23}$$

where we have unrolled the $2 \ 2$ symmetric matrices into 3-vectors. Components $11; 22$ are called normal components whereas component 12 is referred to as shear component. As the resulting matrix is block diagonal, it is easily invertible and the corresponding elasticity tensor (the so-called sti ness matrix) $= K$ is:

$$K = \begin{bmatrix} E_1 & E_1\,{}_{21} & 0 \\ E_2\,{}_{12} & E_2 & 0 \\ 0 & 0 & 2G \end{bmatrix} \tag{3.24}$$

with $= (1 \quad {}_{12}\,{}_{21})^{1}$. Unlike isotropic materials, which are parameterized with a single Young modulus E and a Poisson's ratio per dimension, orthotropic membranes require two Young moduli E_i, two Poisson's ratios ${}_{ij}$ and one shear modulus G. Moreover, under isotropy assumptions, shear modulus can be derived from Young modulus and Poisson's ratio, but when considering orthotropic materials, that condition does not hold. As not all parameter settings lead to stable simulations, we suggest following the approach by Li and Barbic [64] to ease material characterization. In this work, the authors take some assumptions and derive a new parameterization which is reduced to two Young moduli and a single Poisson's ratio. More, precisely, the following conditions must hold:

$$E_1 > 0; \quad E_2 > 0; \quad {}_{12}\,{}_{21} > 1; \quad \frac{{}_{12}}{E_1} = \frac{{}_{21}}{E_2}: \tag{3.25}$$

In general, these conditions derive from enforcing the symmetry and positive-de niteness of the sti ness matrix. Under such conditions, parameters are transformed and the nal formulation of the strain-stress relation is:

$$\begin{bmatrix} {}_{11} \\ {}_{22} \\ {}_{12} \end{bmatrix} = \begin{bmatrix} \frac{E_1}{\sqrt{E_1E_2}} & \frac{\sqrt{E_1E_2}}{E_2} & 0 \\ 0 & 0 & 2G \end{bmatrix} \begin{bmatrix} {}_{11} \\ {}_{22} \\ {}_{12} \end{bmatrix}; \tag{3.26}$$

$$= \frac{1}{(1+\ )(1\ )}; \quad G = \frac{\sqrt{E_1E_2}}{2(1+\ )};$$

where the simpli ed parameters are $2\ (0; 0{:}5)$ and $E_1; E_2 > 0$. Given this strain-stress relation, the corresponding membrane stretch energy density is computed as $V_s = \frac{1}{2}\ :\ $, where the operator : represents the scalar product between the two tensors (i.e., the coe cient sum of the coe cient-wise product):

$$V_s = \frac{1}{2} \sum_{i=1}^{2} \sum_{j=1}^{2} {}_{ij}\,{}_{ij}: \tag{3.27}$$

Expanding this expression and grouping terms, the elastic potential is:

$$V_s = \frac{1}{2} E_1\ {}_{11}^{2} + \frac{1}{2} E_2\ {}_{22}^{2} + \sqrt{E_1E_2}\ {}_{11}\,{}_{22} + (2G)\ {}_{12}^{2}:$$

When considering $E_1 = E_2$, this equation reduces to the classic St.VK elastic energy.

Discrete formulation

We represent the membrane as a 2-manifold triangle mesh $\mathcal{M}$ and discretize the continuum mechanics using a constant-strain triangle FEM. Information of the membrane state is stored on a finite number of 3D points $x_i \in \mathbb{R}^3$, with planar material configuration $x_i \in \mathbb{R}^2$. Finite elements specify a method for reconstructing a continuous deformation map $\hat{\phi} : \mathbb{R}^2 \to \mathbb{R}^3$ through interpolation from discrete values $x_i = \phi(x_i)$. Let us consider a generic scheme in which nodal values of a given variable, e.g., position x, are expressed throughout the element via polynomial interpolation as:

$$x(x) = \sum_{i=1}^{n_v} x_i N_i(x); \tag{3.28}$$

where n_v is the number of vertices in the element and each $N_i(x)$ (called shape function) is the interpolation weight associated with the i th node. Note that we explicitly show the dependence of the shape function on the material coordinate x corresponding to the interpolated point x. In matrix form, this results in $x(x) = D_w N(x)$, where all spatial positions of the element vertices have been assembled into a $3 \times n_v$ matrix $D_w = [x_1; \dots; x_{n_v}]$ and, similarly, shape function values have been assembled in a $n_v \times 1$ vector $N = [N_1; \dots; N_{n_v}]^T$. In this context, a discrete approximation of the deformation gradient can be computed as:

$$F = \frac{\partial x}{\partial x} = D_w \frac{\partial N(x)}{\partial x} = D_w B(x): \tag{3.29}$$

Here, B is the $n_v \times 2$ matrix of partial derivatives of shape functions w.r.t. material coordinates. Finite elements method highly depends on the particular choice of the interpolation function. In the case of a triangular mesh, barycentric interpolation would be the natural election to extend the nodal deformations to the entire interior of the mesh. For this particular case, the matrix B does not depend on the material coordinates of the point, and hence the deformation gradient has a closed form and can be computed as follows:

$$F = D_w G (D_m G)^{-1}; \quad G = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix} \tag{3.30}$$

where all material positions of the element vertices have been assembled into a $n_v \times 2$

matrix $D_m = [x_1; \dots; x_{n_v}]$. The value of the matrix G is derived from the material derivative of shape functions corresponding to isoparametric elements [178]. We found this formulation to be the most generic and extensible to other interpolation schemes and discretization elements, but there are other equivalent derivations for the deformation gradient. For instance, we suggest the reader the course by Sifakis and Barbic [179] for further information.

With a discrete approximation of the deformation gradient, it is then straightforward to compute the elastic potential energy in (3.28) and derive the corresponding force and Jacobian expressions needed for implicit integration.

3.3.2 Discrete shells bending

For completeness, we present here an overview of the thin shells bending model that we use. Flexure models in shell mechanics are usually based on the di erence of the second fundamental forms between the deformed and undeformed con gurations. This tensorial expression measures deformation and is invariant to rigid-body transformations of the coordinate frame. Same as with the membrane energy, these treatment derives expressions over smooth manifolds and then discretizes to carry out the numerics. But because of their degeneracy in one dimension, the development of robust nite element methods for thin shell-equations is still challenging.

In contrast, the work we adopt, discrete shells by Grinspun et al. [63], directly de nes a constitutive model based on discrete geometric operators applied over piecewise-linear surfaces. The resulting expressions are easy to implement and capture the deformation behavior of the cloth well enough to prevent wrinkling and improve the numerical conditioning of the simulation problem.

The discrete shells bending energy is based on the di erence between the shape operator {which measures the local curvature at a point on a smooth surface{ evaluated on the deformed and undeformed surfaces. Several discrete approximations to this magnitude can be found, e.g., in [180]. In discrete shells, the authors propose to discretize the shape operator at edges, measuring the dihedral angle in the deformed and undeformed con gurations. For a given edge $\hat{e}$, the dihedral angle is the angle between the vectors normal to its neighboring faces. Considering this de nition, the discrete exural energy is the summation over mesh edges of the expression:

$$V_b = \frac{\|e\|}{h_e} (\quad e \quad e)^2; \tag{3.31}$$

where h_e is a third of the average of the heights of the two triangles incident to the edge. Implementation-wise, this discrete energy is straightforward to implement, but depending on the method used for the measurement of the dihedral angle, it

might be necessary to include safeguards for numerical robustness.

3.4 Static equilibrium

This thesis deals with design problems for which the main performance criterion is the deformation behavior, more precisely, we consider the equilibrium of fabricated objects for any given boundary conditions and not how they dynamically behave. Technically, the static equilibrium is de ned by the shape with vanishing net forces, that is $f(q) = 0$. As introduced in section 3.1, $q \in R^n$ is the deformed con guration of the object in some known generalized coordinates system. The resulting force $f \in R^n$ is generally a nonlinear vector function that may include internal forces f_s (e.g., elastic forces) and external forces f_e (e.g., gravity force). In general, forces depend on the deformed con guration q, as well as on some other magnitudes (e.g., material parameters, gravity constant, etc.) that are kept constant during simulation.

Design methods rely on numerical simulation to predict the actual behavior of fabricated designs. For some speci c boundary conditions, we seek to compute the corresponding deformed con guration q in the most accurate and e cient manner possible. In the following sections, we will overview a two approaches for the implementation of the static equilibrium solver that might be considered and discuss their practical applicability.

3.4.1 Root- nding solver

Using classical Newtonian mechanics, computing the static equilibrium means solving a system of nonlinear equations or, equivalently, nding the roots of the vector-valued function f. The general approach is to use the Newton-Raphson method to calculate successively better approximations to the function roots. Roughly, given an initial estimate x_0, successive solutions are computed:

$$x_{k+1} = x_k - J^{-1}(x_k)\, f(x_k); \tag{3.32}$$

where $J = \nabla_x f(x_k)$ is the Jacobian of the forces w.r.t. generalized coordinates. In practical terms, that means iteratively solving linear systems of the kind $J_k \Delta x = -f_k$, where $\Delta x = (x_{k+1} - x_k)$ and functions dependent on generalized positions at the k th iteration have been shortened to f_k and J_k. In certain situations, the Newton-Raphson method is an extremely powerful technique and has local quadratic convergence.

Practical problems

However, there are practical considerations {related mainly with the properties of the Jacobian{ that might lead to the failure of the Newton-Raphson method. In the following, we review the main practical issues that must be considered when solving static equilibria, specially with the root- nding algorithms:

Unstable equilibria . An equilibrium is unstable if a small position perturbation will likely produce a large change in the energy of the system. Mechanical equilibria happen at the critical points of the potential energy functions governing the system. These can be either local minima, local maxima or saddle points. Both maximum and saddle points are representative of unstable equilibria. These situations are rather usual, specially in the case of mechanical models where highly anisotropic behaviors are present, and can easily produce buckling e ects For example, when a sti elastic rod is subject to axial compressive stress, buckling may occur and produce a sudden sideways de- ection. Although these are valid equilibrium con gurations, the static solver should avoid moving towards them. This is justi ed because not only they are physically unstable and do not represent a statistically probable state for a fabricated object, but they are also numerically related with inde nite points.

Inde nite points These correspond to points of the space for which the Jacobian is inde nite. Inde nite matrices are characterized by having both positive and negative eigenvalues. This limits the range of linear solvers that can be used to solve the linear system in (3.32), as most used direct and iterative methods require a positive semi-de nite matrix to properly function, e.g., Cholesky and Conjugate Gradient. In addition, an inde nite point indicates the presence of a saddle point in the potential energy function, and so an unstable equilibrium.

Stationary points . These are points of the generalized space at which the Jacobian is singular, and consequently the linear system in (3.32) cannot be solved, causing the algorithm to halt. Analytically, this implies that the null-space of the Jacobian matrix is non-empty, i.e., there exist directions in the space for which the value of the force function does not change. Therefore, the problem has in nite solutions. In mechanical simulation, this usually means that numerical constraints in the form additional boundary conditions must be imposed to avoid the singularity. For instance, as we have explained in the previous section 3.3.1, most elastic energy formulations are invariant to rigid-body transformations and hence translation or rotation steps must be prevented.

Ill-conditioning . Even if the Jacobian matrix is strictly non-singular and

positive de nite, mechanical models can usually lead to ill-conditioned problems. If the Jacobian is nearly singular or has a large condition number {ratio between the maximum and mimimum eigenvalue{ at some iteration, the linear system is not numerically well behaved for the computation of a solution. This might lead to bad estimates that may overshoot the step and diverge from the closest root. A recent work by Kannan et al. [181] has analyzed potential causes and solutions for ill-conditioned problems in structural mechanics.

Practical solutions

Here we identify two solutions that may overcome the problems mentioned above:

Regularization The challenges due to inde nite problems are usually addressed through linear system regularization. The simplest solution is solving the linear system $(J + I) \quad q = \quad f$ with increasingly higher values for , until the resulting matrix is strictly positive de nite. Nevertheless, there are other regularization schemes that may also help with ill-conditioning. For instance, the well-known Thikonov regularization formulates an ordinary least squares problem to solve the linear system and adds a regularization term in order to give preference to particular solutions with desirable properties. This leads to a generic linear system formulation $(J^T J + Q^T Q) \quad q = \quad J^T f$, for some matrix Q. Depending on the choice of Q, Thikonov regularization may improve the conditioning of the problem, thus enabling a direct numerical solution. For a more thorough study on regularization, we refer the reader to [182].

Step selection Even with the previous safeguard, overshoot and divergence from the roots might still happen due to the nonlinearity of the force function and poor initial estimates. To mitigate this problem, a step length selection procedure should be also used. A straightforward solution could be to consider a simple backtracking line-search in the q direction that ensures the reduction of a merit function . Most literature on nonlinear equation system solving suggests using the norm of the function, $= kfk$. More sophisticated line-search methods consider additional convergence conditions such as the Armijo rule (i.e., su cient decrease in the merit function) and the Wolfe condition (i.e., su cient decrease in the curvature) [183]. However, independently on the particular line-search scheme used, Newton-Raphson algorithm together with this merit function is equally attracted to local minima and saddle points, what negatively a ects the overall convergence towards good solutions. In the next section, we will introduce a very similar formulation that originates from Lagrangian mechanics and improves the robustness of the solver.

3.4.2 Optimization solver

As we have seen, the set of solutions to the static equilibrium problem correspond to the critical points of the potential energy function V. Therefore, we can reformulate the problem as the minimization

$$\min_{q} V(q); \tag{3.33}$$

which finds solution at the points where the gradient vanishes $\nabla_q V = -f = 0$. Minimization of the potential energy imposes a bias towards stable equilibrium configurations that avoid the problematic saddle points mentioned in the previous section. Formulating the problem as a nonlinear optimization offers the possibility of using a much broader range of numerical methods. In practical terms, we solve the problem in (3.33) using *sequential quadratic programming* (SQP), for which the step computation is formally equivalent to the root-finding Newton-Raphson method described above, {we refer to Nocedal and Wright [183] for formal proof. The only difference between the two is in the merit function used for the step length selection procedure, which in this case is $= V$, to guarantee that each step produces a valid estimate closer to a stable static equilibrium configuration.

Chapter 4

Inverse elastic shape design

Our tools feature inverse design operators that automatically determine attributes of the elastic structures such that their deformed shapes meet some user-speci ed goals. The associated numerical optimization problem has been repeatedly formulated throughout the extensive literature, having become a standard scenario. In this chapter, we present a formal characterization of such problem, describe some numerical solving methods, and analyze the challenges of their practical implementation. Our goal is to address some common issues across many design problems that t within this framework and discuss alternative solution approaches.

The rst section 4.1 formalizes the optimization problem to establish a common notation and characterize this problem based on its theoretical solution space.

The second section 4.2 analyzes the practical implications of solving this optimization, provides some guidelines on how to formulate design problems to favor nding better solutions, and introduce two solution methodologies.

The third section 4.3 focuses on computational solutions based on the exploration of the constraint manifold, with an emphasis on the usefulness of sensitivity analysis for that purpose, and discuss current issues with commonly-used standard optimization methods.

4.1 Problem characterization

In this thesis, we study the following generic problem: given a parameterized exible object, nd the design {i.e., parameter values{ that optimize some metric dependent on the static equilibrium deformation of the object, for some boundary conditions.

In this section, we extend the generic notation introduced in the last chapter 3 in the context of mechanical simulation and formally characterize the problem to serve as a baseline for further analysis. We then study the theoretical solution space to better understand the underlying mathematical problem.

4.1.1 Problem formulation

We consider a generic computational model $\mathcal{M}(\mathcal{O})$ that describes the mechanical behavior of a deformable object $\mathcal{O} \subset \mathbb{R}^3$ through a conservative energy potential $V(q;p) \in \mathbb{R}$. Here, $q \in \mathbb{R}^n$ concatenates all the independent variables that discretize the deformed shape of the object in some known generalized coordinates system, and $p \in \mathbb{R}^m$ is an arbitrary set of parameters that determines the design of the object for a particular problem. From here onward, we will refer as deformed space to the space $\mathcal{Q} \subset \mathbb{R}^n$ spanned by q, design space to the space $\mathcal{P} \subset \mathbb{R}^m$ spanned by p, and total space to the union of both $\mathbb{R} \supset \mathcal{P} \cup \mathcal{Q} \subset \mathbb{R}^o$ spanned by $r = (q;p)$, with $o = n + m$. This parameter set might contain variables modeling discrete geometry (e.g., the rest-shape) or material properties (e.g., Young modulus), among others. We are omitting here other quantities that might a ect the value of the energy potential (e.g., gravity constant), but whose value is not considered a design parameter.

Altogether, we then de ne the generic shape optimization problem as follows:

$$\begin{aligned} \min_{q_i;p} \quad & \sum_{i=1}^{k_p} h(q_i) \\ \text{s.t.} \quad & f_{q;i}(q_i;p) = 0 \\ & p_m \le p \le p_M . \end{aligned} \tag{4.1}$$

This formulation minimizes an objective function h dependent on a number of k_p deformed poses q_i, subject to the static equilibrium constraint $f_{q;i} = 0$ and box constraints on design parameters $p_m \le p \le p_M$. While the deformed con gura-tion q_i is generally unbounded, design parameters p are often restricted to valid ranges. For instance, materials are characterized by Young's modulus $E \in (0;1)$ and Poisson's ratio $\in (0;0.5)$. For this reason, lower p_m and upper p_M limits are considered. Note that the equilibrium constraint corresponding to each pose $f_{q;i}$ considers its own boundary conditions. This generic formulation allows, for instance, to de ne the desired range of deformations of a fabricated object based on its deformed con gurations corresponding to multiple scenarios, as it will be demonstrated in chapter 5. In the following sections, we will further explain both the static equilibrium constraint and the objective function h.

We note that the formulation of inverse elastic design problems might be more general, containing also other types of constraints. However, for the purpose of our analysis, it su ces to consider the problem (4.1) above.

The static equilibrium

We deal with design problems for which the main performance criterion is the deformation behavior. More precisely, we consider the equilibrium of fabricated objects for any given boundary conditions, i.e., the shape with vanishing net forces. The de ned energy potential yields the expression of the system forces as

$$\begin{aligned} f_q(q;p) &= \frac{\partial V}{\partial q} \in \mathbb{R}^n; \\ f_p(q;p) &= \frac{\partial V}{\partial p} \in \mathbb{R}^m: \end{aligned} \tag{4.2}$$

Note that f_q are the net forces acting on the DoFs corresponding to the deformed shape. The analogous magnitude at design parameters f_p is not required at this point, but it will be of use in later sections. The partial derivative of both force expressions will be also needed and we will refer to them using the following notation:

$$J(q;p) = \begin{bmatrix} J_{qq} & J_{qp} \\ J_{pq} & J_{pp} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_q}{\partial q} & \frac{\partial f_q}{\partial p} \\ \frac{\partial f_p}{\partial q} & \frac{\partial f_p}{\partial p} \end{bmatrix}: \tag{4.3}$$

The expression of static equilibrium $f_q(q_j;p) = 0$ imposes numerical constraints on the total space $\mathbb{R}$ in the form of an underdetermined system of n {generally nonlinear{ equations and o variables. The set of all possible con gurations $q(;p)$ such that these conditions hold de nes an m-manifold embedded in the total space $\mathbb{R}$, which we call equilibrium manifold, E. If the potential energy is conservative and bounded from below for a speci c parameter vector p, there is at least one equilibrium con guration that we denote:

$$\hat{q} = \{ q \mid f_q(q;p) = 0 \}; \tag{4.4}$$

located at an extremal point of the potential energy function. Recall from section 3.4 that both maxima and saddle points correspond to unstable static equilibrium states and are not desirable con gurations. Additionally, the elastic potential is generally invariant to rigid-body transformations. Therefore, from this point onward, we assume that Q has an empty null-space, i.e., translation and rotation displacements have been already constrained through boundary conditions. In any case, in section 3.4, we described a numerical solver that can be used to compute such con gurations

robustly, navigating only the deformed space Q.

The objective function

In this kind of problems, the quality of a given solution is usually measured in terms of a number of k_s shape descriptors $_i \in \mathbb{R}^{s_i}$ of an arbitrary dimensionality s_i. We call the space spanned by the combination of all possible k_s vectors $S = S_1 \times \cdots \times S_{k_s}$ the shape space.

The simplest setting would be the case where we seek design parameters such that the deformed shape of the object q exactly matches a given target configuration q^t. Then the identity function is the unique shape descriptor $(q) = q$, in which cases $= n$ and the shape space equals the deformed space $S = Q$. However, more often objectives focus on a subset of the deformation variables, $q) = T q$, with $T \in \mathbb{R}^{s \times n}$ being a selection matrix. That is the case of the works that will be described in chapters 5 and 6, for instance, where we are interested only in the positions of the centerline nodes of a rod mesh, while the deformed shape is also determined by the roll angles of the material frame. In general, any combination of shape descriptors could potentially be used in the objective function, including more sophisticated metrics like area, volume, smoothness, stability or buoyancy, among others. For each of these shape descriptors, one must provide a corresponding target value ^{t}, the distance to which we aim to minimize. This defines a generic objective function as the squared difference between shape descriptors and target values

$$h = \frac{1}{2} \sum_{j=1}^{k_s} w_j \, \| \, _j(q_i) \quad {}^t_{ij} \|^2; \tag{4.5}$$

where w_j are descriptor weights.

Basic problem formulation

For notation simplicity and without loss of generality, further analysis will consider only one pose $k_p = 1$ and one shape descriptor $k_s = 1$, which will be the identity $(q) = q$. This way, $n = s$ and the shape space is equivalent to the deformed space $S = Q$. With that in mind, we finally define the shape optimization problem as follows:

$$\begin{aligned} \min_{q;p} \; & \frac{1}{2} \| q \quad q^t \|^2 \\ \text{s.t.} \; & f_q(q; p) = 0 \\ & p_m \quad p \quad p_M : \end{aligned} \tag{4.6}$$

In the following section we will analyze the theoretical solution space of this optimization problem and characterize inverse elastic design problems accordingly.

4.1.2 The solution space

In general, equation (4.6) de nes a nonlinear least squares problem that can be solved using a standard constrained nonlinear optimization method. In this thesis, we focus on a solution scheme based on navigating the equilibrium constraint manifold. In this section, we study the solution space of the problem in order to better understand the geometry of such constraint manifold and how it is related to the objective function.

Figure 4.1: This gure represents the three possible situations that might happen depending on the dimension of design P and shape S spaces. The set of perfect solutions (red) is de ned by the intersection between the target (green) and equilibrium (blue) manifolds. Note that manifolds do not intersect in the c) case.

The objective function in the problem (4.6) trivially nds a minimum at $q = q^t$. This condition de nes a m-manifold embedded in the total space R, where the objective function vanishes, which we call target manifold, T. Note that, as the objective function does not depend on the design parameters, this target manifold is constant in the P subspace. The intersection between equilibrium E and target T manifolds de nes the set of perfect solutions with $h = 0$. In general, the maximum dimension of the intersection between two manifolds A, B embedded in a space R is a non-negative value formulated as $\mathrm{Dim}(A \cap B) = \mathrm{Dim}(A) + \mathrm{Dim}(B) - \mathrm{Dim}(R)$, which in the case of our problem results $\mathrm{Dim}(E \cap T) = \max(m - n; 0)$. Considering this fact, inverse design problems can be also characterized in terms of the dimensionality of the design P and shape S spaces. Fig. 4.1 shows a geometric representation of the three situations that might happen:

Case $\mathrm{Dim}(P) > \mathrm{Dim}(S)$. There are potentially in nite perfect solutions. One instance of this problem appears in some interactive computational tools

that feature inverse design capabilities, see e.g. [14, 26, 15, 23]. In these cases, the user sequentially applies geometric changes to the deformed con guration q, e.g., pulling from a mesh vertex handler, to de ne a new target $q^t = q + \quad q^t$. The objective function is then formulated to match the incremental change locally, resulting in a low-dimensional shape space. To select among the in nite possible solutions, regularizing terms must be added to the objective function, e.g., to favor the closest solution to the starting point.

Case $\mathrm{Dim}(P) = \mathrm{Dim}(S)$. There is at most a nite number of perfect solutions and, under some assumptions on the normal and curvature of the equilibrium manifold, it can be shown that a solution exists. This situation corresponds, for example, to design problems where the rest con guration is optimized such that the deformed con guration matches a desired shape $p = q$. In absence of external forces, this problem is trivial, i.e. $p = q = q^t$, With external forces, the problem is not trivial and it has been often studied, e.g., [140, 139, 9].

Case $\mathrm{Dim}(P) < \mathrm{Dim}(S)$. There is at most a nite number of perfect solutions, but the set is generally empty. This is the most common example in the design literature. In material characterization, for instance, only a few material parameters are considered as de ning the design space, but the shape descriptor often considers the whole deformed con guration, e.g., [148, 68, 184]. In more general fabrication-oriented design works, the complexity of solving (4.6) rapidly grows with the size of the design space, and dense parameterizations are more likely to produce noisy results. Hence, there is a general interest in using low-dimensional design spaces.

We may assume that, in most cases, there will be no intersection between the two manifolds. Note that, even when $m >> n$, the target con guration might not be reachable because we are restricted to the feasible range of design parameters $[p_m; p_M]$. Even if the target con guration is realizable, the equilibrium manifold E is generally nonlinear and we can only make a guess about the intersection set based on local linear approximations. In practice, we will look for a point in the equilibrium manifold that locally minimizes the distance function h. In section 4.2, we will explore di erent alternatives for nding this solution.

4.2 Solution methodologies

The optimization problem de ned in (4.6) can be solved using standard optimization methods for general nonlinear least-squares problems. However, we have seen that the equilibrium manifold that de nes the solution space is often nonlinear and full

of perils. Additionally, the target shape can be arbitrarily far from any static equilibrium con guration. In general, this makes the design problem very di cult and produces misleading results when computing the numerical approximations needed to solve the problem.

During the development of this thesis, we have dealt with two main aspects that largely a ect the complexity of the design problem and identi ed simple approaches that facilitate its solution: i) the selection of the design space, and ii) the de nition of the target shape. In the next two sections 4.2.1 and 4.2.2, we will describe each one of them. Then, we will introduce the two main approaches that we have considered for solving the optimization problem: constrained optimization, in section 4.2.3, and constraint manifold exploration, in section 4.2.4.

4.2.1 Design space selection

There are some aspects of the optimization problem that bene t from an adequate selection of the design parameters. In general, we aim for formulations that have two complementary attributes:

1. Naturally smooth solutions . Fabrication-oriented design applications generally bene t from producing smooth shapes and designs, not only for aesthetic reasons. The presence of high frequency noise in the shape or the material distribution of an object often reduces the accuracy of the discretizations used to approximate the continuum, produces numerical instabilities in the computation of the gradient and the Hessian, makes the fabricated object more prone to malfunction or fracture, and hinder the manufacturing process.

2. Compact parameter sets . In general, the complexity of an inverse elastic design problem rapidly grows with the size of the design space and hence there is an interest in using compact parameterizations. Although this actually involves shrinking the solution space and making it more di cult to nd equilibrium con gurations close to the target shape, it is less prone to over tting, which is also associated with non-smooth solutions.

Consequently, we are interested in selecting a compact parameter set such that the part of the solution space that is not accessible corresponds mainly to non-smooth and over tted solutions. One usual way of producing smooth solutions is to add regularization terms to the objective function, but these terms are usually not trivial to control and do not make the design space more compact. Instead of using regularization, we propose de ning the design space as a numerical interpolation from a reduced set of parameter control points $p = z( )$, with $2 R^a$ and $a << m$. The particular nature of the interpolation function z depends on the speci c design

problem. For instance, in chapter 5, we employ cubic Hermite interpolation to de ne the radii of a rod section along the centerline. Conversely, in chapter 6, we use Laplacian meshing to determine the rest con guration of the membrane of a tensile structure. In any case, it is convenient to choose interpolations where all nonlinearities can be precomputed and the nal relationship is linear $\mathbf{p} = \mathbf{Z}\mathbf{m}_{\mathbf{a}}$. Hence, for any magnitude $\mathbf{w}$,

$$\frac{\partial \mathbf{w}}{\partial \mathbf{a}} = \frac{\partial \mathbf{w}}{\partial \mathbf{p}}\mathbf{Z}; \qquad \frac{\partial^2 \mathbf{w}}{\partial \mathbf{a}^2} = \mathbf{Z}^T \frac{\partial^2 \mathbf{w}}{\partial \mathbf{p}^2}\mathbf{Z}: \tag{4.7}$$

This relationship allows us to reformulate the optimization problem, without any loss of generality, in terms of the parameter interpolation control points. To ease the notation though, we will continue using the basic formulation in (4.6).

4.2.2 Target shape de nition

As we introduced in section 4.1.2, the speci ed target shape is most likely infeasible, independently of the parameterization of the problem. In fact, it might be arbitrarily far from any possible equilibrium con guration. Under such conditions, numerical optimization methods usually produce poor results. Introducing prior knowledge about the design problem in the de nition of the target shape is generally necessary to obtain good solutions. We propose two approaches for this purpose: using physically-based targets or equilibrium manifold approximations.

Physically-based target Often, shape optimization approaches try to match a target shape that has been de ned by the user employing some geometry deformer. Alternatively, using physically-based deformers for this task highly increases the chances of producing target shapes that can be successfully matched within the considered design space. Even if the underlying mechanical model of the deformer does not accurately match the model that determines the equilibrium constraint, similar principles often apply in terms of volume conservation and deformation ratios. In chapter 5, we show how thin-shell and FEM computational models can be used to de ne a target shape that is successfully matched by a rod mesh model.

Manifold approximations Direct freeform manipulation of the target is always possible (e.g., de ning artistic shapes with an aesthetic intention), but it will most likely produce targets that are far from feasible. Instead, using an incremental target exploration ensures that the targets that are de ned remain close to feasible. Starting from an initial feasible solution, one can compute local approximations of the equilibrium manifold and successively deform the target shape such that the approximate constraints hold. Once

satis ed with the deformed shape, this can be used as the target con guration of the full nonlinear optimization problem. Although the resulting objective is probably still not feasible, the chances of the design optimization to nd a valid approximation are much higher. In chapter 6, we present a two-step optimization that is based on this principle.

4.2.3 Constrained optimization solution

Many works in the literature solve the resulting constrained optimization problems using standard optimization methods, e.g., [8, 10, 31, 28]. Our constrained nonlinear least squares problem can be generally solved using a *sequential quadratic programming* (SQP) method. This common denomination includes a large variety of optimization algorithms based on iteratively approximating the nonlinear problem (4.6) with a simpler *quadratic programming* (QP) that is easier to solve. Let us assume for now that there are no box constraints, i.e. $p_m = 1$ and $p = 1$. In such case, each SQP iteration implies:

1. A quadratic approximation of the objective function

$$\tilde{h}(\ r; r_0) = g(r_0)^T\ r + \frac{1}{2}\ r^T H(r_0)\ r; \tag{4.8}$$

   where $g = r_r h$ and $H = r_r^2 h$.

2. A linear approximation of the equilibrium constraint

$$\tilde{f}_q(\ r; r_0) = f_o(r_0) + J_q(r_0)^T\ r = 0; \tag{4.9}$$

   where $J_q = [J_{qq}; J_{qp}] = r_r f_q$.

In each SQP iteration, the solver nds a guess step r such that the linear constraint is approximately satis ed and the quadratic objective function is minimized. Depending on how the formulation handles the interaction between objective function and the constraints, di erent submethods can be considered. Next, we will brie y review the three most widely used approaches: *Lagrange multipliers*, *penalty methods* and *augmented Lagrangian*. For a more thorough dissertation on the topic, we refer the reader to the work by Nocedal and Wright [183].

Lagrange multipliers

The method of Lagrange multipliers is a standard strategy in mathematical optimization for nding the local minima of a function subject to constraints. This

method is based on considering a set of auxiliary variables $\in \mathbb{R}^m$ called Lagrange multipliers and nding the critical points of the Lagrangian function:

$$L(\ r;\ ;r_0) = \tilde{h}(\ r;r_0) + \ ^T \tilde{f}_q(\ r;r_0): \tag{4.10}$$

The critical points of the Lagrangian occur at saddle points. In the case of an equality-constrained QP problem, nding the solution is equivalent to solving a linear system whose matrix is generally inde nite of the kind

$$\begin{pmatrix} H & J_q^T \\ J_q & 0 \end{pmatrix} \begin{pmatrix} r \\ \ \end{pmatrix} = \begin{pmatrix} g \\ f_q \end{pmatrix} \tag{4.11}$$

where the matrix and the right-hand side are evaluated at t_0.

Penalty method

The penalty method replaces the constrained QP iteration with an unconstrained QP, adding an additional term to the objective function called penalty function. This function consists of a penalty parameter multiplied by a measure of the violation of constraints, commonly the squared norm. Consequently, each iteration minimizes the following quadratic function:

$$g(\ r;r_0) = \tilde{h}(\ r;r_0) + \frac{}{2}k\tilde{f}_q(\ r;r_0)k^2; \tag{4.12}$$

which simply reduces to solving a linear system of the form

$$(H + \ J_q^T J_q)\ \ r = \ g \quad J_q^T f_q; \tag{4.13}$$

with all magnitudes evaluated at r_0. In each iteration of the SQP, the penalty coef- cient is successively incremented to guarantee the enforcement of the constraints. Ideally, the solution of these series of unconstrained problems would converge to the solution of the original constrained problem. However, nding the appropriate update scheme for the penalty coe cient is not trivial, and it highly determines the success of the algorithm. Additionally, taking $\ ! 1$ in late iterations may yield ill-conditioned system matrices.

Augmented Lagrangian

Similar to the penalty method, the augmented Lagrangian method also replaces the constrained SQP by a series of unconstrained QPs, by adding a penalty term to the

objective function. However, to overcome the potential di culties of the penalty method, the augmented Lagrangian adds an extra penalty term designed to mimic Lagrangian multipliers:

$$g(\ r; r_0) = \tilde{h}(\ r; r_0) + \frac{}{2}k\tilde{f}_q(\ r; r_0)k^2 + \ ^T\tilde{f}_q(\ r; r_0); \tag{4.14}$$

which, once again, reduces to solving a linear system of the form

$$(H + \ J_q^T J_q)\ \ r = \ g \quad J_q^T f_q + \ ^T J_q: \tag{4.15}$$

Here is an estimation of the Lagrangian multipliers, which is updated after each iteration as $f_q(r\ )$, for a given candidate QP solution $r\ = r_0 + \ r\ .$ The resulting estimate improves at every step, progressively reducing the need for the penalty term. Consequently, it is not necessary to take $!\ 1$, and thus ill-conditioned problems are avoided.

Practical considerations

There are some important aspects for the practical implementation of these methods:

1. Box constraints . Given the formulations presented above, it is easy to add box constraints on the design parameters $p_m < p < p_M$ to each quadratic subproblem. One might simply project the resulting step within the bounds, but that does not necessarily guarantee that the solution is optimal and hence compromises the quadratic convergence of the SQP method. The literature on box-constrained QP problems is extensive and lies out of the scope of this text, but we suggest using the *gradient projection* method, see e.g., [183].

2. Linear system solving . Solving the linear systems associated to each of these formulations is not free of perils. A positive de nite Hessian H is preferred to guarantee the convergence of the SQP algorithm. The regularization methods suggested in section 3.4 are also applicable in this context. However, it is also possible to use approximations to the full Hessian such that positive de niteness is satis ed. We will later describe these approaches in section 4.3.3 in the context of constraint exploration solutions.

3. Step length selection . Independently of the particular method used, each iteration results in a new guess step r and hence a candidate solution $r_k = r_{k\ 1} + \ r$, which should be closer to a local minimum of the objective. Here, represents an adapted step length. This is necessary because the QP is just an approximation of the more complex nonlinear problem, and its resulting step does not always lead to the improvement of the nonlinear objective

function h. Independently of the particular step-length selection method, e.g., line-search, trust-region, etc., it is necessary to de ne a merit function $(h; f)$ that measures the quality of a given solution. For an unconstrained problem, there is a straightforward solution, i.e., $= h$. However, in the constrained case there is no unique way of formulating this metric. Recall that most design problems have no perfect solution; therefore, objective minimization h and constraint ful llment $f_q = 0$ are opposing goals. A common choice is to use the penalty method function, $= h + \|f\|^2$, where is a weighting factor that adaptively changes depending on the convergence properties of the problem. should be small at the beginning of the optimization, and is progressively increased to ensure that the nal solution is in static equilibrium.

Box constraints and the regularization of the linear system are standard issues in the resolution of QP subproblems. Therefore, from this point onward, we will ignore these issues and refer the reader to this section whenever a box-constrained QP problem is formulated.

Discussion

A few practical observation about the constrained SQP method:

1. At each iteration, the static equilibrium constraint may not hold for the candidate solution. This allows a broad exploration of the total space, which might facilitate avoiding local minima.

2. Appropriately selecting a valid merit function, as well as adaptively nding a compromise between the objective function h and constraint ful llment $f_q = 0$ is highly non-trivial.

3. When using Lagrange multipliers, the linear system (4.11) solved at each iteration might be relatively costly to solve. The system size is t, with $t = 2n + m$. Additionally, the matrix A is usually inde nite; therefore, fast linear solvers for symmetric positive semi-de nite matrices cannot be used.

4. When using penalty and augmented Lagrangian methods, tuning the penalty coe cient appropriately can be di cult and, as we have seen, leads to ill-conditioned system matrices.

5. When considering at the same time several N_p deformed poses with their own boundary conditions, these methods do not scale well. All the poses share the same parameter set and the formulation cannot be easily decoupled. For N_p poses, the size of the problems results $t_p 2n + m$.

Except for the first observation, the rest constitute serious disadvantages on the applicability of the constrained SQP method in our particular scenario. In the following section, we introduce how we alternatively approach the problem (4.6), using sensitivity analysis to iteratively explore the equilibrium constraint looking for better solutions.

4.2.4 Constraint exploration solution

Some works in the literature follow alternative approaches for the solution of the constrained problem (4.6), based on the exploration of the constraint manifold, e.g., [14, 22, 23, 148, 68, 165]. Given some initial guess r_0, such that $f_q(r_0) = 0$, these methods iteratively compute a tentative step Δr, such that the candidate solution $r_1 = r_0 + \Delta r$ reduces the objective function h and approximately satisfies the constraint $f_q \approx 0$. This same idea can be implemented using several numerical methods. Yang et al. [26], for instance, generate steps on the manifold considering both first and second order approximants {i.e., tangent space and quadratically parameterized osculant surface} that can be directly computed in the parameter space. Alternatively, we approximate the constraint manifold using a sensitivity analysis matrix computed by virtue of the implicit function theorem. In the next section, we will explain in detail how to solve the optimization problem (4.6) using sensitivity-based constraint exploration.

4.3 Sensitivity-based optimization

In this section, we explain in detail an alternative solution to the nonlinear constrained optimization, based on the exploration of the constraint manifold. Roughly, this method finds an explicit expression of the equilibrium manifold through local approximation, by virtue of sensitivity analysis. In the next section 4.3.1, we explain how this is done using the implicit function theorem, and we study the resulting implicit equilibrium function. Then, in section 4.3.2, we reformulate the problem (4.6) using this function, and we outline a SQP method for solving it. Finally, we will focus on two specific aspects of the algorithm that strongly determine its performance: the quadratic subproblem 4.3.3 and the projection to the equilibrium manifold 4.3.4.

4.3.1 Implicit equilibrium

We seek a mathematical expression of the the static equilibrium constraint manifold. The implicit function theorem states that if an equation $f(q_1, \dots, q_n, p_1, \dots, p_m) = f(q, p) = 0$ satisfies continuity and differentiability conditions, then there exists

an implicit function $\hat{q} : \mathbb{R}^m \to \mathbb{R}^n$ such that $\hat{q}(p)$ locally approximates the locus defined by $f(q;p) = 0$. That is, $\hat{q}$ is an implicit function parameterized by p of the static equilibrium configuration as defined in (4.4). Although there might not exist a closed analytic expression of $\hat{q}$, we can easily compute a linear approximation as follows. Assuming that the constraint f_q is a continuously differentiable function and there exists a point $(q_0; p_0)$ for which its value satisfies $f(q_0; p_0) = 0$, we can apply the Taylor expansion series on the function f_q, assuming that the implicit function $\hat{q}(p)$ does exist:

$$f_q(\hat{q}(p); p) \approx f_q \simeq f_q(p_0) + \frac{df_q}{dp} \Delta p = 0. \tag{4.16}$$

We know $f_q(p_0) = 0$; therefore, applying the chain rule we have:

$$\frac{df_q}{dp} \Delta p = 0 \Rightarrow J_{qq} \frac{\partial \hat{q}}{\partial p} \Delta p + J_{qp} \Delta p = 0. \tag{4.17}$$

Finally, factoring Δp out and reordering terms:

$$J_{qq} \frac{\partial \hat{q}}{\partial p} + J_{qp} = 0 \Rightarrow J_{qq} \frac{\partial \hat{q}}{\partial p} = -J_{qp}. \tag{4.18}$$

The matrix resulting from solving the linear system,

$$S = \frac{\partial \hat{q}}{\partial p}; \tag{4.19}$$

is called sensitivity matrix, and the corresponding linear approximation of the implicit function, $\hat{q} \simeq q_0 + S \Delta p$, defines a linear manifold that is locally tangent to the equilibrium manifold. This can be easily proved by substituting $\Delta q = S \Delta p$ in the equation that defines the null-space of the equilibrium constrain:

$$\begin{bmatrix} J_{qq} & J_{qp} \end{bmatrix} \begin{bmatrix} \Delta q \\ \Delta p \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}; \tag{4.20}$$

which holds true independently of the value of Δp.

Sensitivity matrix

The sensitivity matrix spans the plane tangent to the equilibrium constraint at the evaluation point r_0. We will refer to the resulting linear manifold $\bar{E}$ as the

Figure 4.2: This picture shows linear approximations to the equilibrium manifold, for the di erent situations that were described in section 4.1.2. The top row depicts some good approximation examples. However, depending on the nonlinearities of the equilibrium manifold, the linear manifold might produce poor estimates (bottom left) or degenerate cases with in nite (bottom center) or singular (bottom right) functions.

linear equilibrium manifold. Fig. 4.2 shows the linear equilibrium approximations corresponding to the three design situations introduced in section 4.1.2. In practical terms, the sensitivity matrix allows us to compute, for a given parameter change p, the displacement in the deformed con guration q such that the linearized static equilibrium constraint $\tilde{f}_q = 0$ holds. Some other useful conclusions can be drawn from the numerical properties of the matrix:

Linear equilibrium : In general, the sensitivity matrix S has dimension $m \quad m$ and $[S^T; I]$, for $I_{m \quad m}$, is a basis whose rows span the linear equilibrium m-manifold. The intersection of this linear manifold and the target manifold, if it exists, approximates the set of perfect solutions to the problem introduced in section 4.1.2 (Fig. 4.2, top row).

In nite sensitivity : If J_{qq} is rank-de cient, the linear system in (4.18) cannot be solved. This corresponds to situations for which the linear equilibrium approximation runs parallel to the shape space Q {see Fig. 4.2 picture e. In practical terms, this means that the deformed con guration is in nitely sensitive w.r.t. some of the design parameters, and the matrix is not valid.

Singular sensitivity : If J_{qp} is rank-de cient, the linear system in (4.18) can be solved, but the resulting sensitivity matrix is also rank-de cient. This corresponds to points where there exists at least a direction in the design space P for which the deformed con guration does not change. Here, the sensitivity matrix is still valid but it may produce di cult situations when solving the optimization. In the worst case scenario, the linear equilibrium approximation runs parallel to the design space P {see Fig. 4.2, picture f . In such case, traversing the linear equilibrium does not change the objective function.

Ill-conditioning : Sometimes, J_{qq} and J_{qp} matrices are ill-conditioned or numerically close to being rank-de cient. This might cause that S is not well-behaved in two ways. On one hand, the condition number {i.e., the di erence between minimum and maximum singular values{ may be high, what means that the deformed con guration is much more sensitive to some design parameters than others. On the other hand, the linear equilibrium manifold may be almost parallel to the target manifold, what means its navigation have very little e ect on reducing the objective function.

In general, the sensitivity matrix provides us with a formal framework to rewrite the optimization problem (4.6) in terms of the implicit function $\hat{q}$, as we will see in the next section.

4.3.2 Sensitivity-based SQP

The implicit equilibrium $\hat{q}(p)$ de nes the equilibrium con guration corresponding to a given parameter vector p. Using this function, we can remove the static equilibrium constraint in (4.6), and reformulate the objective function h to evaluate shape descriptors only on equilibrium con gurations:

$$\hat{h} = \frac{1}{2} k \hat{q}(p) \quad q^t k^2 : \tag{4.21}$$

This way, the nal equilibrium shape optimization problem is:

$$\begin{aligned} &\min_p \hat{h} \\ &\text{s.t. } p_m \quad p \quad p_M : \end{aligned} \tag{4.22}$$

Note that the force equilibrium constraint is no longer needed as we implicitly consider only deformed con gurations in static equilibrium $\hat{q}$. Generally, there is no explicit analytic expression for this magnitude, which can be highly nonlinear, and one can only rely on locally tangent approximations $\hat{q}$ {e.g., the one de ned in the previous section based on the sensitivity matrix. This problem nds minima at:

$$\hat{g} = \frac{\partial \hat{h}}{\partial p} = 0 \;\Rightarrow\; (q - q^t)^T \frac{\partial \hat{q}}{\partial p} = 0. \tag{4.23}$$

Algorithm 1: Sensitivity-based SQP

Data: Target shape value: q^t
Data: Initial guess: $\{(q_0, p_0) \mid f_q(q_0, p_0) = 0\}$
Data: Gradient and constraints tolerance: t_g, t_f
Result: Optimal solution:
$\{(q, p) \mid \|f_q(q, p)\| \le t_f \text{ and } \|\hat{g}(q, p, q^t)\| \le t_g\}$

1. Assign initial solution $(q_k, p_k) \leftarrow (q_0, p_0)$;

while $\|\hat{g}(q_k, p_k, q^t)\| > t_g$ do

  2. Quadratically approximate objective $\tilde{h}(p; r_k) \approx \hat{h}(\hat{q})$;
  3. Solve QP subproblem $p_{k+1} \leftarrow$ SOLVE$(\tilde{h}, p_m, p_M)$;
  4. Compute the implicit deformation $q_{k+1} = \hat{q}(p_{k+1})$;
  5. Compute candidate steps: $\hat{q} = q_{k+1} - q_k$ and $\hat{p} = p_{k+1} - p_k$;

  do

    6. Compute candidate solutions: $q_{k+1} \leftarrow q_k + \hat{q}$, $p_{k+1} \leftarrow p_k + \hat{p}$;
    7. Recompute equilibrium $(q_{k+1}, p_{k+1}) \leftarrow$ PROJECT(q_{k+1}, p_{k+1}, t_f);
    8. Reduce candidate steps length: $\hat{q} \leftarrow \hat{q}$ and $\hat{p} \leftarrow \hat{p}$;

  while $(h(q_{k+1}, p_{k+1})) < h(q_k, p_k))$;

end

The resulting optimization is a nonlinear least squares problem. The algorithm 1 outlines the basic steps for the resolution of the optimization problem (4.22) using a standard unconstrained SQP method with a simple line-search procedure for step length selection. There are three key parts of the algorithm that will define its performance:

1. Quadratic subproblem solution (Steps 2–3). In each iteration of the algorithm, the nonlinear objective function (4.21) is approximated with a quadratic function, which turns the minimization into a standard box-constrained QP problem. The subroutine SOLVE encapsulates the computation of a solution for such problem. Depending on the type of approximation to the objective function that is used, the overall SQP convergence properties vary. As we will see, this is also related with how the implicit equilibrium function is estimated $\hat{q}(p; r_k) \approx \hat{q}$. In section 4.3.3, we go deeper into the relation between QP methods and equilibrium manifold estimation.

2. Static equilibrium computation (Step 7). The implicit equilibrium function estimation $\hat{q}(p; r_k)$ is only locally tangent to the static equilibrium constraint. Solving the quadratic subproblem will produce a candidate step

$(\tilde{q}; \tilde{p})$ that lies in the manifold de ned by the implicit equilibrium approximation but for which the static equilibrium constraint does not necessarily hold. The longer the step, the larger the deviation from the equilibrium manifold $k f_q(q_k + \tilde{q}; p_k + \tilde{p}) k > 0$. The objective function is de ned only in terms of a deformed con guration in static equilibrium. For that reason, after each quadratic step, the solution must be projected back to the equilibrium manifold using the subroutine PROJECT. We have already explained how to compute the static equilibrium deformation for a given constant parameter vector in section 3.4. In section 4.3.4, we introduce an alternative solution scheme that is advantageous in some situations.

3. Step length selection (Steps 6{8) . Minimizing the quadratic approximation of the objective function $\tilde{h}(p; r_k)$ reduces the nonlinear objective h for a su ciently small step. In this algorithm outline, we consider the most basic line-search procedure: the length of the step is iteratively bisected until the objective function h is reduced. Adequately selecting the step length is critical for our problem, as each time a new candidate solution is evaluated, it is necessary to compute the corresponding static equilibrium, what might be costly. The study of alternative step selection schemes, e.g., Armijo line-search or trust-region methods, goes beyond the scope of this thesis. For more information, we refer the reader to [183].

4.3.3 Quadratic subproblem

The standard quadratic approximation of the objective function (4.21) is:

$$\hat{h}' \quad \tilde{h} = \frac{1}{2} \quad p^T H \quad p + \quad t^T g + c; \tag{4.24}$$

where the vector $g = r_p \hat{h}$ is the gradient of the objective function, the matrix $H = r^2_{pp} \hat{h}$ is the Hessian, $p = p_{k+1} \quad p_k$, $t = q^t \quad q_k$, and c is a scalar constant that does not a ect the result of the minimization. Computing the critical point w.r.t. p of this quadratic function reduces to solving the linear system $H \quad p = \quad g$. As long as the system matrix is positive de nite, the resulting step lies on a descent direction of the objective function $\hat{h}$, and the step p improves the nonlinear solution for a su ciently small value of .

Quadratic subproblems are completely characterized by the approximation to the Hessian and gradient. Depending on the particular approximation of the Hessian, the overall SQP convergence properties vary. As we will see, this is also related with how the implicit equilibrium function $\tilde{q}(p; r_k)' \quad \hat{q}$ is estimated. In the following section, we review the most common approximations that have been used in the

literature, and we illustrate their geometrical interpretation.

Linear approximation (Gauss-Newton step)

The simplest estimation of the implicit equilibrium function $\hat{q}$ is the linear approximation, which uses the sensitivity matrix previously introduced in section 4.3.1:

$$\hat{q} \simeq \tilde{q}(p; r_k) = q_k + S \quad p: \tag{4.25}$$

Using this estimation, the quadratic objective function becomes:

$$\hat{h} \simeq \tilde{h} = \frac{1}{2} \| q_0 + S \quad p \quad q^t \|^2; \tag{4.26}$$

which in the standard form de ned in equation (4.24) is

$$\hat{h} \simeq \tilde{h} = \frac{1}{2} \quad p^T S^T S \quad p + \quad t^T S \quad p + \quad t^T \quad t; \tag{4.27}$$

and the corresponding gradient and Hessian are

$$g = S^T \quad t; \qquad H = S^T S: \tag{4.28}$$

This formulation is equivalent to the so-called Gauss-Newton step and the resulting Hessian H is symmetric and semi-positive de nite by de nition. The geometric interpretation of this solution follows straightforwardly from gure Fig. 4.2, where linear approximations to the equilibrium constraint manifold are depicted. The target T and equilibrium E manifolds become linear functions. The problem reduces to the computation of the shortest distance between two linear manifolds, and it can be analyzed in terms of the dimensionality of these manifolds:

Dim $(P) >$ Dim (S). This is analogous to the intersection of two planes in a ne 3D space, which has in nite solutions as shown in Fig. 4.2, picture a. In this situation, S has dimension $n \quad m$ with $m >> n$; therefore, $H = S^T S$ is generally singular. The step can be computed by solving

$$\min_p \frac{1}{2} \quad p^T \quad p \quad \text{s.t} \quad H \quad p = \quad g;$$

what selects the solution with the smallest norm.

Dim $(P) =$ Dim (S). This is analogous to the intersection of two lines in a ne 2D space, which has a single solution as shown in Fig. 4.2, picture b. In this

situation, S has dimension $n \quad n$; therefore, $H = S^T S$ is generally invertible.

$\mathrm{Dim}(P) < \mathrm{Dim}(S)$. This is analogous to the intersection of two lines in a ne 3D space, which has generally no solution as shown in Fig. 4.2, picture c. In this situation, S has dimension $n \quad m$ with $m < n$; therefore $H = S^T S$ is generally invertible. The problem is equivalent to the one of nding the shortest distance between two lines.

Figure 4.3: This picture shows two iterations of Gauss-Newton steps for a 2D example problem. Left) in the rst iteration, the method provides a good approximation. Right) in the second iteration, the method clearly overshoots the solution, and the candidate step requires a bisection to guarantee convergence.

In section 4.3.1, we introduced potential problems connected with the properties of the sensitivity matrix S. An interesting geometric observation is that if S is almost singular, ill-conditioned, or has relatively small norm, the linear equilibrium and target manifolds might be close to parallel in some direction of the design space P. If the estimation point r_k is still far from the optimum, then the calculation of the closest point between the linear equilibrium and target manifolds would result in large steps, as depicted in Fig. 4.3 (right). These steps often overshoot the disc for which the quadratic function $\tilde{n}$ is a good estimator of the nonlinear function $\hat{n}$ and, consequently, they yield poor performance due to the costly line-search. In such cases, it might be advisable to use SQP schemes with a more conservative step length selection procedure, e.g., the Levenberb-Marquardt trust-region algorithm.

Quadratic approximation (Newton-Raphson step)

To reduce the problem of overshooting the step, a higher-order expression could be used to estimate the implicit equilibrium $\hat{q}$. Its quadratic approximation yields:

$$\hat{q} \approx \hat{q}(p; r_k) = q_k + S \quad p + \frac{1}{2} \quad p^T C \quad p; \tag{4.29}$$

where C is the third-order tensor of second derivatives $C = r^2_p q$, which indicates the curvature of the implicit function $\hat{q}$. Using this estimation, the corresponding objective function is:

$$\hat{h} \approx \tilde{h} = \frac{1}{2} k q_k + S \quad p + \frac{1}{2} \quad p^T C \quad p \quad q^t k^2; \tag{4.30}$$

which in the standard form de ned in equation (4.24) is

$$\hat{h} \approx \tilde{h} = \frac{1}{2} \quad p^T (S^T S + \quad t^T C) \quad p + \quad t^T S \quad p + \quad t^T \quad t: \tag{4.31}$$

Here, third and fourth order terms have been removed to consider only a quadratic approximation. The corresponding gradient and Hessian of the QP problem are:

$$g = S^T \quad t; \qquad H = S^T S + \quad t^T C: \tag{4.32}$$

This formulation is equivalent to the so-called Newton-Raphson step, resulting from the direct application of the quadratic Taylor expansion to the nonlinear objective function $\hat{h}$. Fig. 4.4 shows a depiction of Newton step in the same scenario presented above. Note that although the expression (4.30) uses a quadratic approximation of the implicit equilibrium function $\hat{q}$, the QP step is computed minimizing a distance where the third and fourth order terms have been removed. Consequently, the QP problem does not exactly minimize the distance from the target manifold to a quadratic expression of the implicit equilibrium, but an approximation.

Also note that the Gauss-Newton problem formulated in (4.27) di ers from the Newton-Raphson problem formulated in (4.31) only in the expression for the Hessian. Gauss-Newton misses the term with second derivatives of the implicit equilibrium manifold C. However, as with Gauss-Newton, there are also reasons for not using the standard Hessian in Newton's method:

> Analytically computing the second derivatives of the implicit equilibrium manifold C is generally very costly. It is possible to compute a numerical estimation of this term using nite di erences, but that requires solving m static equilibria plus m sensitivity computations, one per di erential change in p.
>
> The term C in (4.31) is scaled by t, which is the current di erence vector to the solution q^t. Consequently, the in uence of this term on the Hessian is progressively smaller as the solution converges to the optimum. In practical

Figure 4.4: This picture shows the example depicted in Fig. 4.3 using Newton-Raphson steps. In both cases, the higher-order estimation of the equilibrium manifold leads to better candidate solutions and improved convergence.

terms, for a problem where the equilibrium and target manifolds are not far apart, the term depending on the first-order approximation S will suffice for a good convergence.

There is no guarantee that the complete Hessian $H = S^T S + \lambda t^T C$ is definite positive. Indefinite matrices can produce non-descendent steps when solving the linear system associated to the QP problem. In general, indefinite Hessians arise when there are saddle points in the objective function $\hat{m}$, which is relatively common considering that the equilibrium manifold is often highly nonlinear.

To avoid some of the limitations mentioned above, many classic works on optimization have proposed alternative approximations to the Hessian, mainly based on rank-one iterative updates. Next, we will discuss two of them.

Quasi-Newton steps

Quasi-Newton methods are a class of optimization algorithms characterized by approximating the Hessian with a matrix that fulfills certain properties. Quasi-Newton methods are often used when the full Hessian is unavailable, too expensive, or ill-behaved. In general, the approximation is iteratively computed as $B_k = B_{k-1} + \Delta B$ such that the secant equation is satisfied. Considering the gradient g and the parameters p in the current and previous steps, the secant equation is stated as $y_k = B s_k$, for $y_k = g(p_k) - g(p_{k-1})$, and $s_k = p_k - p_{k-1}$. If the dimensionality of the problem $m > 1$, the secant equation is underdetermined. The various quasi-Newton methods are formulated on the basis of the choice of constraints imposed to the resolution

of the secant equation, which in turn lead to di erent one-rank updates. We follow the well-known subclass called BFGS (Broyden-Fletcher-Goldfarb-Shanno), which guarantees that the resulting matrix is symmetric and positive de nite. Given that $y_k^T s_k > t_b$ holds, for some tolerance t_b, two alternative updates can be used:

1. Basic BFGS version . Starting from $B_0 = I$, the following formula computes an update B to the approximation of the full Hessian, $B_k = B_{k-1} + B \approx H_k$, such that it remains symmetric and positive de nite:

$$B = \frac{y_k y_k^T}{y_k^T s_k} - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k}. \tag{4.33}$$

2. Gauss-Newton version . Starting from $B_0 = S_0^T S_0$ and $D_0 = 0$, the following formula computes an update D only to the approximation of the third order term $-t^T C$ introduced in (4.30), with

$$D = \frac{z_k y_k^T + y_k z_k^T}{y_k^T s_k} - \frac{z_k^T s_k}{(y_k^T s_k)^2} y_k y_k^T; \tag{4.34}$$

where $z_k = (y^{\#} - D_k s_k)$ and $y_k^{\#} = S_k^T t_k - S_{k-1}^T t_k$. Then, the approximation to the full Hessian is iteratively updated as $B_k = S_k^T S_k + D_k \approx H_k$, with $D_k = D_{k-1} + D$.

Note that the latter option is specially attractive as it takes advantage of the already computed sensitivity matrix S and only re nes the third order term iteratively, while keeping the Hessian well-behaved. In both cases, as an additional feature, BFGS o ers the possibility of estimating either the Hessian H, or directly its inverse H^{-1} using the Sherman-Morrison formula. For further details on the numerical properties of quasi-Newton methods, we refer the reader to Nocedal and Wright [183]. Note that, from a geometric perspective, the quasi-Newton step originates from the distance to a quadratic approximation of the implicit equilibrium in (4.30), in which the third and fourth order terms have been neglected. Consequently, the same limitation mentioned above for the Newton-Raphson step applies.

Additional note about performance

The computation of the sensitivity matrix S requires solving m linear systems, what might exceed the time-budget of an interactive design tool, like the one we present in chapter 6. For the purpose of analysis, throughout this chapter, we have formulated the sensitivity matrix explicitly. However, solving the QP subproblem requires only evaluating the gradient g and Hessian H of the objective function, and hence we could apply the adjoint method instead, and avoid computing S explicitly.

4.3.4 Equilibrium projection

Steps resulting from the solution of the QP subproblem satisfy the equilibrium constraint only approximately. That is, new candidate solutions satisfy $\tilde{f}_q(q_{k+1}; p_{k+1}) = 0$, only for some approximation $\tilde{f}_q \approx f_q$, i.e., in the cases that we have considered, linear or quadratic. It is necessary to project each candidate solution back to the equilibrium manifold for two reasons:

1. The evaluation of the objective function is defined only for deformed configurations in static equilibrium. Therefore, to assess whether or not there has been an improvement in the nonlinear minimization for a given QP candidate solution, it is necessary to recompute the static equilibrium.

2. The implicit function theorem used for the estimation of Ω in the next step assumes the hypothesis $f_q = 0$. Therefore, it is mandatory to perform manifold projection at the beginning of each step.

In this part of the thesis we analyze possible implementations of such projection subroutine. First, let us formalize some concepts that will be revisited in the following sections. Fig. 4.5 shows a simplified representation of the total space, where all variables corresponding to the deformed Q and design P subspaces are respectively represented in the q and p axes. We have thoroughly studied the constraint $f_q = 0$, which represents the m-manifold of static equilibrium configurations. However, as introduced at the beginning of this chapter, $f_p = 0$ defines another n-manifold at which the partial derivatives of the elastic potential w.r.t. design parameters vanish. Recall from section 3.4 that the minimization of potential energy leads to equilibrium constraint satisfaction. In this context, we might formulate the projection subroutine in three possible ways:

1. Potential energy minimization in R (Fig. 4.5, left): $\min_r V(r)$.

2. Potential energy minimization in Q (Fig. 4.5, center): $\min_q V(r)$.

3. Root-finding in R (Fig. 4.5, right): find (r) such that $f_q(r) = 0$

In the following section, we will consider each of this options and analyze their advantages in terms of flexibility, robustness and performance, for the particular case of inverse elastic design problems solved through constraint exploration.

Potential energy minimization in R

Even though this approach is not practical, as we will discuss next, it deserves some attention and analysis. Intersections between the two constraints $f_q = 0$ and $f_p = 0$

Figure 4.5: This picture shows a compact representation of the projection problem in the total space R, where all the DoFs corresponding to deformed Q and design P spaces have been condensed in the q and p axes. Three possible projection schemes are shown, for two di erent starting points r_1 and r_2: a) potential energy minimization in total space R nds the critical point of the potential energy function V w.r.t. q and p; b) potential energy minimization in deformed space Q nds the critical point of V w.r.t. q; and c) root- nding in total space R solves the nonlinear system of equations $f_q(q;p) = 0$.

de ne critical points of the elastic potential function w.r.t. all the variables of the total space R. It would be possible to implement the projection by computing the minimization $\min_{p;q} V(r)$ s.t. $p_m \quad p \quad p_M$. However, this idea might not be advisable for two reasons:

First, we are only interested on satisfying static equilibrium constraints on deformation variables, other considerations might deviate projection from the original goal and produce worse performance results.

Second, the behavior of the potential energy function in the total space R is usually full of perils that lead to numerically di cult situations such as inde nite and stationary points. For instance, deformed and material con g- urations often produce opposite e ects on elastic strains, what can easily lead to R having a non-empty null-space and hence an in nite number of solutions. Similarly, material parameters often lead to trivial solutions if they are not bounded {e.g., lower elastic sti ness produce always smaller potentials. Something similar happens with the gravitational potential energy, which vanishes if the undeformed con guration of the object collapses.

In practical terms, there is no straightforward application, in the inverse elastic design scenario, for nding the design space con guration with minimum internal energy. In addition, there might be too di cult or even impossible for the minimization to converge using standard procedures, making this option not worthy.

Potential energy minimization in $\mathcal{Q}$

Intersections between the equilibrium manifold $\mathcal{E}$ and a fixed-parameter subspace $p = p_c$, define critical points of the elastic potential function w.r.t. the variables of the deformed space $\mathcal{Q}$. The problem is reduced to solving the minimization problem $\min_q V(r)$. Note that, even though the potential energy is affected by both deformed configuration q and design parameters p, this optimization minimizes only the elastic potential w.r.t the deformed configuration. If we restrict to the deformed space $\mathcal{Q}$, the potential energy is often relatively well-behaved. A robust SQP algorithm for the solution of this optimization problem has already been defined in section 3.4, together with various techniques to improve the convergence and robustness of the method against indefinite and stationary points.

Fig. 4.5 (center) shows the geometric interpretation of this particular projection subroutine. In general, the method produces good projected solutions (point r_1). However, it is possible to find scenarios for which the equilibrium point is relatively far from the initial point in deformed space $\mathcal{Q}$, compared to the corresponding distance in the total space $\mathcal{R}$ (point r_2). This might have two main disadvantages:

1. Convergence to the equilibrium point may require more iterations of the optimization. As it has been mentioned before, a slow projection subroutine has a big impact on the performance of constraint exploration optimization methods, as it is sometimes necessary to evaluate the objective function {i.e., recompute projection{ several times during step length selection.

2. In the specific case of sensitivity-based optimization methods, the candidate solution to project will be potentially close to a target configuration q^t. Long projection distances would lead to deformed configurations far from the target, and hence slow down the overall convergence of the design optimization problem.

Motivated by these facts, in the next section we discuss an additional formulation, based on using a root-finding algorithm to solve a nonlinear system of equations, that might outperform the usual solution in some scenarios.

Root-finding in total space $\mathcal{R}$

Fig. 4.5 (right) shows that, given a candidate solution in total space r_k, there is actually an infinite number of possible projection points if we consider traversing the total space $\mathcal{R}$ instead of just the deformed space $\mathcal{Q}$. Many of these options produce projected solutions that i) are closer to the initial candidate solution and/or ii) are closer to the target deformed configuration. We seek an algorithm to traverse the total space $\mathcal{R}$ and robustly converge to any of the solutions defined by the

constraint $f_q = 0$, subject to some performance criterion we might need depending on the particular application. In this section, we propose an approach based on the under-determined Newton algorithm. We can de ne the problem at hand analogously to the formulation of the static equilibrium introduced in section 3.4 based on Newtonian mechanics. We consider the following root nding problem:

$$\text{nd}(r) \quad \text{s.t.} \quad p_m \quad p \quad p_M \quad \text{such that} \quad f_q(r) = 0; \tag{4.35}$$

where $f_q : R^o ! R^n$ is a continuously di erentiable function with $o > n$. In this situation, Newton's method begins assuming an initial guess r_0 and then generates a sequence of iterates via

$$r_{k+1} = r_k \quad J_q^{\ 1} f_q(r_k): \tag{4.36}$$

In practice, this involves solving the linear system:

$$J_q \quad r = \begin{bmatrix} J_{qq} & J_{qp} \end{bmatrix} \begin{bmatrix} q \\ p \end{bmatrix} = \quad f_q(r_k): \tag{4.37}$$

The main di erence with the root- nding algorithm de ned in (3.32) is that, in this case, the linear system (4.37) is under-determined, i.e., it may have an in nite number of solutions. In order to develop a well-de ned algorithm, additional constraints must be imposed so that a unique step r_k can be de ned. The most basic solution is the so-called *normal ow method*, which chooses the solution with minimum Euclidean norm. This can be easily computed by solving the QP problem:

$$\begin{aligned} \min_r \ & \frac{1}{2} \quad r^T \quad r \\ \text{s.t.} \ & J_q \quad r = \quad f_q(r_k) \\ & p_m \quad p \quad p_M : \end{aligned} \tag{4.38}$$

In this explanation, we ignore box constraints and refer the reader to the practical considerations in section 4.2.3. The step resulting from solving (4.38) is $s = \quad J_q^+ f_q$, where $J_q^+ = J_q^T (J_q J_q^T)^{\ 1}$ is the well-known Moore-Penrose pseudo-inverse. This step is generally referred to as the Moore-Penrose step. This solution of the linear system is a natural choice for a Newton step because it is the shortest step from the current iterate to a root of the linear problem and, therefore, the linear model is likely to be a better representation of the nonlinear function at that step than at other solutions. Under certain assumptions {i.e., f_q and J_q continuous and di erentiable and J_q full rank{ this algorithm has proven local convergence. Fig. 4.6 (left) shows a depiction of the subsequent steps corresponding to (4.38).

Figure 4.6: This gure shows two convergence examples for the root- nding method in total space R. The left picture corresponds to an extension of the normal ow method: the steps are necessarily aligned with the normal to the constraint de ned by the linear system in equation (4.37). The right picture corresponds to a projection scheme for which the shortest step objective $r^T \quad r$ used in equation (4.38) has been modi ed to account also for the design target.

While local convergence is guaranteed, our problems are rarely quadratic. Consequently, globalized inexact versions of the method are required. Many global extensions to the normal ow method have been proposed in the literature, implementing some kind of step length selection.

Line-search . These are based on the use of a merit function, which is usually chosen to be $= kf_q k$. A line-search stage is added after the resolution of the QP subproblem in (4.38) to select such that $kf_q(r + \quad r)k < kf_q(r)k$.

Trust-region . These are based on solving the QP subproblem subject to the constraint $r < \quad$, for some trust-region area . Characteristic algorithms of this class are *Dogleg methods*, which build a piecewise linear curve approximating the solution using a combination of the Moore-Penrose step and the point minimizing the problem (4.38) in the steepest descent direction.

For additional information on basic extensions to this algorithm, we refer the reader to [185]. Alternatively, this problem has also been studied as the minimization of a nonlinear least squares error function $= 1=2kf_q k^2$, using both line-search and trust-region Gauss-Newton-like formulations, for instance, the Levenberg-Marquardt solutions proposed in [186, 187].

Nevertheless, the normal ow method is not free of limitations:

Performance . Most of the global extensions use the norm of the forces in the deformed con guration $kf_q k$ as the measure of progress. However, we have

seen in section 3.4 that critical points of this function could be saddle points of the potential energy V, and hence correspond to unstable equilibria. Furthermore, as introduced before, the equilibrium manifold is highly nonlinear and traversing the total space is usually full of perils. Consequently, more robust methods are required to guarantee the fast convergence of these algorithms.

Flexibility . Taking always steps in the direction of the constraint normal (i.e., the Moore-Penrose direction) does not o er much improvement in terms of exibility with respect to the potential energy minimization in deformed space Q. Ideally, we would like to control as much as possible how the algorithms converge to the nonlinear manifold, for instance, considering the distance to the design target con guration t within the QP subproblem formulation in (4.38) resulting:

$$\begin{aligned} \min_{r} \; & \frac{1}{2} \quad r^T \quad r + \quad t^T \quad t \\ \text{s.t. } & J_q \quad r = \quad f_q(r_k) \\ & p_m \quad p \quad p_M ; \end{aligned} \tag{4.39}$$

for some weight controlling the deviation from the normal ow method. An algorithm of this kind might e ectively project an approximate solution back to the equilibrium manifold, while simultaneously improving the convergence of the overall inverse design problem, Fig. 4.6 (right). However, to our knowledge, there are no theoretical studies that analyze how steps moving away from the Moore-Penrose direction might a ect convergence, and further work is needed in this area.

4.3.5 Discussion

In this section, we have analyzed in detail the sensitivity-based SQP algorithm for the solution of inverse elastic design problems. We have discussed two aspects of the algorithm that critically de ne its performance: i) the formulation of the quadratic subproblem and ii) the projection to the constraint manifold. The optimal choice highly depends on the particular problem. During the development of this thesis, we have found that the following approaches perform signi cantly better than others, for the kind of design problems that we have dealt with:

Quadratic subproblem . Gauss-Newton BFGS method generally outperforms other options. The BFGS update is only valid when the condition $y_k^T s_k > t_b$ holds; in other cases, the update yields Hessians with very large norms, and makes necessary to restart the iterative approximation $B_k = I$.

We have observed that update failures happen commonly, when the full Hessian becomes inde nite or ill-conditioned, being necessary to switch to steepest descent. Gauss-Newton BFGS provides a safeguard for such situations because it switches to Gauss-Newton instead $B_k = S_k^T S_k$, which has better convergence properties. Moreover, Gauss-Newton BFGS provides better overall approximations to the Hessian. Nevertheless, we have seen that Gauss-Newton steps are prone to overshoot the solution, and hence a conservative step-length selection procedure is advisable. In our work, we have opted for a simple line-search and for adaptively controlling the maximum allowed step length depending on success. However, it would be interesting to consider a more complex trust-region algorithm.

Projection scheme . Minimization in deformed space Q currently outperforms other options. If we restrict to the deformed space Q, the potential energy is su ciently well-behaved, and the various safeguard techniques de ned in section 3.4 guarantee the convergence and the robustness of the method. Our early attempts to employ the root- nding algorithm in total space S suggest that, in some cases, this approach does provide faster convergence rates of the inverse elastic design problem. However, projection times are often slower due to the lack of robustness.

Future work in this area arises from the limitations of our current methods. First, di erent higher-order approximations to the equilibrium constraint manifold could be considered. In particular, the work by Yang et al. [26] also employs second-order approximants, using quadratically parameterized osculant surfaces. It would be interesting to further study the relationship between that solution and the approaches analyzed in this chapter, seeking more accurate manifold approximations.

Additionally, the projection scheme based on the root- nding algorithm in total space R, to our knowledge, has not been explored yet in the context of inverse elastic design problem. This might open the possibility of formulating projection schemes that consider the design objective, hence improving the overall convergence. For that to be possible, we would need to formally characterize the problem, and devise techniques to overcome the numerical perils of the root- nding algorithm.

Chapter 5

Flexible rod meshes

Flexible rod meshes are lightweight structures for which heterogeneous global deformation behavior can be achieved using a single material, by locally varying the cross-section of the rods. In this chapter, we study the computational design and digital fabrication of exible rod meshes as an implementation of deformable surfaces. Built upon the mechanical model for the simulation of rod meshes that was introduced in chapter 3, section 3.2.2, and the inverse elastic shape design framework that was studied in chapter 4, we propose an optimization-based design tool that automatically designs printable rod structures that deform with desired behavior.

The rst section 5.1 introduces the problem, motivate the use of exible rod meshes as an option for the prototyping of deformable objects, and summarize the contributions of this work.

The second section 5.2 describes how we have used the tools introduced in previous chapters to create a computational model of the design space and deformation behavior.

The third section 5.3 explains how we solved the associated optimization problem using the sensitivity-based SQP algorithm described in chapter 4.

The fourth section 5.4 presents the results and fabricated physical prototypes that validate our method and discusses limitations and potential future work.

5.1 Introduction

In this work, we explore the use of exible rod meshes as an implementation of deformable surfaces. They are lightweight, relatively low-cost structures, that can also

form the support structure of solid objects if lled or dressed. But most importantly, the global deformation properties of a rod mesh can be adjusted simply by locally varying the cross-sectional parameters of the rods. As a result, a heterogeneous deformable object can be fabricated in one piece and from a single base material using a variety of rapid prototyping technologies.

We propose a computational tool for the example-based design and fabrication of exible rod meshes. Our method takes as input several poses of a deformable surface with known position and/or force constraints, and automatically computes a printable representation of a rod mesh that best approximates the desired shapes (Fig. 5.1). As we will see, the use of multiple deformed poses with di erent boundary conditions allow us to approximately de ne in a compact and easy manner the desired range of movements of a fabricated object. Starting from several deformation instances with di erent elasticity properties, we seek a unique design that generalizes all the behaviors. This facilitates reproducing complex deformation features like anisotropy, heterogeneity, or model merging (see section 5.4).

Figure 5.1: Comparison between hat simulations and printed results. The two left columns show the hat with the default rod mesh, which does not deform as desired (shown in transparent gray). The two right columns show that, by optimizing the radii and rest-shape of the rod mesh, we can fabricate in one piece a hat that deforms as desired.

One of the main features of our method is the choice of design space. We use rod meshes dominated by hexagonal faces, because hexagons can stretch, shear and bend by deforming their edges. Given such a mesh, the design space consists of its rest shape and two orthogonal radii describing the ellipsoidal cross section of the rods. By adjusting two orthogonal radii, we manage to control in-plane deformations (stretch and shear) and out-of-plane deformations (bending) of the surface independently.

In order to estimate the design parameters, we propose a simulation-based optimization approach. To this end, we extended the discrete elastic rod model [45, 46] to handle connections between multiple rods. The resulting rod mesh mechanical model has been describe in chapter 3, section 3.2.2. Built upon this simulation model, we propose an optimization framework that determines design parameters

by minimizing the approximation error with respect to the input poses while satisfying static equilibrium constraints. In order to handle these constraints e ciently, we compute gradients of the objective function that satisfy the constraints implicitly using sensitivity analysis, also already introduced in the last chapter 4, section 4.3.1.

We have applied our design and fabrication method to a variety of examples. In particular, we have explored the potential of the method for toy and apparel design. In order to validate the behavior of our designs in practice, we additionally fabricated three physical prototypes.

The rest of the chapter is structured as follows. First, in section 5.2 we describe the computational model upon which this design tool is built. We brie y recall the fundamentals of the discretization used in our mechanical model and describe in depth how it is related with the de nition of the design space. Then, we explain the formulation of our optimization problem and the speci cs of the scheme we have used for solving it. To conclude, in section 5.4, we show some fabricated results along with a discussion on the limitations and future work.

5.2 Computational model

In this section, we describe the computational model used for representing the deformation behavior and design space of a rod mesh, following the framework introduced in chapters 3 and 4. Then, we formulate the optimization problem solved for automatically nding the design.

5.2.1 Mechanical model

As introduced in section 3.2.2, we have extended the discrete elastic rod model [45, 46] to handle connections between multiple rods. We have followed a co-rotational approach to estimate the orientations of connections kinematically from incident rods, we have de ned a coupling energy and we have showed how to transmit bending and twist forces between connected rods. Our model derives completely from the classic theory of Kirchho rods and features elastic behavior at the connections, modeling realistic large deformation ever for relatively coarse discretizations. This fact together with the absence of numerical constraints allows us to formulate and solve e ciently a complex optimization problem.

Recall from chapter 3, section 3.2.1, that our mechanical model discretize continuous adapted-frame rod kinematics in two sets of variables: centerline nodes c_i 2 R ^{3}, which explicitly describe the con guration of the curve, and edge roll angles $_i$ 2 R , which represent the material frame as the rotation along the tangent of an adapted reference frame. Centerline nodes corresponding to rod ends incident

in the same connection are shared resulting in a total of n_n nodes and n_e edges. In this work, we adopt the implicit approach for handling the rotation of connections described in section 3.2.2 and hence no additional DoF are needed. The resulting deformed con guration is $q = \{c; \ g\} \in \mathbb{R}^{3n_n + n_e}$ with $c \in \mathbb{R}^{n_n}$ and $\ \in \mathbb{R}^{n_e}$. We denote magnitudes corresponding to the undeformed con guration as a and $\ $.

To ease the fabrication of the exible rod meshes, we assume that the whole rod mesh is built with a single material, and the mass and deformation properties of the mesh can be controlled by tuning the cross-section geometry of rods. To this end, the anisotropic bending model of Bergou et al. [21] provides the appropriate exibility. The continuous rod mesh cross-section shape is discretized at edges with a set of variables $r_i = \{r_i^b; r_i^n\}$, with r^b the radius in the binormal direction and r^n the radius in the normal direction. Hence, for a given undeformed con guration and fabrication material properties, the deformation behavior is completely determined by $r \in \mathbb{R}^{2n_e}$.

5.2.2 Design space

One of the main features of our approach is the choice of design space, which aims at minimizing fabrication complexity while maximizing design exibility. There are two major design decisions in our approach that make this possible: i) we de ne a hexagonal topology and keep it constant throughout the design process and ii) we select a compact parameter set that produces a smooth design through interpolation.

Figure 5.2: Comparison of the structural sti ness of a triangle, a quad and a hexagon made of rods. We xed the bottom-most edge of the structure, enforced the indicated vertex to be at each position in the eld, and computed the static equilibrium. The plot shows the resulting deformation energy. Notice that the area of low deformation energy under stretch is clearly bigger for the hexagonal mesh.

Hexagonal topology

We seek for a mesh topology that minimizes structural sti ness, thus enabling a larger range of feasible designs under the same magnitude of external forces. Rods are radically more compliant in bending than stretch, hence the topology of the mesh should be one that enables deformations in all directions simply by bending rods as seen in Fig. 5.2. We have opted for a mesh topology dominated by hexagonal faces, as hexagons can deform in all directions by bending their edges. Fig. 5.3 compares the structural sti ness of a triangle-mesh, a quad-mesh, and a hexagon-mesh of rods for a particular deformation scenario. The hexagon-mesh can stretch and shear in all directions, the quad-mesh resists stretch in directions aligned with quad edges, and the triangle-mesh is almost inextensible.

Figure 5.3: With the same material and mesh density, a hexagonal mesh is more compliant to stretch and shear deformations than a triangle or quad mesh, thereby increasing the exibility of our optimization method. In this example, we show meshes with approximately the same number of cells, in their rest con guration (top), and after hanging from them the same weight (bottom).

To construct a mostly hexagonal mesh, we have explored two options in our

examples. One is building the dual mesh of an input triangle mesh. The other one is to initialize a number of samples on a surface and then compute a centroidal Voronoi tessellation [188]. With hexagon meshes, we have observed that our optimization method is largely insensitive to the specific mesh topology. Fig. 5.4 shows an optimization result that is matched using three different meshes.

Figure 5.4: A hat mesh is optimized to reach the target deformation (shown in light gray) with three different meshings. All meshes were obtained through Voronoi tessellations; the first two with 64 cells and different initializations, and the third one with 128 cells.

Compact parameterization

Our design space consists of the rest shape of the rod mesh $c \in \mathbb{R}^{3n_n}$ and the two radii of the elliptical cross-section at each point of rod mesh $r = \mathbb{R}^{2n_e}$. By adjusting the radii, we manage to control the overall stiffness of the mesh. Tuning also the rest configuration allow us to obtain even more accurate results when considering multiple target poses.

Both parameter sets change the geometry of the mesh and hence the appearance of the printed object. Wiggly designs affect the aesthetics of the mesh but also compromise the capability of the discretization to approximate the continuum deformation behavior. Furthermore, in this multi-objective scenario, there is high chance of overfitting. This problem not only reduces the generalization properties we seek, but also produces noisy results if the optimization method locally overfits for one particular target. With these observations in mind, we follow the heuristic for the definition of a compact design space with naturally smooth shapes introduced section 4.2.1.

Both for radii and rest positions, we set as design parameters control points along the rods c_c and r_c. We place four control points per rod, two at the end connections and two evenly spaced, and smooth their values using cubic Hermite interpolation. Overall this choice results in a parameter vector $p = (r_c; c_c)$, with $(r; c) = Zp$, for a

precomputed matrix Z of cubic Hermite coefficients. To further improve the visual appearance and deformation behavior of the resulting rod meshes, we also enforce continuity and alignment of cross-sections at connections in two ways:

1. Connection sharing. Analogously to the coupling technique used for mechanical simulation, we share control points among all rods incident in a connection, such that radii magnitude smoothly vary throughout the rod mesh.

2. Normal alignment. We keep material and reference frames at connections aligned in the rest configuration. Whenever the rest shape changes, we fit a plane to the incident edges of each connection. We use the normal of this plane as the normal axis of rest-shape frames of incident edges. This choice typically defines the first and the last rest frames of each rod in the mesh; the alignment of the other frames of the rod is linearly interpolated from the extremal values. We span the orthogonal cross-section radii at each point in the rod mesh using material frames. Thus, as an additional effect of frame alignment, we gain more flexibility during optimization because adjusting the two orthogonal radii is guaranteed to control tangent plane and out-of-plane deformations.

5.3 Optimization problem

Our optimization framework takes as input a few deformed poses of a rod mesh under known boundary conditions. Then, it automatically computes the rest shape and cross-section of the rods such that the mesh best matches the input poses under the same boundary conditions. The resulting rod mesh geometry is finally used as input for an automatic fabrication process.

In this section, we first describe the general optimization framework. We use a Newton optimization method subject to boundary conditions, design constraints, and static equilibrium constraints, which are enforced implicitly. Then, we describe an optimization scheme to optimize both the material (i.e., radii) and rest shape of the rod mesh.

5.3.1 Numerical optimization

Following the notation described in chapter 4, our optimization framework receives as input a set of k_p target poses $\{q_k^t\}$ in static equilibrium, and for each pose a subset of DoFs $b_k = B_k q_k$ specified as fixed boundary conditions. Here B_k is a per-pose selection matrix. The method also supports using forces as boundary conditions.

We denote as p a generic set of m design parameters, which could be shape parameters (i.e., rest-shape coordinates of rod nodes) and/or material parameters (i.e., radii of rod cross-sections). We then formulate the objective as the squared error between rod node positions and the input poses:

$$h(p; \{q_k^t\}; \{b_k\}) = \frac{1}{2} \sum_{k=1}^{k_p} w_k \| \quad (q_k) \qquad (q_k^t) \|^2; \tag{5.1}$$

where is a shape descriptor that selects only the nodes of the deformed and target con guration. Then, we pose a constrained optimization that minimizes this objective function, subject to static equilibrium, box constraints on design parameters, and the input boundary conditions:

$$\begin{aligned} &(q; p) = \arg\min h(p; \{q_k^t\}; \{b_k\}); \\ &\text{s.t. } f_{q;k}(q_k; p; b_k) = 0; \forall k: \\ &p_m \quad p \quad p_M \end{aligned} \tag{5.2}$$

$f_{q;k}$ denotes a vector that concatenates the forces on all rod nodes for the kth pose, w_k is a scalar to weight target poses, and p_m, p_M are minimum and maximum constraints on design parameters. To solve this constrained optimization problem, we follow the sensitivity-based SQP approach de ned in section 4.3.2: we enforce the equilibrium constraints implicitly and iterate QP updates of the design parameters with box-constraint projection.

In practice, for the calculation of the sensitivity matrix S_k de ned in section 4.3.1, we compute $J_{qq;k}$ analytically, but we evaluate $J_{qp;k}$ using nite di erences to facilitate the exploration of arbitrary design parameters. By handling the static equilibrium constraints implicitly, the minimization in (5.2) turns into a box-constrained nonlinear least-squares problem. To handle box constraints on the radii, we simply project the gradient for those parameters that have reached a limit. Depending on the approximation of the Hessian of (5.2) and the equilibrium projection scheme, the convergence properties of the algorithm vary.

Quadratic subproblem . Computing the full Hessian of (5.2) is prohibitive, so we have tested approximate methods instead, speci cally Gauss-Newton with line-search and Levenberg-Marquardt. Both methods approximate the Hessian using the sensitivity matrix, and we found them to behave similarly in our examples. We have also observed that the full Hessian occasionally becomes inde nite and its approximations ill-posed. This induces close to non-descendent search directions, being necessary to switch to steepest descent. As a compromise solution, BFGS with line-search provides enough robustness as Hessian updates are often invalid in such situations, thus switching to steepest

descent. Plus, controlling the step length is key to the performance of our method; therefore, we enforce a maximum length which is adaptively re ned depending on success.

Equilibrium projection . After every QP update of the parameter vector p, we solve a static equilibrium problem on all poses. To do this, we use the energy minimization solver introduced in section 3.4.

Finally, to apply position boundary conditions $\{b_k\}$, we found that xing simulation nodes increases excessively the error on free nodes. Instead, we enforce boundary conditions using a penalty force, and we progressively adjust the penalty sti ness such that the error at constrained nodes is similar to elsewhere in the mesh.

5.3.2 Optimization scheme

As already mentioned, the vector of design parameters p may be formed by material parameters (i.e., rod cross-section radii) and/or shape parameters (i.e., rest-shape coordinates of rod nodes). We have observed that adjusting cross-section radii has the largest e ect on the bulk tting error, while adjusting rest-shape coordinates increases tting quality for designs that are close to the optimum and have to deal with objectives. For this reason, we propose the following optimization scheme.

We start with a multiresolution optimization of the material of the rod mesh. We rst optimize only the radii control points at connections, linearly interpolated along rods; then we add radii control points in the middle of rods, with quadratic interpolation; and we nally optimize for all radii control points with cubic interpolation. Once the material optimization alone has converged, we start iterating steps of material and shape optimization directly on all control points. Please see Fig. 5.5 for examples of convergence with our optimization scheme

5.4 Experimental results

In all our examples we follow a similar design pipeline, with small variations. We start with a surface description of the deformable object to be fabricated, typically a high-res triangle mesh H. At this point, we de ne the connectivity of the rod mesh R to be optimized. If the user does not provide a speci c topology, we automatically compute one such that it is close to an hexagonal mesh. We project the resulting connections onto H, and construct rods connecting them by following geodesics. For each vertex of R, we store a mapping to its projection triangle in H.

To create the target poses, we apply some deformer to H. In our examples, we have explored di erent deformers, mainly direct artist manipulation and embed-

ding in another physical model. Given a target deformation de ned on $\mathbb{H}$, we can determine the target con guration of the rod mesh by its stored mapping.

Table 5.1 summarizes the rod mesh size, complexity, continuum material parameters and radii constraints for all our benchmarks. We have printed some of the obtained results to serve as a validation of both the simulation model and the optimization procedure. In all the cases, we have used laser sintering technology with the material TPU 92A (elastic modulus $E = 32$MPa, Poisson ration $= 0.48$ and density $= 1200$Kg=m^3).

Model	Sheet	Hat	Dino	Smiley
Nodes	693	1727	3219	2663
Edges	710	1790	3330	2892
Rods	71	179	333	723
Connections	54	116	222	494
Material Params.	392	948	1776	2664
Rest-Shape Params.	588	1422	2664	3663
Young mod. (MPa)	32	32	32	10.0
Poisson ratio	0.48	0.48	0.48	0.48
Density (Kg/m^3)	1200.0	1200.0	1200.0	1000.0
Size (m)	0.2	0.23	0.20	0.25
Max. Radius (mm)	5	3	3	4
Min. Radius (mm)	1	1	1	1
RMS Error (mm)	1.03	1.13	0.56	0.41
Computation Time	35min	1h45min	2h20min	3h10min

Table 5.1: Statistics of the benchmarks: model discretization, optimization complexity, mechanical parameters, design constraints, resulting RMS error, and computation time.

5.4.1 Performance

Our rod structure model introduces additional computational complexity to the discrete elastic rods approach. Computing the energy gradient and Hessian at connections is costly and involves a reduction in performance. However, we are aiming for sparse rod structures therefore we can expect the overall complexity to grow linearly on the number of rods.

Our optimization scheme considers implicit constraints for each target pose, which are independent from each other. The step is broken into a set of individual problems which can be solved separately, and the complexity grows linearly in the number of target poses. As a counterpart, two operations are specially costly: i)

Figure 5.5: Convergence of the optimization for the three printed results, showing the evolution of RMS error (in log scale). White dots indicate the iteration from which the optimization alternates radii and rest-shape optimization.

performing the sensibility analysis and ii) computing the static equilibrium.

The cost of the sensitivity analysis is partially alleviated by incrementally adding resolution to the parameterization. As shown in Fig. 5.5, error is greatly reduced in the rst steps of the optimization, when only one control point per connection is used. However, improvement gained by subsequentially re ning the parameterization and iterating radii and rest pose optimization steps is signi cant. At this stage of the optimization, steps are more time-consuming so a trade-o must be considered between computation time and the quality of the solution.

The cost of computing the static equilibrium is highly dependent on the step length, as the static solve is computed from the previous equilibrium con guration. If the new static equilibrium is too far from the previous one, it is more convenient to compute equilibrium from the rest con guration. Moreover, convergence problems may activate line-search bisections, leading to additional static solves. Adaptively controlling the maximum allowed step length, along with using the quasi-Newton update helps to reduce failed steps and has an important positive impact on performance. Table 5.1 shows an approximation of the time needed to converge to the optimal solution for each benchmark.

5.4.2 Printed examples

Sheet

Our first example consists of a simple sheet (Fig. 5.6), showing the capability of hexagonal rod meshes of exhibiting varied behaviors. The rod mesh is 200mm side and consists of 71 rods and 54 connections forming a regular hexagonal pattern, with a total of 693 nodes.

Four target poses are defined by computing the static equilibrium of the rod mesh for different radii configurations and boundary conditions (fixed points and weights). These targets are designed to be opposing: the first pair (first and second row) imposes bending anisotropy; the second pair (third and forth row) imposes stretching capabilities. The optimized rod mesh effectively captures all required behaviors. Simulated results has been validated by testing physical printed realizations, which closely match the behavior predicted by the simulation. Although low frequency deformations are perfectly replicated, small differences appear which may be cause by inaccuracy in the validation process.

The images in Fig. 5.6 show in semi-transparent gray the target configuration of the rod mesh. The default behavior at the beginning of the optimization is shown in red against the deformations obtained with our optimization framework in green. Cross-sectional radii optimization is capable of reducing the RMS error to 4.1756mm per node. We achieve an overall fitting of 1.0322mm upon convergence of rest-shape optimization.

Hat

Our second example is a hat of approximately 0.23m long, shown in Fig. 5.7 and Fig. 5.1. The connectivity of the rod mesh is automatically computed using a centroidal Voronoi tesselation on a high-res triangle mesh. This yiedls to a close-to-hexagonal mesh consisting of 179 rods, 116 connections and a total of 1727 nodes.

Five target poses (rows in Fig. 5.7) are defined using a thin-shell deformer on the triangle mesh. We have designed several heterogeneous models and used them to create target poses. Target poses 1, 2 and 3, use a model where the front side of the hat is much softer than the back side. Target pose 4 uses a homogeneous model and is intended to make the hat maintain its rest shape under gravity. Target pose 5 uses a model where the left side of the hat is clearly softer than the right side. We fix some points and pull from handles using forces to achieve interesting deformations. The deformed position of the handle is considered a boundary condition for our simulated rod mesh.