TECHNISCHE UNIVERSITÄT MÜNCHEN
TUM School of Engineering and Design

Automatic design of compliant medical instruments using bionic structural optimization methods

Yilun Sun

Vollständiger Abdruck der von der TUM School of Engineering and Design der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften (Dr.-Ing.)

genehmigten Dissertation.

Vorsitz: Prof. Dr.-Ing. Wolfram Volk

Prüfer der Dissertation:

1. Prof. Dr. rer. nat. Tim C. Lüth
2. Prof. Dr. Markus Zimmermann

Die Dissertation wurde am 18.03.2021 bei der Technischen Universität München eingereicht und durch die TUM School of Engineering and Design am 29.11.2021 angenommen.

Acknowledgements

Completing this dissertation, a product of three years' work, I would like to thank a lot of people who have greatly inspired and supported me during my study and research at the Institute of Micro Technology and Medical Device Technology (MiMed, TUM). Throughout these years, I have learned a lot and my research ability has continually improved.

In particular, I would like to thank my supervisor, Prof. Dr. rer. Nat. Tim C. Lueth, for his previous guidance, academic stimulus and generous help. From the inception on, Professor Lueth has devoted so much into my research. I would never forget the very first discussion with him about the structural optimization, which was the beginning of my research. Besides, he always encouraged me and led me into the field of biomedical engineering. Without his supervision, I could not have completed this challenging doctoral research.

I would also like to thank my colleagues, Alexandra Mercader, Suat Coemert, Christoph Rehekampff, Yanick Krieger, Simon Schiele, Dingzhi Zhang, Samuel Detzel and Christian Dietz for their help in my research and teaching.

Besides, I offer my thanks to Mrs. Renate Heuser and Dr.-Ing. Franz Irlinger for their help in my studies and work at MiMed.

In addition, I would like to thank my students, Yuqing Liu and Lingji Xu, for their contributions to my research studies.

Furthermore, I want to thank Prof. Dr.-Ing. Houde Dai and Prof. Dr. Angela Faragasso for their patient guidance and help with my research and publications.

Finally, I would like to thank my family for their eternal support and encouragement.

Yilun Sun, March, 2021

Abstract

Compliant mechanisms are widely used in the design of medical instruments because of their monolithic structure and high flexibility. Many compliant mechanisms derive their design ideas from nature, since the structure of biological organisms sometimes offers a better solution than the conventional mechanisms. However, the bio-inspired structures usually have very complex geometries which cannot be easily modeled and synthesized using traditional methods. In this dissertation, the author presents an automatic design framework to simplify the design process of bio-inspired compliant medical instruments. FEM-based mechanics modeling methods and topology optimization based synthesis methods are employed to realize the automatic design. Several design cases are also presented to show the performance of the proposed framework. The results of the conducted simulation and experimental tests have demonstrated that, the developed method greatly simplifies the design process and makes it possible to efficiently realize task- and patient-specific compliant medical instruments.

Keywords: Automatic design, topology optimization, compliant mechanism, medical instrument.

Zusammenfassung

Aufgrund der monolithischen Struktur und der hohen Flexibilität werden die nachgiebigen Mechanismen häufig beim Entwurf der medizinischen Instrumente eingesetzt. Viele nachgiebige Mechanismen sind bio-inspiriert, da die Struktur der biologischen Organismen manchmal eine bessere Lösung bietet als die konventionellen Mechanismen. Es ist jedoch kompliziert und ineffizient, mit den konventionellen Methoden die bio-inspirierten Instrumente zu modellieren und synthetisieren, weil sie normalerweise sehr komplexe Geometrien haben. In dieser Dissertation stellte der Autor ein automatischer Entwurfswerkzeug vor, um den Entwurfsprozess von bio-inspirierten nachgiebigen medizinischen Instrumenten zu vereinfachen. FEM-basierte Modellierungsmethoden und Topologieoptimierung-basierte Synthesemethoden wurden eingesetzt, um den automatischen Entwurf zu realisieren. Einige Synthesebeispiele wurden vorgestellt, um die Leistungsfähigkeit des vorgeschlagenen Werkzeugs zu zeigen. Die Ergebnisse der durchgeführten Simulationen und Experimente haben gezeigt, dass die entwickelte Methode den Entwurfsprozess stark vereinfacht und es ermöglicht, aufgaben- und patientenspezifische nachgiebige medizinische Instrumente effizient zu realisieren.

Schlagwörter: Automatischer Entwurf, Topologieoptimierung, nachgiebiger Mechanismus, medizinisches Instrument.

Contents

1 Introduction

1.1 Compliant Mechanisms in Medical Applications

Medical instruments, such as forceps and scissors, are important tools for the surgeons to perform operations on the patients. In last decades, to improve the performance of the surgery and also to reduce patients' trauma, miniaturized surgical instruments were developed, which allow the surgeons to conduct complicated operations within incisions of a few millimeters. In those minimally invasive surgeries (MIS), the end-effectors of the instruments are usually fabricated in micro-scale size but show a high degree of dexterity to accomplish the complicated operation tasks in a constrained workspace. With the help of MIS, patients can have a much shorter hospital stay and leave less scarring. However, in order to achieve complex movements, such as the laparoscopic suturing, advanced surgical techniques are required from the surgeons' side, which becomes a remarkable obstacle in promoting the use of the miniaturized surgical instruments. To cope with this problem, robot-assisted systems, such as the da Vinci system (Guthart and Salisbury, 2000), have incorporated the endoscopic camera and computer-aided control systems to reduce the difficulty of using the MIS instruments.

Currently, many MIS instruments and robots are based on rigid-link mechanisms, whose motions are enabled by rotational joints (see Fig. 1-1a)). Although the rigid-link-based medical instruments are stable and robust, limitations related to the material strength and the accuracy, when fabrication is performed in micro-scale, make their miniaturization difficult. To cope with this problem, compliant mechanisms are incorporated into the design of MIS instruments (see Fig. 1-1b)). Unlike the rigid-link mechanisms, the compliant mechanisms gain at least some of their mobility from the deflection of flexible members rather than only from movable joints. With this unique feature, it is possible to design a dexterous instrument using a monolithic structure. In this way, the problem of wear and additional lubrication after long-term use can be avoided. Besides, the compliant mechanism based instruments are easy to assemble and sterilize. Since the high flexibility of the compliant mechanisms is quite similar to the nature of living creatures, many compliant medical instruments have a bionic design.

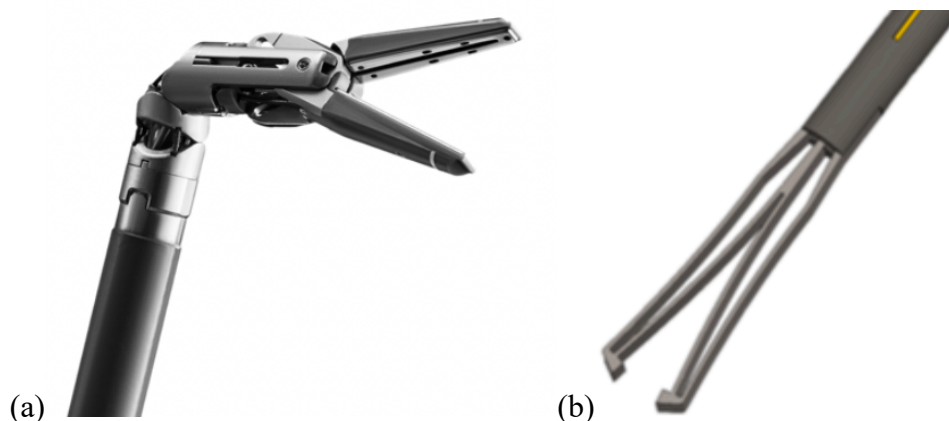(a)   (b)

Figure 1-1: Rigid-link-based and compliant MIS forceps: (a) A rigid-link-based forceps of the da Vinci surgical system (reproduced from Intuitive Surgical Inc., 2020), (b) Compliant MIS forceps for robotic vitreoretinal surgery (reproduced from Gonenc *et al.*, 2013).
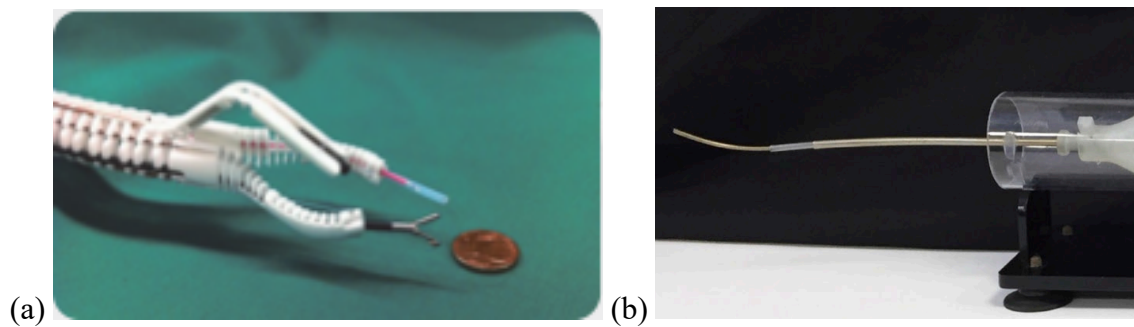
Figure 1-2: 3D-printed compliant medical instruments: (a) A SLS-printed multi-arm snake-like robot (reproduced from Roppenecker et al., 2014), (b) A SLS-printed concentric tube robot (reproduced from Morimoto, Hawkes and Okamura, 2017).

However, due to the high complexity of the shape of the bionic compliant instruments, it is difficult to use the traditional manufacturing methods, such as milling and drilling, to fabricate them. To cope with this problem, many research studies were carried out, using the additive manufacturing technologies as a new strategy to prototype compliant instruments. For example, researchers have used selective laser sintering (SLS) technology to fabricate a snake-like continuum robot (Roppenecker *et al.*, 2014), where the polyamide PA2200 (EOS GmbH, Germany) was employed as 3D printing material (see Fig. 1-2a)). In other studies, another kind of polyamide, the PEBA2301 (EOS GmbH, Germany), was employed to prototype a concentric-tube-based continuum robot (see Fig. 1-2b)), also using SLS printing (Morimoto, Hawkes and Okamura, 2017).

1.2 Automatic Design of Compliant Medical Instruments

The mechanical mechanism design is generally comprised of two basic tasks: mechanism modeling (analysis) and mechanism synthesis. The first task, the mechanism modeling, is used to determine the character motion of a prescribed mechanism. As for the second task, the mechanism synthesis, the aim is to find a proper mechanism which can realize a predefined motion. For the design of conventional medical instruments, the rigid-link-based mechanism theory is used since those instruments are usually assembled by rigid parts. However, due to the continuum-structure effect of the compliant mechanisms and the resulting coupling of kinematics and dynamics, the conventional rigid-link-based mechanism theory cannot be directly applied in the modeling and synthesis of compliant instruments. Although some modified kinematic methods, such as the pseudo-rigid-body-model (PRBM) method, can achieve the modeling and shape synthesis of compliant mechanisms based on specific mechanism models, they are still inefficient to realize the general design of different kinds of compliant mechanisms. To cope with this problem, research studies are conducted in the Institute of Micro Technology and Medical
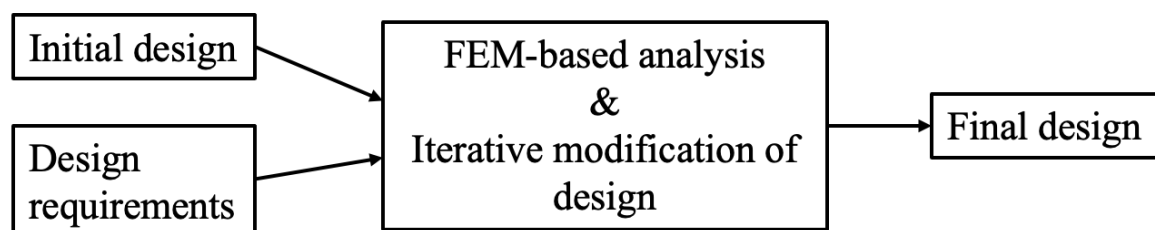

Figure 1-3: The concept diagram of automatic design of compliant medical instruments.

Device Technology (MiMed) at Technical University of Munich (TUM) with the aim to develop advanced methods to accelerate the general design process of compliant medical instruments. The basic idea of the automatic design concept is illustrated in Fig. 1-3. Instead of performing the entire design process empirically, the engineer only needs to provide an initial design and the corresponding design requirements (boundary conditions). The initial design should be analyzed by finite element method (FEM) based simulations and then iteratively modified until the prescribed requirements are fulfilled. In this way, the design potential of the compliant instruments can be deeply explored and is not be limited by the experience of the engineers.

1.3 Research Objectives

In this dissertation, the author aims to develop a systematic design framework to realize the automatic design of compliant medical instruments. FEM-based bionic structural optimization methods will be employed as key techniques to achieve the automatic synthesis goal. The entire framework will be implemented in a single developing environment (MATLAB) so that the code can be easily modified and extended. The following topics will be included in the dissertation to illustrate and analyze the proposed framework:

- FEM-based mechanics modeling methods will be developed to achieve high-fidelity modeling of compliant medical instruments, where large displacements, tendon-driven mechanisms and contact problems are taken into consideration.

- A 2D topology optimization based methodology will be developed to achieve the automatic synthesis of compliant medical instruments, whose geometry is based on the extrusion of optimized 2D contours.

- A 3D topology optimization based methodology will be developed to extend the 2D-based design algorithm, so that fully 3D features will be incorporated into the design results.

- In order to study the efficiency of the proposed design framework, impact of different design inputs on the final results will be analyzed. Herein, the design results based on 2D and 3D optimization algorithm will be compared. Other parameters, such as volume fraction and geometrical constraints, will be analyzed as well.

- In order to verify the design methods proposed in the framework, several design applications will be presented. The realized design proposals will be fabricated using different additive manufacturing technologies. Their performance, such as the expected movements and stiffness, will be evaluated by experimental tests.

1.4 Organization of the Dissertation

This dissertation is organized as follows: Chapter 2 firstly describes the current state of the art of the design methods for compliant mechanisms. The disadvantages of the current state of the art are also discussed in Chapter 2. Chapter 3 introduces the developed FEM modeling methods which are the basics of our design framework. Chapter 4 provides a detailed illustration of the topology optimization based synthesis methods, including the 2D and 3D algorithms, and their implementation in MATLAB. Chapter 5 analyzes the design results of the proposed synthesis methods by comparing the 2D and 3D optimized results and studying the impacts of different design inputs. Chapter 6 presents several design applications to compliant medical instruments

and their experimental evaluation to verify the feasibility of the proposed design framework. Finally, the work in this dissertation is concluded in Chapter 7, where the future work is also outlined.

2 State of the Art

This chapter first reviews the methods in the literature for modeling and synthesizing compliant mechanisms (Section 2.1 and 2.2). Then, current applications of the design methods to compliant medical instruments are presented (Section 2.3). In Section 2.4, the limitations of the current state of the art are discussed.

2.1 Modeling Methods for Compliant Mechanisms

As is already mentioned in Chapter 1, it is not easy to model and analyze compliant mechanisms using the classic rigid-link mechanism theory, since the deformable continua cannot simply be treated rigid bodies. To cope with this problem, a lot of methods have been developed to incorporate the elastic property into the kinematic analysis of compliant mechanisms. In this section, 4 types of such methods are introduced, which are the Pseudo-Rigid-Body method, the beam constraint model method and the finite element method.

2.1.1 Pseudo-Rigid-Body-Model (PRBM) Method

Lumped Compliant Mechanisms (With Flexure Hinges)

The concept of PRBM method was firstly established by Howell in his work (Howell and Midha, 1994), in which he simplified the flexible segments in the compliant mechanism as torsional springs attached to rigid-body joints (see Fig. 2-1). In that work, the simplifying assumptions were made for lumped compliant mechanisms as the flexural pivots (also called flexure hinges) are much smaller than the lengths of the rigid parts. The reason was mathematically illustrated
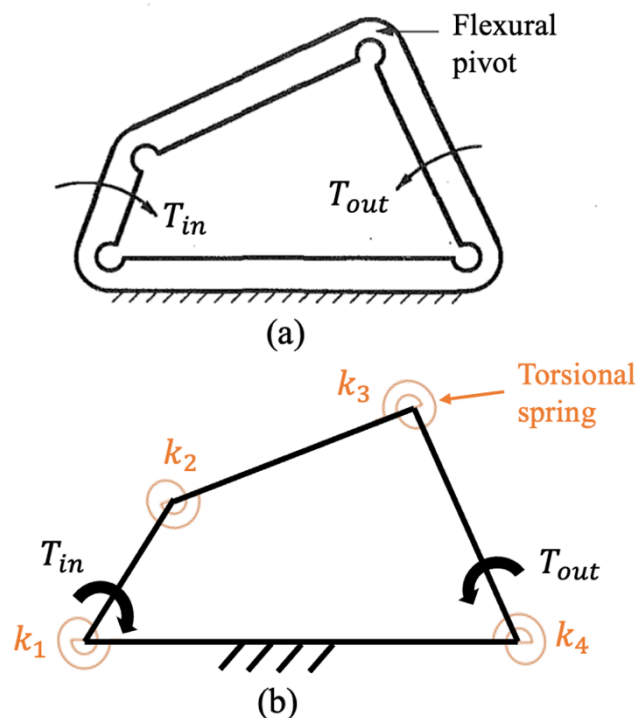
Figure 2-1: A compliant mechanism and its pseudo-rigid-body model (reproduced from Howell and Midha, 1994): a) A four-bar compliant mechanism, b) Its pseudo-rigid-body model.
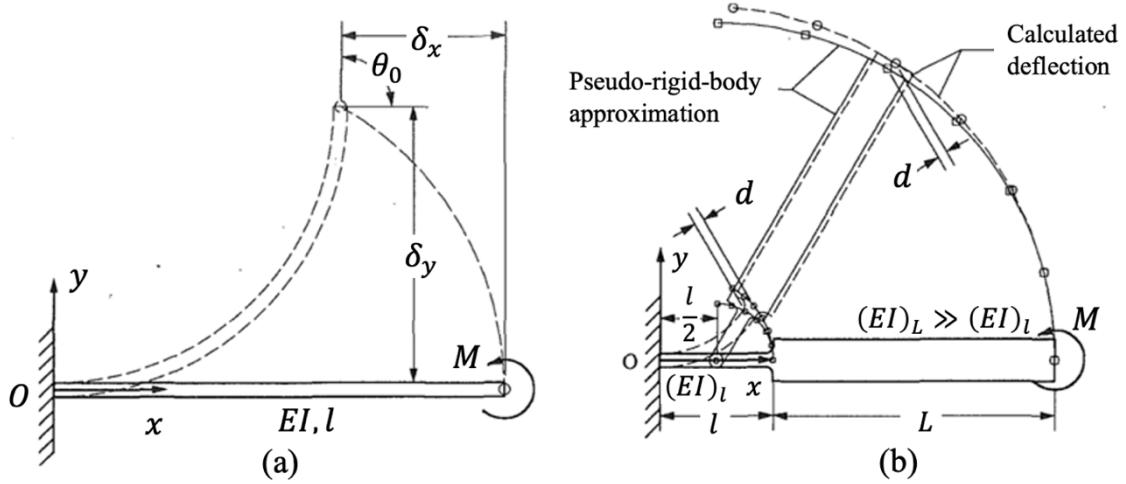
Figure 2-2: The graphical illustration of the simplifying assumptions of the PRBM method (reproduced from Howell and Midha, 1994): a) Large deflection of a flexible beam with an end moment, b) A pseudo-rigid-body approximation of a beam consisting a flexible and a rigid segment.

as follows. Fig. 2-2a) shows a flexible beam with uniform cross-section. The deflection of the beam when applying an end moment M can be described by the Bernoulli-Euler beam equation:

$$\frac{d\theta}{ds} = \frac{M}{EI} \tag{2-1}$$

where $\frac{d\theta}{ds}$ is the rate of change in angular deflection along the beam. x and y are the coordinate of the undeflected beam axis and the transverse deflection respectively. EI represents the flexural rigidity. The angular deflection θ_0 in Fig. 2-2a) can be obtained after separating variables and integrating Equation (2-1):

$$\int_0^{\theta_0} d\theta = \frac{M}{EI} \int_0^l ds \tag{2-2}$$

$$\theta_0 = \frac{Ml}{EI} \tag{2-3}$$

The vertical deflection δ_y and horizontal deflection δ_x can also be calculated by combining Equation (2-1) and (2-3) and performing variable separation:

$$\frac{\theta_0}{l} = \frac{d\theta}{ds} = \frac{d\theta}{dy}\frac{dy}{ds} = \frac{d\theta}{dy}\sin\theta \tag{2-4a}$$

$$\frac{\theta_0}{l} = \frac{d\theta}{ds} = \frac{d\theta}{dx}\frac{dx}{ds} = \frac{d\theta}{dx}\cos\theta \tag{2-4b}$$

$$\frac{1}{l}\int_0^{\delta_y} dy = \frac{1}{\theta_0}\int_0^{\theta_0} \sin\theta\, d\theta \tag{2-5a}$$

$$\frac{1}{l}\int_0^{\delta_x} dx = \frac{1}{\theta_0}\int_0^{\theta_0} \cos\theta\, d\theta \tag{2-5b}$$

$$\delta_y = l\frac{1-\cos\theta_0}{\theta_0} \tag{2-6a}$$

$$\delta_x = l\frac{\theta_0 - \sin\theta_0}{\theta_0} \tag{2-6b}$$

If the beam shown in Fig. 2-2b) is comprised of a flexible segment (length l and flexural rigidity $(EI)_l$) and a rigid segment (length L and flexural rigidity $(EI)_L$) with the condition $(EI)_L \gg (EI)_l$, the deflection of the new beam can be calculated from Equation (2-6a) and (2-6b). The dashed line in Fig. 2-2b) shows the calculated deflection curve. On the other hand, the authors in (Howell and Midha, 1994) stated that the moving trajectory can also be approximated by two rigid links connected by a revolute joint at the middle ($l/2$) of the flexible segment. The resulted trajectory is shown in solid line in Fig. 2-2b). When θ_0 is given and l is much smaller than L, the calculated and approximated beams are parallel and very close to each other. From this point of view, the authors in (Howell and Midha, 1994) indicated that the simplified pseudo-rigid-body model can be used to analyze compliant mechanisms with small-length flexure hinges. The bending stiffness k of the simplified torsional spring can be calculated as:

$$k = \frac{M}{\theta_0} = \frac{EI}{l} \tag{2-7}$$

Distributed Compliant Mechanisms

Different from the lumped compliant mechanisms, the deformations of the distributed compliant mechanisms are not concentrated in small-length flexure hinges but in slender beam-like segments. The modeling theory of the lumped compliant mechanisms (Howell and Midha, 1994) is not applicable to the distributed compliant mechanisms since the prerequisite of small beam
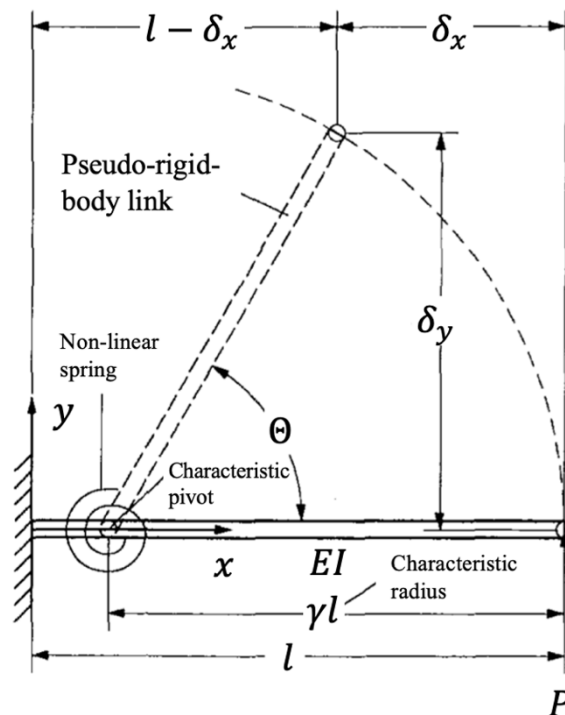
Figure 2-3: A pseudo-rigid-body model for a large-deflection beam with end forces (reproduced from Howell et al., 1995).
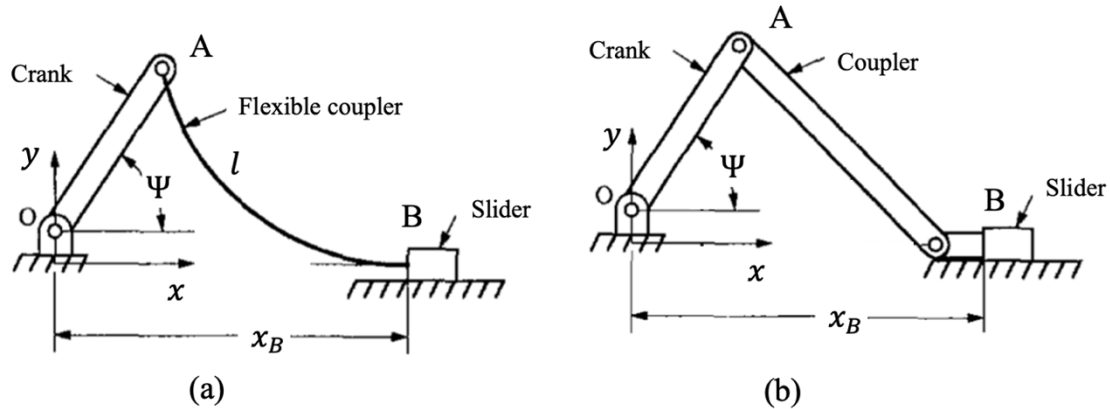
Figure 2-4: Modeling of distributed compliant mechanism using PRBM method (taken from Howell et al., 1995): a) A distributed compliant slider-crank mechanism, b) The corresponding PRB model.

length l is not valid anymore. To cope with this problem, Howell further developed the PRBM method in his work (Howell and Midha, 1995) to analyze the distributed compliant mechanisms. In that work, the authors have found that the motion paths of the end point of a large-deflection beam (see Fig. 2-2a)) are initially nearly circular when end forces are applied. This phenomenon was then used to approximate slender-beam-based compliant mechanisms. A modified pseudo-rigid-body model is shown in Fig. 2-3. Herein, the authors assumed that the nearly circular motion trajectory can be modeled by two rigid links connected at a pivot along the beam. Similar to the lumped compliant mechanisms, a torsional spring was also placed at the connecting pivot to represent the compliance of the beam. The difference is that the spring in Fig. 2-3 has a non-linear spring stiffness. The distance between the pivot and the end of the beam is γl, where γ is also called "characteristic radius factor". γ is determined by the Golden Section elimination method (Rao, 1983). The torsional spring stiffness, or the load-deflection relation, is approximated by a polynomial equation. In this way, the traditional rigid-link-based mechanism theory can be used for analyzing distributed compliant mechanisms. Fig. 2-4 shows a compliant slider-crank mechanism with a large-deflection flexible coupler and the corresponding pseudo-rigid-body model.
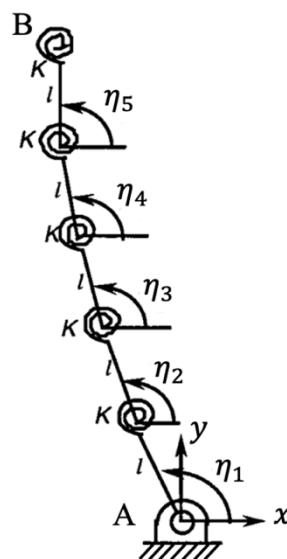
Figure 2-5: Modeling of flexible beam using several torsional springs and rigid linkages (reproduced from Saggere and Kota, 2001).

8
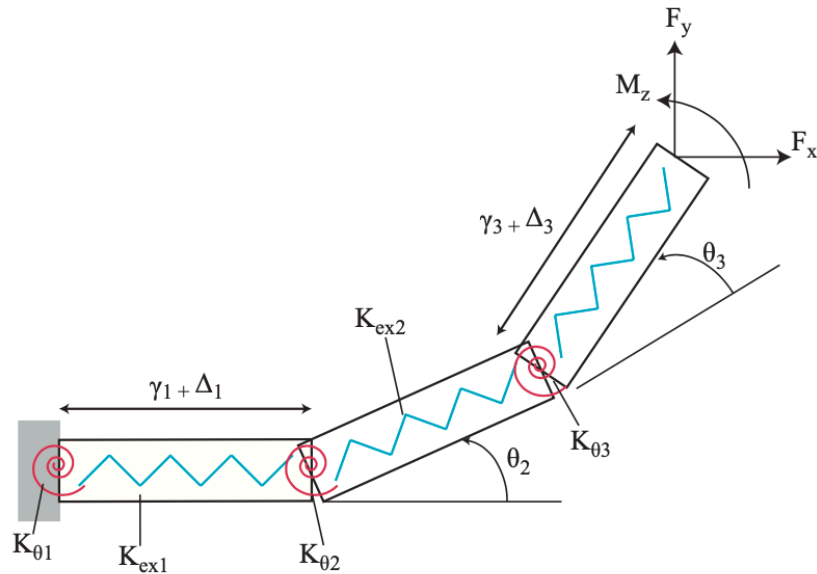
Figure 2-6: A PRB 3R model incorporating 3 torsional springs ($K_{\theta 1}, K_{\theta 2}, K_{\theta 3}$) and 3 extension springs ($K_{ex1}, K_{ex2}, K_{ex3}$) (reproduced from Venkiteswaran and Su, 2015).

The PRBM method for distributed compliant mechanisms was further developed by Saggere and Kota, 1999, where the flexible beam was decomposed into n (no less than 3) rigid segments connected by n linear torsional springs for modeling (see Fig. 2-5). The stiffness of each spring can be calculated as $K = EI/l$, where l is the length of each rigid segment. The advantage of the method in Saggere and Kota, 2001 is that the simplified spring stiffness is load independent and a wide range of loads can also be incorporated. However, the modeling accuracy depends on the number of decomposed segments and too many DOFs could lead to high computational cost. To cope with this problem, a PRB 3R model, consisting of 4 rigid linkages connected by 3 revolute joints and 3 torsional springs, was proposed in Su, 2009 to analyze distributed compliant mechanisms, where high accuracy of the proposed PRB model was also demonstrated. Other research studies, such as Venkiteswaran and Su, 2015, have improved the PRB model by incorporating elastic extension effect and shearing force of the flexible Timoshenko beams (see Fig. 2-6).

2.1.2 Beam Constraint Model (BCM) Method

Although the PRBM method can be successfully used for approximating lumped and distributed compliant mechanisms, its inherent lumped-compliance assumption has neglected elasto-kinematic effects. To cope with this problem, a closed-form analytical solution for planar

Figure 2-7: Generalized flexible beam model for BCM method (reproduced from Awtar, Slocum and Sevincer, 2007).

9

deflection was proposed in Awtar, Slocum and Sevincer, 2006 a to analyze medium-large deflections, using a so-called beam constraint model (BCM). The BCM model of the deflected flexible beam is presented in Fig. 2-7. y and θ represent the transverse deflections of the beam. $\boldsymbol{p}$ is the axial force. $\boldsymbol{f}$ and $\boldsymbol{m}$ are the transverse force and moment respectively. It was assumed that y is smalle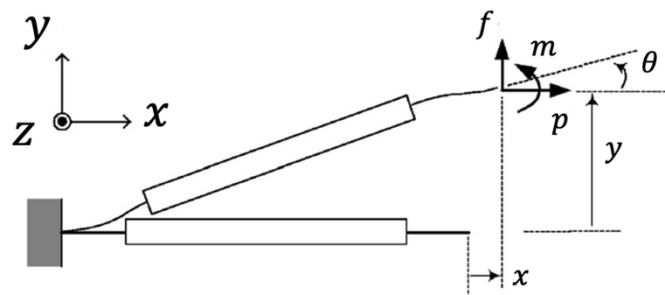r than 0.1 times of the beam length, so that the beam curvature can be linearized by assuming small slopes. Awtar, Slocum and Sevincer, 2006 have provided a general equation system to analytically describe the deflection of the flexible beam:

$$\begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{m} \end{bmatrix} = \begin{bmatrix} a & c \\ c & b \end{bmatrix} \begin{bmatrix} y \\ \theta \end{bmatrix} + \boldsymbol{p} \begin{bmatrix} e & h \\ h & g \end{bmatrix} \begin{bmatrix} y \\ \theta \end{bmatrix} \tag{2-8}$$

$$x = \frac{\boldsymbol{p}}{d} + [y \quad \theta] \begin{bmatrix} i & k \\ k & j \end{bmatrix} \begin{bmatrix} y \\ \theta \end{bmatrix} + \boldsymbol{p}[y \quad \theta] \begin{bmatrix} r & q \\ q & s \end{bmatrix} \begin{bmatrix} y \\ \theta \end{bmatrix} \tag{2-9}$$

where $d = 12/t^2$ and t is the uniform thickness of the beam. The parameters a, b, c, e, g, h, i, j, k, q, r and s were nondimensional numbers that are dependent on the shape of the beam. For a simple beam with uniform thickness, the values of these parameters are listed in Table 2-1.

Table 2-1: Modeling coefficients for a simple beam with uniform thickness (taken from Awtar et al., 2007).

a	12	e	1.2	i	-0.6	r	1/700
b	4	g	2/15	j	-1/15	s	11/6300
c	-6	h	-0.1	k	1/20	q	-1/1400

The authors in Awtar et al., 2007 stated that, the proposed analytical solutions were shown to yield very accurate closed-form load-deflection relations of the flexible beam and other beam-based compliant mechanisms. On the other hand, the presented results also contributed to a better physical understanding of the distributed compliant mechanisms. The BCM method was further developed by the same author in Awtar and Sen, 2010 to analyze beams with variable cross-sections. Based on Awtar's work, Hao, Kong and Reuben, 2011 have extended the BCM method into 3D. In that work, a spatial three-beam model (see Fig. 2-8) was analyzed by nonlinear BCM method. The accuracy of the results in Hao, Kong and Reuben, 2011 was later verified by the spatial BCM of symmetrical beams developed by Sen and Awtar, 2013.
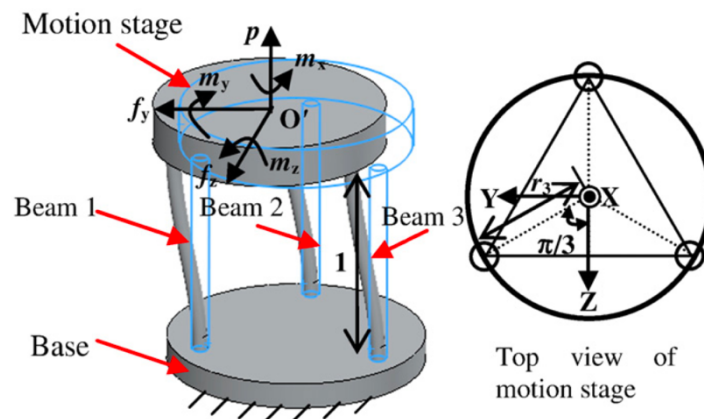

Figure 2-8: A spatial three-beam model (reproduced from Hao, Kong and Reuben, 2011).

2.1.3 Finite Element Method (FEM)

Another important method for analyzing compliant mechanisms is the finite element method. The basic idea of FEM is to divide a continuum structure into a plenty of discrete elements and a single element is described by a certain equation (see Fig. 2-9). The equations of the elements are then assembled into a larger system of algebraic equations. The FEM method is actually a numerical method for solving the governing partial differential equations over the modeling domain. Compared to the PRBM method, the FEM method is a high-fidelity method which presents the continuous details of the deformed part of compliant mechanisms. Originally, the FEM was developed for the analysis of statics problems in architectural and structural engineering since the linear elastic material theory was used in the modeling and it was valid for small-displacement mechanics problems (Turner *et al.*, 1956). In order to apply the FEM method in the analysis of compliant mechanisms, the linear theory was extended so that the large deflections could be taken into account (De Borst *et al.*, 2012).
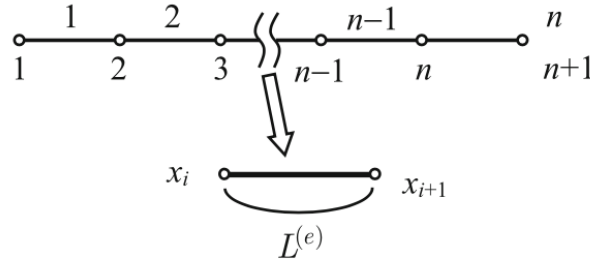
Figure 2-9: Finite-element discretization of a 1-D beam model.

According to the theory of continuum mechanics (Zienkiewics and Taylor, 2000), the strong form of the governing equations and boundary conditions of a linear elastic continuum body (with a spatial Lipschitz domain $\mathbf{\Omega}$) in static case can be formulated as:

Mass balance:	$\dot{\rho} = 0$
Momentum balance:	$\text{div}\,\boldsymbol{\sigma} + \mathbf{b_0} = 0$
Constitutive equation:	$\boldsymbol{\sigma} = \mathbf{E} \cdot \boldsymbol{\varepsilon}$
Linear strain tensor:	$\boldsymbol{\varepsilon} = \dfrac{1}{2}(\nabla\mathbf{u} + \nabla\mathbf{u^T})$
Neumann boundary condition:	$\boldsymbol{\sigma} \cdot \mathbf{n} = \mathbf{t_0} \quad \text{on } \mathbf{\Gamma_N}$
Dirichlet boundary condition:	$\mathbf{u} = \mathbf{u_0} \quad \text{on } \mathbf{\Gamma_D}$

$$(2\text{-}10)$$

Where ρ is the density, $\boldsymbol{\sigma}$ is the stress tensor, $\boldsymbol{\varepsilon}$ is the strain tensor, $\mathbf{u}$ is the displacement, $\mathbf{b_0}$ is the body force, $\mathbf{E}$ is the elasticity tensor, $\mathbf{\Gamma_N}$ and $\mathbf{\Gamma_D}$ are the Neumann and Dirichlet boundaries, $\mathbf{t_0}$ and $\mathbf{u_0}$ are the predefined load and displacement in the boundary conditions, $\mathbf{n}$ is the normal vector of the Neumann boundary. A requirement of the strong form is that the solutions must be differentiable in the entire domain $\mathbf{\Omega}$, which is sometimes too strict for real problems. Therefore, the weak or integral form in Equation (2-11) is used in the FEA to ease the calculations. The weak form can be derived from the principle of virtual work, in which a virtual

displacement field $\mathbf{v}$ is defined. This $\mathbf{v}$ can be visualized as an extremely small change in the real displacement field u, which doesn't change the potential energy of the entire system of the solid.

$$\int_{\Omega} \mathbf{E}\boldsymbol{\varepsilon}(\mathbf{u}) : \boldsymbol{\varepsilon}(\mathbf{v}) \cdot d\Omega = \int_{\Omega} \mathbf{b_0}\mathbf{v} \cdot d\Omega + \int_{\Omega} \mathbf{t_0}\mathbf{v} \cdot d\mathbf{A}, \quad \mathbf{v} = \mathbf{0} \text{ on } \mathbf{\Gamma_D} \tag{2-11}$$

$$\mathbf{u_h(x)} = \sum_{i=1}^{M} U_i \mathbf{N_i(x)} \qquad \mathbf{v_h(x)} = \sum_{j=1}^{M} V_j \mathbf{N_j(x)} \tag{2-12}$$

$$V_j \sum_{i=1}^{M} \left(\left(\int_{\Omega} \mathbf{E}\boldsymbol{\varepsilon}(\mathbf{u}) : \boldsymbol{\varepsilon}(\mathbf{v}) \cdot d\Omega \right) U_i \right) = V_j \left(\int_{\Omega} \mathbf{b_0}\mathbf{N_j(x)} \cdot d\Omega + \int_{\Omega} \mathbf{t_0}\mathbf{N_j(x)} \cdot d\mathbf{A} \right) \tag{2-13}$$

$$V_j \left(\sum_{i=1}^{M} \mathbf{K_{ij}} U_i - \mathbf{F_j} \right) = 0 \quad \Rightarrow \quad \mathbf{KU = F} \tag{2-14}$$

After performing finite element discretization on the solid, the displacement field $\mathbf{u}$ and the continuous form of the virtual work equation in Equation (2-11) can be transformed into a discrete form as in Equation (2-12) and (2-13), where $\mathbf{u_h}$ and $\mathbf{v_h}$ are the approximated real and virtual displacement field, M is the degree of freedom of the displacement vector $U = [U_1 \ U_2 \ ... \ U_M]^T$ of the discrete space, $\{N_i(x)\}_{i=1,2,...,M}$ is the basis of the discrete space for interpolation. It is supposed that the approximated displacement field uh is interpolated by the displacement vector $\mathbf{U}$. The Equation (2-13) can be rewritten in the short form as in Equation (2-14), where $\mathbf{K_{ij}}$ is the (i,j)-th entry of the so-called global stiffness matrix $\mathbf{K}$, $\mathbf{F_j}$ is the j-th entry of the right-hand load vector $\mathbf{F}$. The displacement vector $\mathbf{U}$ can be calculated by solving the system of linear equations $\mathbf{KU = F}$ after applying the boundary conditions.

As is already mentioned before, although the linear model takes less time for calculation, it cannot be directly used for the modeling of compliant mechanisms due to their large-displacement feature. As can be seen in Fig. 2-10, when a bending moment M is applied on a flexible straight beam, the tip displacement direction of the linear FE-model (see Fig. 2-10b)) is vertically upwards. However, in the real situation (see Fig. 2-10a)), the total length of the beam is

Real beam   Linear beam
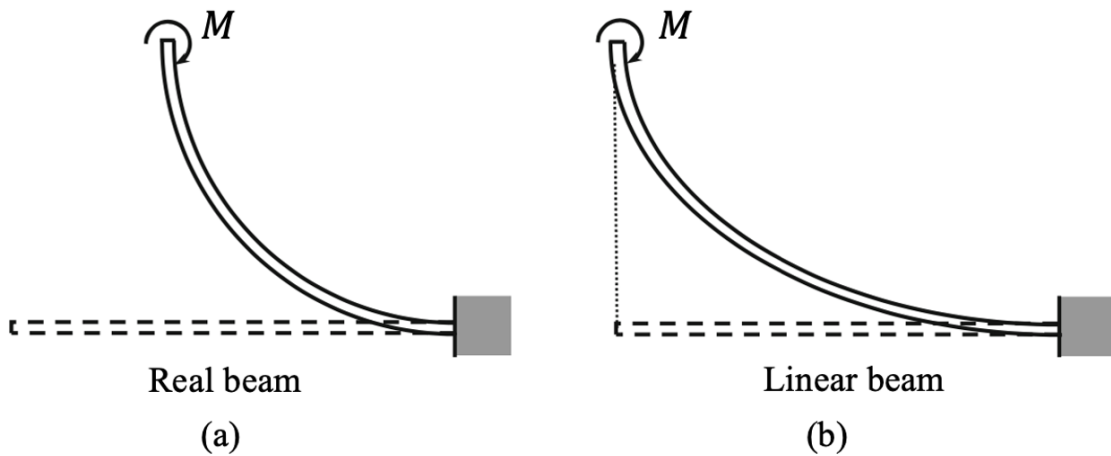(a)   (b)

Figure 2-10: Graphical illustration of the modeling error of the linear beam model (reproduced from Kim, 2014): a) The deformation of a real beam under a bending moment, b) The deformation of a linear beam FE-model.

constant and hence, the tip should be pulled back with a certain distance. This phenomenon arises from the one-step calculation of the linear model. To cope with this problem, the governing equations of the linear model were modified to incorporate the geometrical non-linearity. As the linear strain tensor in Equation (2-10) is a first-order approximation of the real strain tensor, the following modification is made to take the geometrical non-linearity into account:

$$\varepsilon(x) = \frac{du(x)}{dx} \quad \rightarrow \quad \varepsilon(x) = \frac{du(x)}{dx} + \frac{1}{2}\left(\frac{du(x)}{dx}\right)^2 \qquad (2\text{-}15)$$

where the one-dimensional case was used for illustration. Then, instead of solving the global equation system (Equation (2-14)) in one step, iterative processes were developed to calculate the displacement vector **U**. Some popular methods include displacement-based incremental method and load-based incremental method (De Borst *et al.*, 2012; Kim, 2014).

Since the FEM theory has been developed for many years, many commercial software are already available to perform robust non-linear FE-analysis of compliant mechanisms, such as Abaqus (Dassault Systemes, France), ANSYS (Ansys, USA) and Solidworks (Dassault Systemes, France). Fig. 2-11a) and Fig. 2-11b) show the FEM modeling examples using ANSYS and Abaqus respectively. These software are usually black box, which means the modeling functions cannot be modified by the user. However, there are also some open-source platforms for researchers to develop FEM algorithms themselves, such as Python and MATLAB. Fig. 2-11c) shows a modeling example of flexible beam using MATLAB.
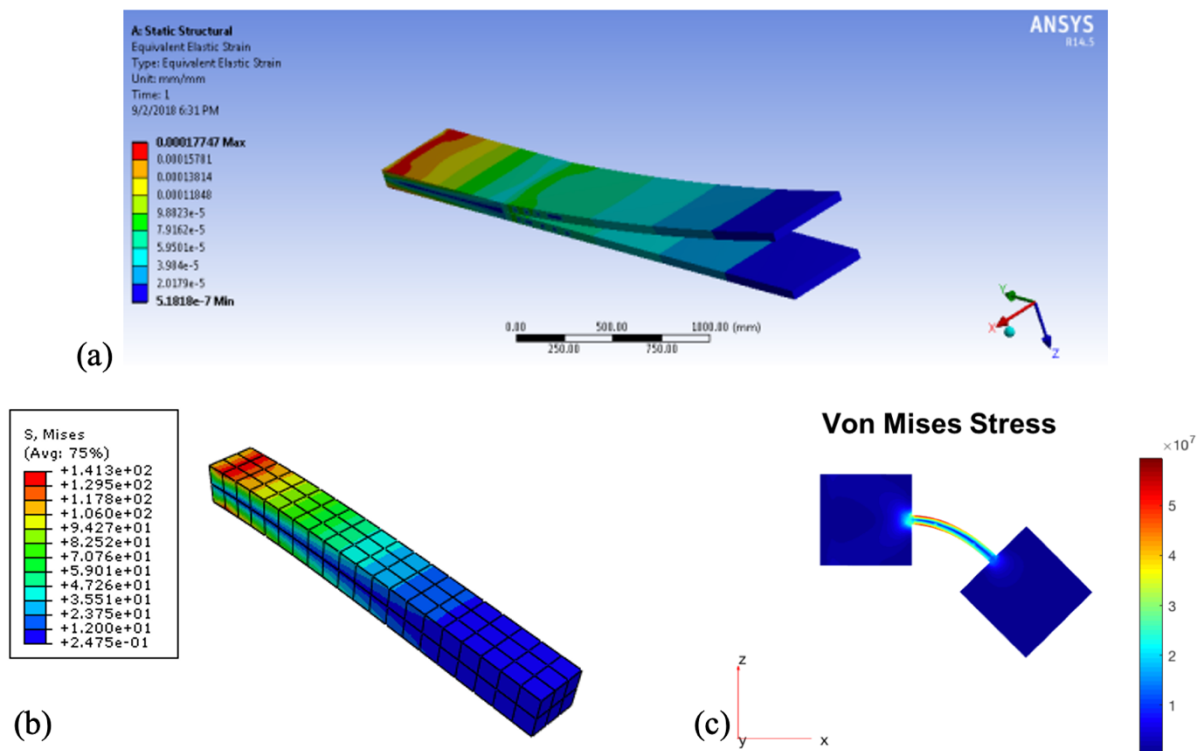
Figure 2-11: FEM modeling using different platforms: a) Modeling using ANSYS, b) Modeling using Abaqus, c) Modeling using MATLAB (reproduced from (Sun, Liu and Lueth, 2019)).

2.2 Synthesis Methods for Compliant Mechanisms

As is mentioned in Section 1.2, the mechanism design has two tasks: mechanism modeling and mechanism synthesis. Since the first task is already discussed in Section 2.1, this section will focus on the second task. Generally, there are three types of synthesis methods for compliant mechanisms: rigid-body replacement synthesis, building blocks methods and structural optimization methods.

2.2.1 Rigid-Body Replacement Synthesis

The rigid-body replacement synthesis is based on the PRBM method, which is already described in Section 2.1.1. The basic workflow of this synthesis method is shown in Fig. 2-12. Firstly, the topology of a rigid-body mechanism (see Fig. 2-13a)) is predefined which can achieve specific functions. Secondly, the rigid links are replaced by torsional springs to build a pseudo-rigid-body mechanism. Then, the created PRB model is analyzed using the modeling methods introduced in Section 2.1.1 to check if the design requirements are fulfilled. If not, the parameter of the PRB model will be iteratively modified. Once an appropriate PRB model is

Figure 2-12: Basic workflow of the rigid-body replacement synthesis (reproduced from Howell and Midha, 1994).
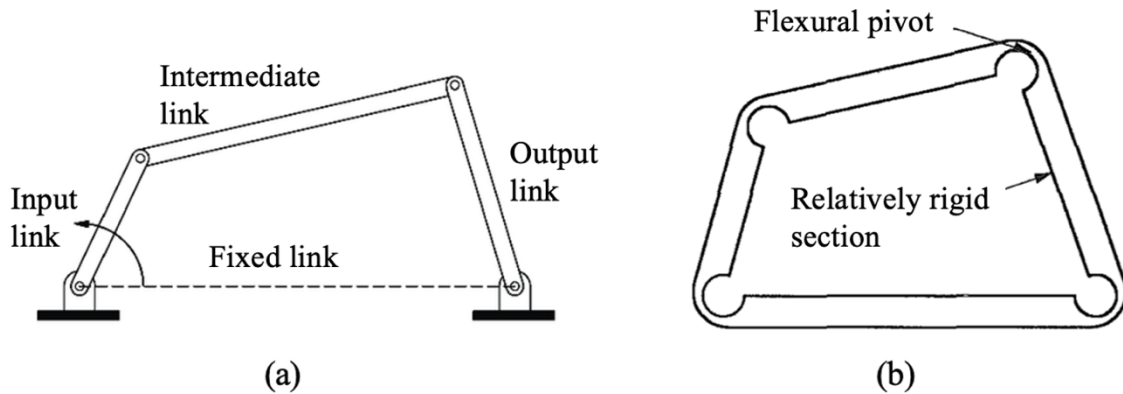
Figure 2-13: Rigid-body mechanism (a) and fully compliant mechanism (b) (reproduced from Howell and Midha, 1994).

found, it will be transferred to a fully compliant mechanism (see Fig. 2-13b)). A second round of model verification is the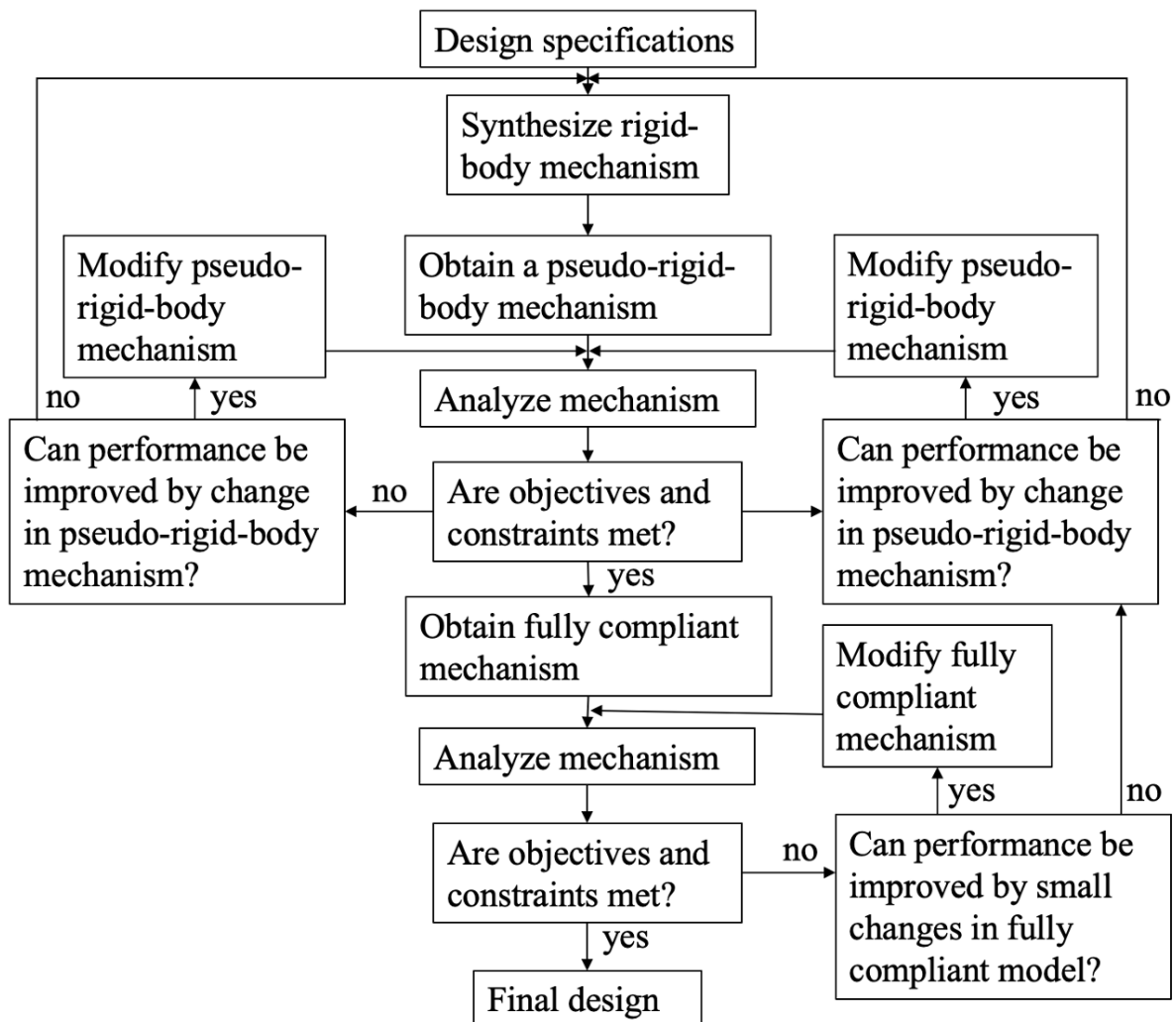n carried out to make the final compliant mechanism achieves the design objective. Typical applications of the rigid-body replacement synthesis can be found in (Murphy, Midha and Howell, 1996; Pucheta and Cardona, 2010).

2.2.2 Building Blocks Methods

The basic idea of the building blocks methods is to concatenate a set of basic compliant mechanisms to create a compliant mechanism which can achieve complex tasks. There are basically two kinds of building blocks methods, which are instant center based building blocks method and flexible building blocks method.

Instant Center Based Building Blocks Method

The instant center based building blocks method was firstly proposed in Kim, Kota and Moon, 2005. The method is a conceptual design procedure and aims to find a mechanism that generates an output displacement with desired motion direction and geometric advantage (GA) for a pre-defined input (see Fig. 2-14). For the synthesis, two kinds of building blocks are used as basic elements to generate the entire compliant mechanism, which are the compliant dyad building block (CDB) and the compliant four-bar building block (C4B). Fig. 2-15a) and Fig. 2-15b) illustrate the two building blocks graphically. Fig. 2-15c) and Fig. 2-15d) show two design examples that are created from the two building blocks.
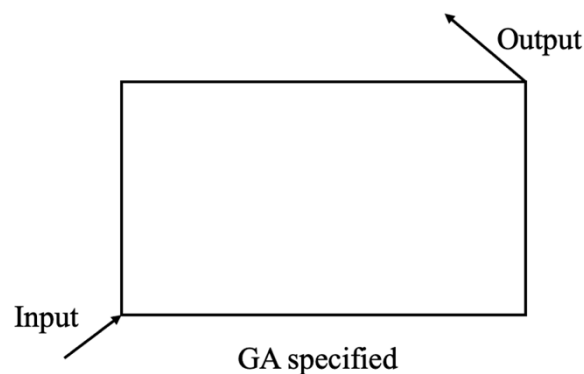
Figure 2-14: Single-input-single-output design problem for instant center based building blocks method (reproduced from Kim, Kota and Moon, 2005).

15
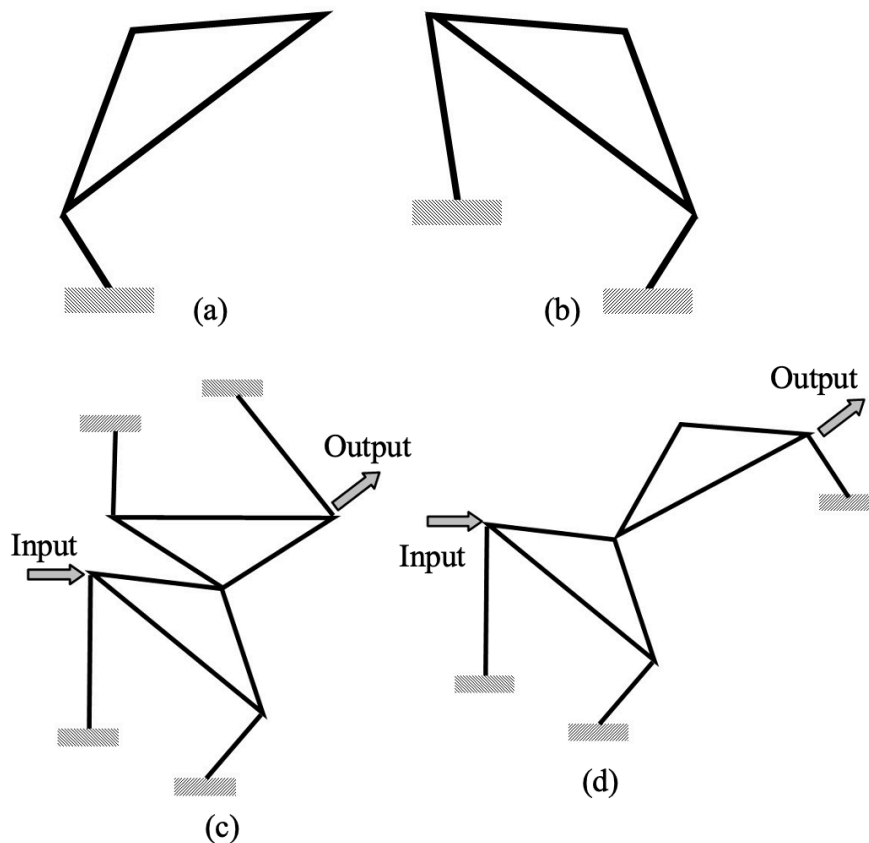
Figure 2-15: Two basic building blocks und their combinations (reproduced from Kim, Kota and Moon, 2005): a) The compliant dyad building block (CDB), b) the compliant four-bar building block (C4B), c) Combination of two C4B, d) combination of C4B and CDB.

In order to realize the mechanism synthesis, a dual stage synthesis process was proposed in Kim, Kota and Moon, 2005, which involves the concatenation of any two building blocks. Herein, three concepts are introduced, which are the principal compliance vector (PCV), the instant center and the decomposition point. The PCV is a vector at the input or output port that shows the direction of the major compliance (see Fig. 2-16a)). The instant center is defined as the instant rotating center of the rigid body, while the decomposition point is connection point of the two chosen building blocks (see Fig. 2-16b)).
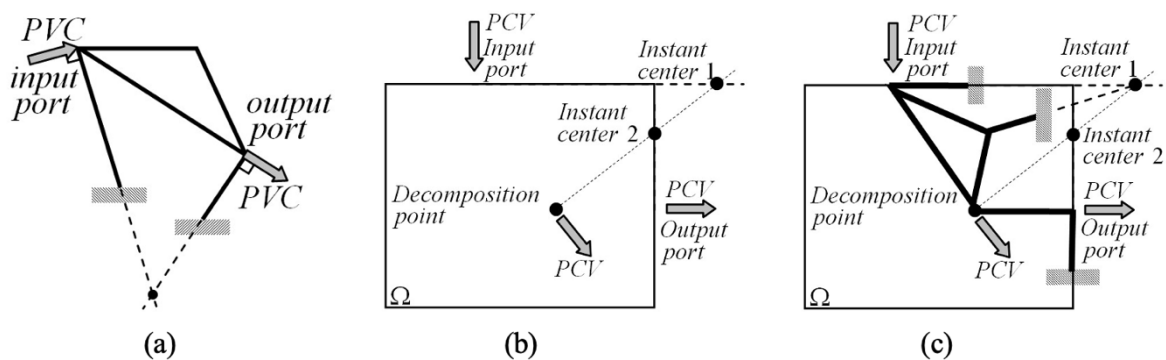
Figure 2-16: Illustration of the PCV, instant center and decomposition point as well as a synthesis example (reproduced from (Gallego and Herder, 2009)): a) Instant center of a C4B, b) A design problem, c) Final design using a C4B and a CDB.
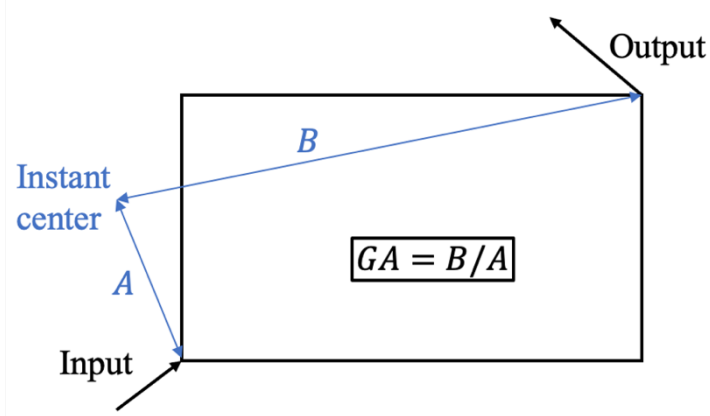
Figure 2-17: The definition of the geometric advantage (reproduced from Kim, Kota and Moon, 2005).

The first step of the dual stage synthesis process is to find an appropriate decomposition point, where the direction of PCV ensures the prescribed total geometric advantage (GA). The GA can be calculated as the ratio B over A (see Fig. 12-7), where A and B are the length from the input port to the instant center and the length from the output port to the instant center, respectively. The total GA is defined as follows:

$$GA_{total} = GA_1 \cdot GA_2 \qquad (2\text{-}16)$$

where GA_1 and GA_2 are the GA of the first and second building block, respectively. In order to find the optimal decomposition point, the geometric advantage index n_{GA} is introduced as the fitness function:

$$n_{GA} = \log_{GA_{target}} GA_2 \qquad (2\text{-}17)$$

where the ideal n_{GA} is 0.5, which means the two building blocks have the same GA.

The second step of the dual stage synthesis process is the selection of the moving junctions. The moving junctions connect the floating links to the ground, which are similar to the flexible beams in the PRB model. In order to provide a criterium for the selection, the geometric advantage error is introduced:

$$GA_{error} = \frac{GA_{FEA} - GA_{target}}{GA_{target}} \qquad (2\text{-}18)$$

where GA_{FEA} and GA_{target} are the GA calculated by FEA and the target GA, respectively. The aim is to find a building block geometry with GA_{error} close to 0. In this way, the balance between the relative motion of the two building blocks and the effective motion transmission can be obtained.

Flexible Building Blocks Method

The basic idea of the flexible building blocks method is to assemble a certain number of basic flexible building blocks to create a total compliant mechanism for achieving specific motion tasks. The flexible building blocks method was firstly proposed in Bernardoni *et al.*, 2004. The
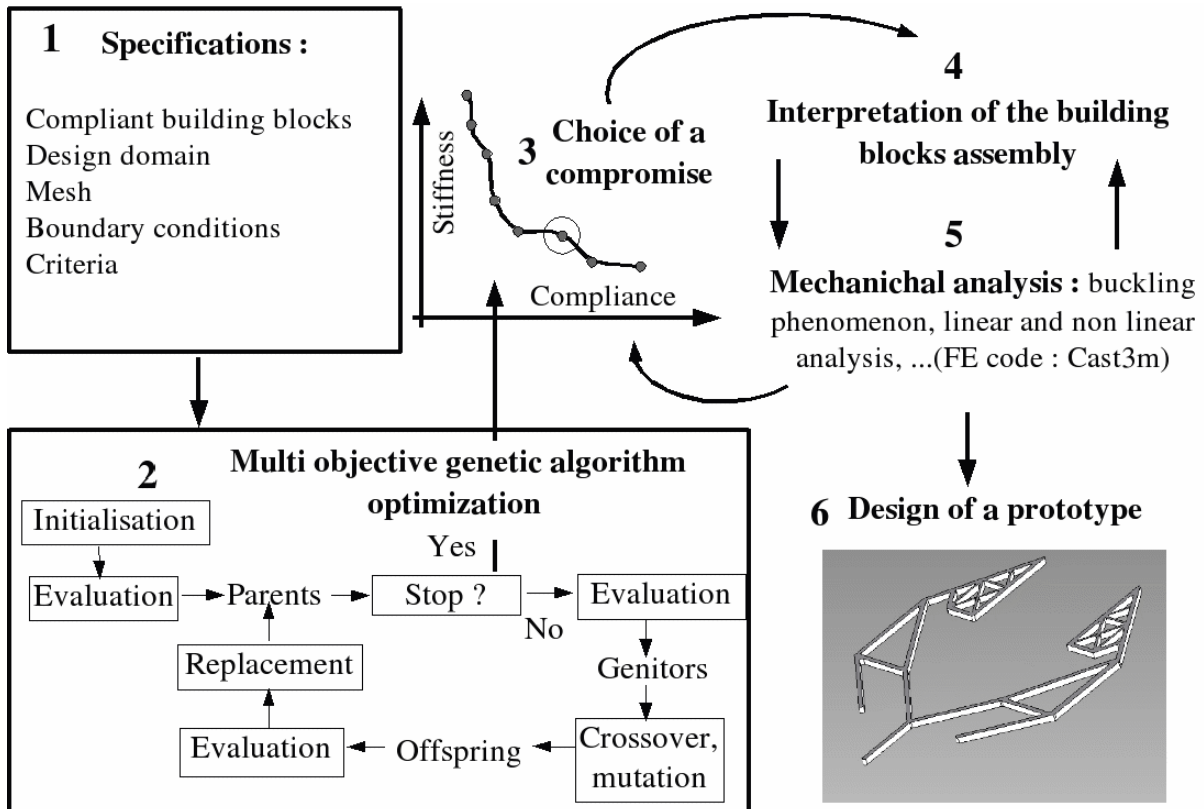
Figure 2-18: Overview of the flexible building blocks method (reproduced from Bernardoni et al., 2004).

workflow of the method is shown in Fig. 2-18 and the essence is to find an optimal distribution of the basic flexible building blocks in a meshed design domain. An overview of the basic flexible building blocks is provided in Fig. 2-19 and these basic blocks are joined through the nodes on their edges inside a mesh to form a total compliant mechanism. As each basic building block has its own stiffness matrix, the global stiffness matrix of the formed compliant mechanism can be calculated by assembling the stiffness matrices of all the basic building blocks. To solve the synthesis problem, a multi-objective generic algorithm is introduced to search for the
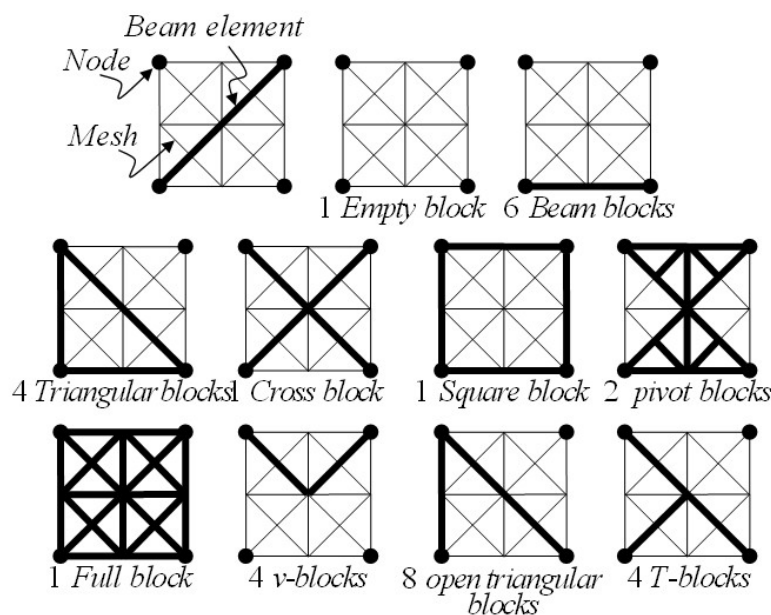
Figure 2-19: Overview of the basic flexible building blocks (reproduced from Bernardoni et al., 2004).
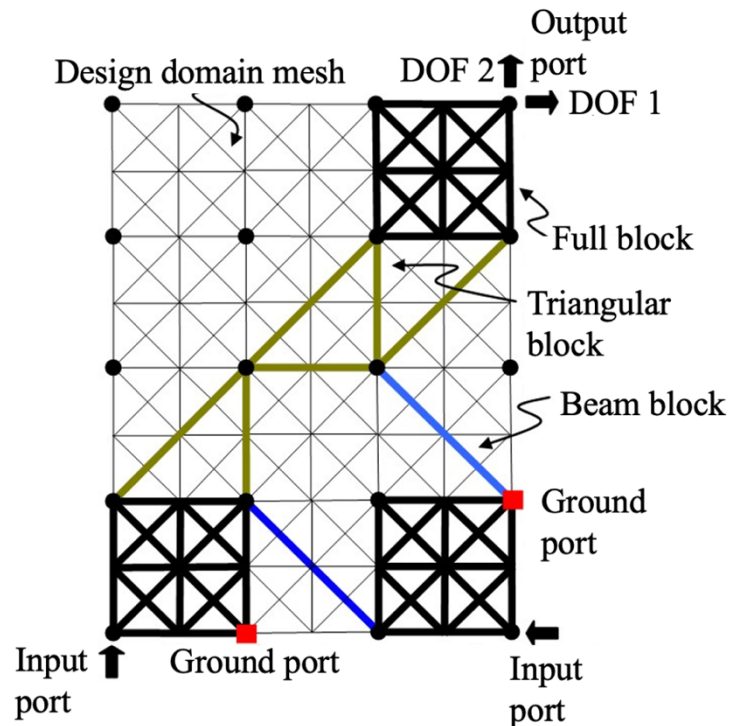
Figure 2-20: Application of the flexible building blocks method to the design of a compliant mechanism (reproduced from Bernardoni et al., 2004).

optimal distribution of the basic building blocks. The solution is obtained by optimizing the balance between the following parameter pairs: compliance and stiffness, force and displacement. Different design objectives, such as the mutual potential energy, GA, mechanical advantage (MA), are used for optimization. A synthesis example is presented in Fig. 2-20 to show the design effect of the flexible building blocks method.

2.2.3 Structural Optimization Methods

The basic idea of the structural optimization methods is to find an optimal compliant mechanism that fulfills an objective function using optimization techniques. Generally, there are three main types of structural optimization methods, which are size optimization, shape optimization and topology optimization. The difference of the three optimization methods is graphically
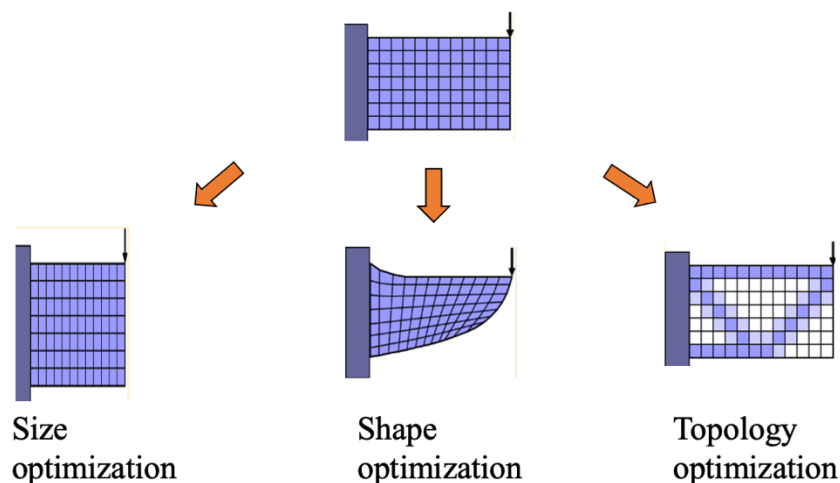
Figure 2-21: Graphical illustration of the three types of structural optimization.

19

illustrated in Fig. 2-21. Among the three methods, the size optimization is at the lowest level since it modifies only the major parameters of the mechanism, such as the thickness of the flexible beam or the length of the rigid segments. A step further is the shape optimization which searches for the optimal contour or surface of the compliant mechanism to achieve prescribed task while the mechanism topology is predefined and unchanged. The highest level is the topology optimization, where the material connectivity among the entire design domain are determined. In this section, the latter two methods, the shape and topology optimization methods, are discussed with respect to their applications in the synthesis of compliant mechanisms.

Geometric Model Based Shape Optimization Methods

Generally, an optimization problem for designing compliant mechanisms can be formulated as follows:

$$\max_{x}: \quad f(\mathbf{x})$$

$$\text{subject to:} \quad h_i(\mathbf{x}) = 0, \quad i = 1, 2, \dots, p$$

$$: \quad g_j(\mathbf{x}) \leq 0, \quad j = 1, 2, \dots, m \qquad (2\text{-}19)$$

$$: \quad \mathbf{x} = [x_1 \ x_2 \ \dots \ x_n]$$

where it is assumed that maximizing the objective function $f(\mathbf{x})$ can achieve the optimal design of the compliant mechanism and $\mathbf{x}$ are the design variables. $h_i(x) = 0$ and $g_j(x) \leq 0$ are the design conditions and design constraints, respectively. Originally, there are two options to select design variables $\mathbf{x}$ for the general shape optimization, which are based on FE models and geometric models, respectively. However, in the current state of the art, the geometric models are mostly used in the shape optimization for compliant mechanisms. Typical geometrical models are, for example, splines and Bezier curves.

The authors in Vehar and Kota, 2006; Jutte and Kota, 2007 proposed a shape-optimization-based method to synthesize compliant mechanisms with a prescribed non-linear load-displacement function. In that method, the basic branch-like topology of the compliant mechanism is predefined and cubic B-splines with 5 control points are used as the geometric model to characterize the shape of each branch (see Fig. 2-23a)). The objective function is to minimize the error of the load-displacement curve of the design proposal relative to a predefined target curve (see Fig. 2-22). To obtain the load-displacement curve, the commercial FEM modeling Abaqus
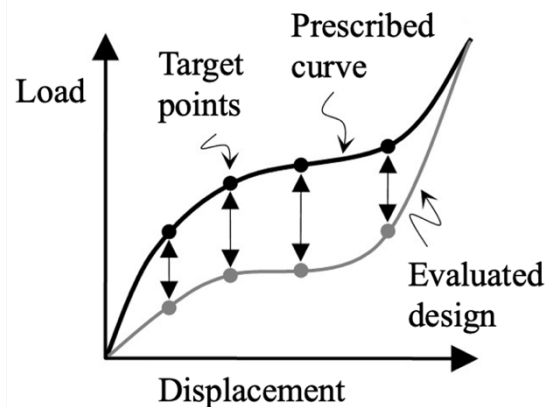
Figure 2-22: Error between the load-displacement curve of the design proposal and a predefined target curve (taken from Vehar and Kota, 2006).
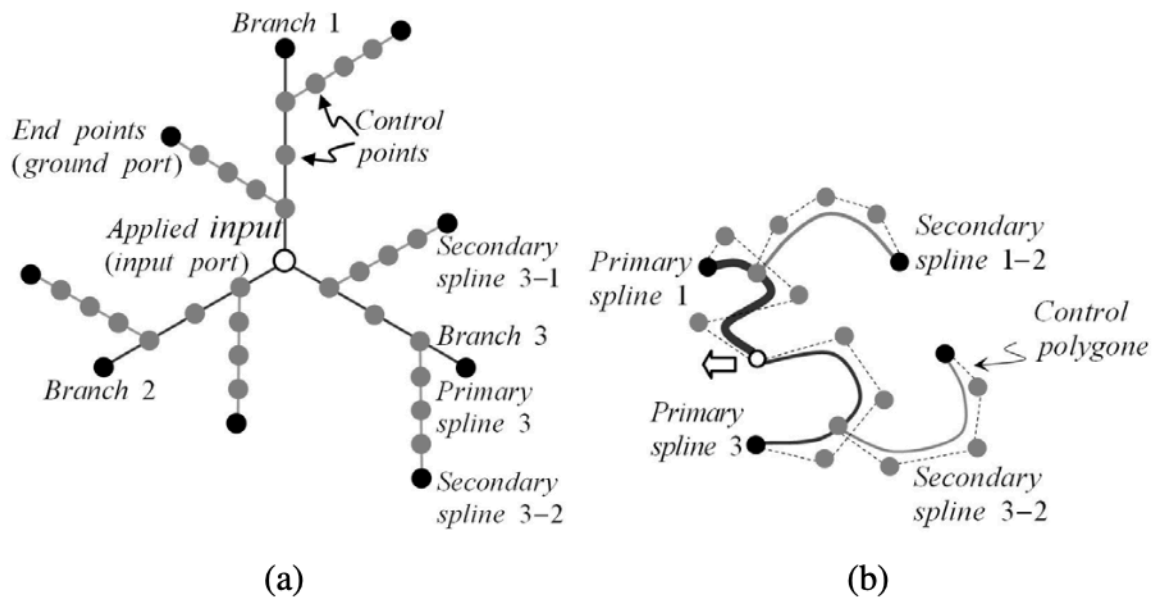
Figure 2-23: Graphical illustration of a spine-based shape optimization method (reproduced from Vehar and Kota, 2006): a) The basic branch-like topology of the compliant mechanism, b) A synthesis example.

is used using the beam element model B21H. In the modification of the design variables (5 control points of a spline), a generic algorithm is employed. A synthesis example is presented in Fig. 2-23b).

A typical Bezier-curve-based shape optimization method is proposed in Xu and Ananthasuresh, 2003. Similar to the spline-based method, the topology of the final design is predefined, and the Bezier curve is used as the basic geometric model with two control points. Fig. 2-24 shows a design example the Bezier-curve-based method, which is to synthesize a symmetric compliant gripper. The objective function is to maximize the mutual strain energy (MSE) which is analogous to the displacement d at the output port. Due to the symmetry of the gripper mechanism, only a half gripper is included in the design domain and the symmetric line is modeled as a
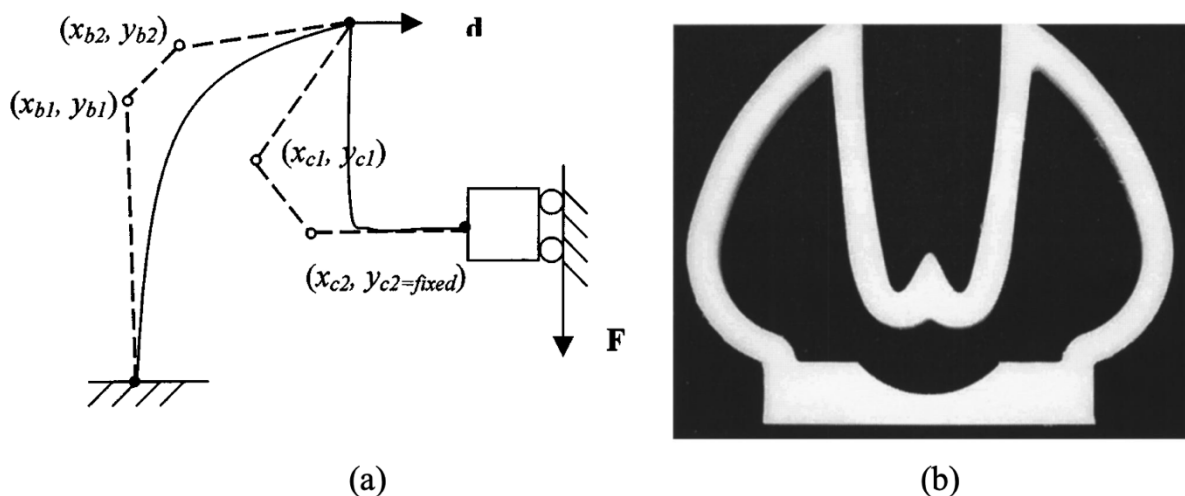
Figure 2-24: A gripper design example of a Bezier-curve-based shape optimization method (reproduced from Xu and Ananthasuresh, 2003): a) Schematic representation of the design problem, b) Final design of the gripper.

21

sliding mechanism. The initial topology of the half gripper contains two Bezier curves and the control points of each curve are $(x_{ci}, y_{ci})_{i=1,2}$ in the Fig. 2-24a). A force F is applied along the symmetric line to actuate the gripper. To solve the optimization problem, the sequential quadratic programming algorithm implemented in MATLAB is used. The final design is shown in Fig. 2-24b). Further development of the Bezier-curve-based shape optimization methods includes the work of Zhou and Ting, 2005. In that work, the wide Bezier curve is used as the geometric model, which can be considered as a Bezier curve with variable cross-section area.

FEM-Based Topology Optimization Methods

As mentioned before, the topology optimization methods consider the entire geometry of the initial design as design domain, also including the interior material. Therefore, from the modeling point of view, the FEM is currently the mostly used modeling and analysis method in topology optimization as it can basically handle the modeling of any complex geometries. This section is thus focused on the discussion of FEM-based topology optimization methods for compliant mechanisms.

The first work that introduced the FEM-based topology optimization method to the design of compliant mechanisms is Sigmund, 1997. A typical design problem is presented in Fig. 2-25, where the compliant mechanism is supposed to be actuated by an input force f_{in} and have a target motion u_{out}. The linear springs k_{in} and k_{out} can be treated as input and output resistance respectively. In order to maximize the target motion u_{out}, the available material in the design domain should be distributed in the most efficient way. The corresponding design problem can be mathematically formulated as:

$$\max_{\boldsymbol{\rho}}: \quad u_{out}$$

$$\text{subject to:} \quad \mathbf{KU} = \mathbf{F}$$

$$: \quad \sum_{e=1}^{N} v_e \rho_e \leq V, \quad 0 < \rho_{min} \leq \rho_e \leq 1$$

(2-20)

where $\mathbf{K}$ and $\mathbf{U}$, according to the FEM theory described in Section 2.1.3, are the global stiffness matrix and displacement vector respectively. $\mathbf{F}$ is the load vector. Since the design variable $\boldsymbol{\rho}$ (density) is not directly included in the FEM formulation, a relationship between Young's
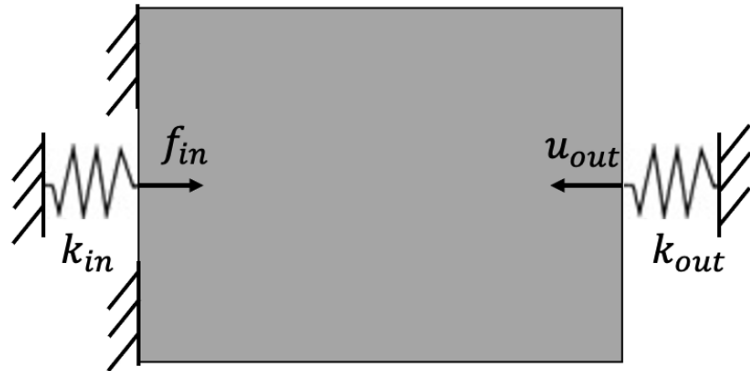
Figure 2-25: A typical compliant mechanism design problem using topology optimization (reproduced from Bendsoe and Sigmund, 2003).

modulus and the density is modeled by the Solid Isotropic Material with Penetration (SIMP) method (Bendsøe, 1989):

$$E_e = E_0 \rho_e^p \qquad (2\text{-}21)$$

The design problem in Equation (2-20) is solved by the Sequential Linear Programming (SLP) method (Sigmund, 1997).

Generally, the FEM-based topology optimization methods can be divided into two categories in terms of the element types, which are the discrete ground structure based methods and the continuum structure based methods. The work in Sigmund, 1997 is based on continuum structure. The applications of these two kinds of methods to the design of compliant mechanisms are illustrated and compared in Frecker *et al.*, 1997. A design example of a gripper mechanism is shown in Fig. 2-26, where both methods are used to synthesize the mechanism. Using the ground structure based method, each node in the design domain is connected with all the other nodes via frame-like elements (see Fig. 2-26b)), which is different from the continuum mesh presented in Fig. 2-26e). The corresponding design results as well as FEM-based motion simulations are also presented in Fig. 2-26. Since the basic idea of the FEM-based topology optimization is similar to the evolutionary process in nature and the realized continuum structure also has a bionic shape, we can place it into bionic design methods.
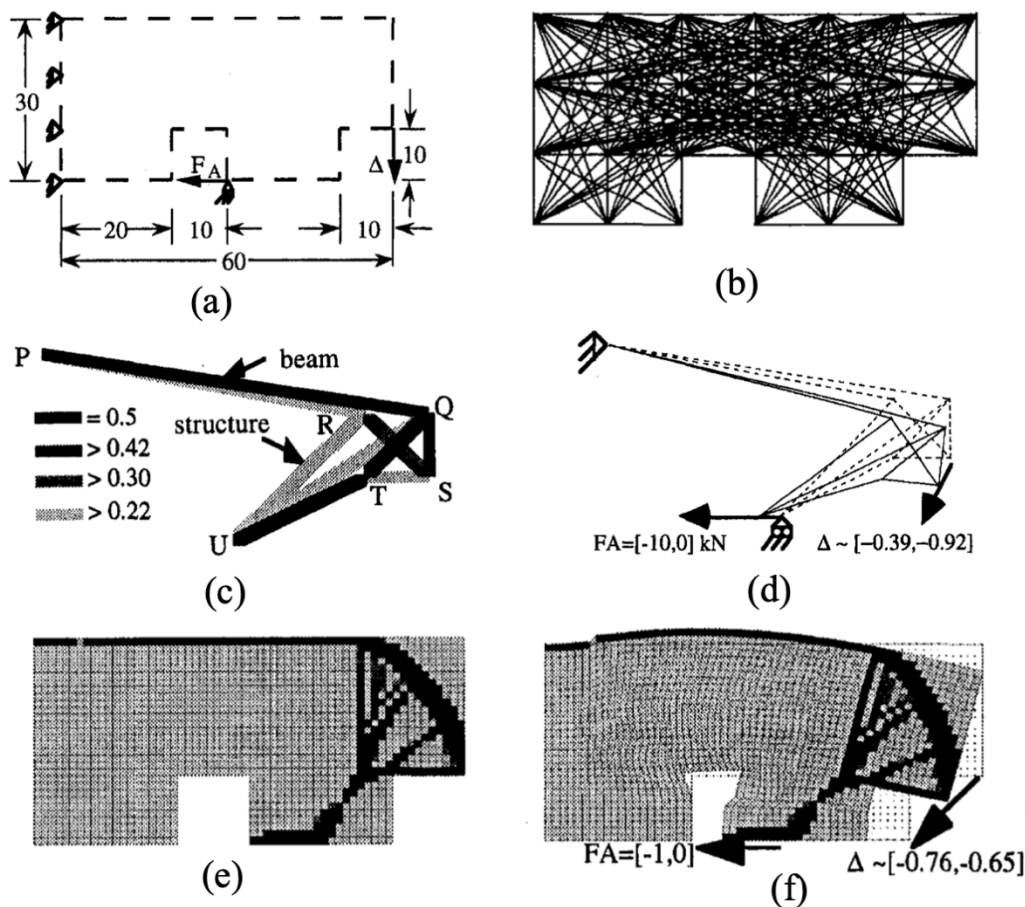
Figure 2-26: Design of a gripper mechanism using ground structure based and continuum structure based topology optimization methods (reproduced from Frecker et al., 1997): a) The design problem, b) Fully ground structure, c) Design result of the ground structure based method, d) FE simulation of the first design result, e) Design result of the continuum structure based method, f) FE simulation of the second design result.

23

On the other hand, although the design examples in Fig. 2-26 are 2D, some 3D topology optimized design results are also proposed in Liu and Tovar, 2014.

Open-Source Codes for FEM-Based Topology Optimization Methods

Currently, a lot of open-source codes have been developed for FEM-based topology optimization, which is very useful to the engineers and researchers because the commercial FEM codes are usually black box and cannot be further developed by the user. The very first open-source topology optimization code was the 99-line MATLAB code provided by Sigmund, 2001. Although that code was focused on the minimum compliance design, it provided the possibility to be modified to adapt to the application of compliant mechanisms. A following study in Andreassen *et al.*, 2011 has provided an 88-line MATLAB codes to improve the performance of the 99-line code. However, the above codes utilize the square element which is difficult for the meshing of a geometry with an arbitrary shape. To cope with this problem, Talischi *et al.*, 2012 developed the PolyTop code in MATLAB to incorporate unstructured polygonal finite element into topology optimization. An extension of PolyTop for the design of compliant mechanisms can be found in (Pereira *et al.*, 2011). In order to cover the 3D design task, Liu and Tovar, 2014 has presented a 169-line MATLAB code for topology optimization, in which the synthesis of compliant mechanisms was also included. A similar 3D topology optimization code was provided by Zuo and Xie, 2015, which was implemented in the Python environment.

2.3 Current Applications to Compliant Medical Instruments

The modeling and synthesis methods for general-use compliant mechanisms have already been developed for many years. However, their medical applications are still rare. In this section, we will list the current applications of the described design methods for compliant medical instruments.

2.3.1 Modeling of Continuum Surgical Robots

An important medical application of compliant mechanisms are the continuum surgical robots. Unlike the traditional robots whose rigid arms are connect by revolute joints, flexible beams are widely used in the continuum robots to realize dexterous robotic motions.

Piecewise Constant Curvature Model (PCCM)

A popular method for the kinematic modeling of continuum surgical robots is the piecewise constant curvature model. This method was originally developed for modeling tendon-driven continuum robots (Gravagne and Walker, 2000; Gravagne, Rahn and Walker, 2003; Jones and Walker, 2006), but was then also extended for the application of concentric tube robots (Webster et al., 2009). The basic idea of the PCCM method is to simplify each bent section of a continuum robot as an arc with constant curvature. For the tendon-driven continuum surgical robots (see Fig. 2-27a)), the constant-curvature approximation can be graphically illustrated in Fig. 2-27b) (Camarillo *et al.*, 2008). The single section is treated as a cantilever beam, which is deflected into a circular arc by the external load of a pulling tendon. The deflected beam is comprised of infinite concentric arcs, and one of these circular arcs can be described as:

$$\phi(s) = \left(\frac{1}{R_c + x}\right) s \qquad (2\text{-}22)$$

(a)   (b)
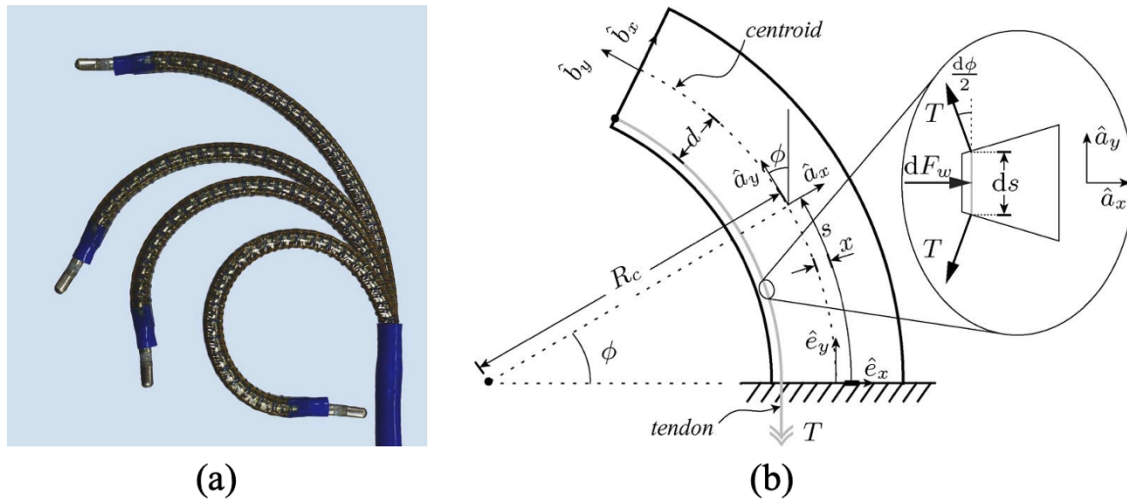
Figure 2-27: Kinematic modeling of tendon-driven continuum manipulators using PCCM method (reproduced from Camarillo et al., 2008): a) A steerable cardiac catheter (tendon-driven continuum manipulator), b) Constant-curvature approximation of a single-tendon case.

where s and $\phi(s)$ are the arc length and the corresponding deflection angle, respectively. R_c and x represent the radius of the centroidal arc and the distance between the centroid and the measured arc, respectively. The curvature κ of the measured arc is determined as:

$$\kappa(x) = \frac{d\phi}{ds} = \frac{1}{R_c + x} \tag{2-23}$$

The following mechanical modeling is all based on this constant-curvature assumption and can be found in Camarillo *et al.*, 2008. From the modeling point of view, the PCCM method is similar to the PRBM method since they are both based on the simplification of the bending flexible beam.
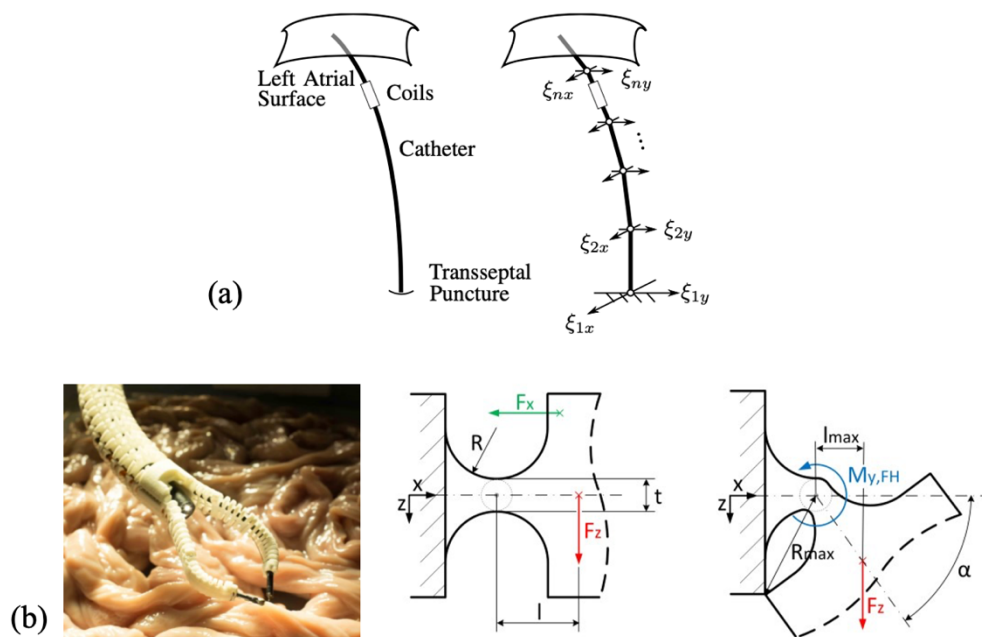
Figure 2-28: Modeling of continuum surgical robots using PRBM method: a) PRB model of an MRI-actuated catheter (taken from Greigarn and Cavusoglu, 2015), b) Modeling the flexure hinges of a snake-like surgical robot using PRB model (taken from Krieger et al., 2017).

PRB Modeling

As the motions of the continuum surgical robots are mostly realized by flexible beams, the PRBM method is frequently used to simplify the modeling process. A typical PRBM-based modeling example is provided by Greigarn and Çavuşoğlu, 2015, where an MRI-actuated catheter was analyzed (see Fig. 2-28a)). The flexible catheter is simplified as a combination of a series of torsional springs and rigid part, while each joint has two bending DOFs. Since the catheter is actuated by the Lorenz force from the magnetic moment of the coils and the MRI's magnetic field, the actuation load is modeled as joint torques applying on each connecting joint. In this way, the catheter system can be modeled using the PRBM theory introduced in Section 2.1.1.

Another example is the modeling of the flexure hinges in a multi-arm snake-like surgical robot (Krieger *et al.*, 2017). The proposed continuum surgical robot is tendon-driven and utilizes small-length flexure hinges to realize dexterous motions (see Fig. 2-28b)). Using the PRBM theory to model the flexure hinge presented in Fig. 2-28b), the stiffness k of the simplified torsional spring can be calculated as:

$$k = \frac{M}{\alpha} = \frac{2Ewt^{2.5}}{9\pi\sqrt{R}} \tag{2-24}$$

where E and t are the Young's modulus and thickness of the flexure hinge, respectively. w and R are the width and radius of the flexure hinge. Besides, experiments were also carried out in Krieger *et al.*, 2017 to evaluate the fatigue strength of the flexure hinges, which verified the feasibility of the proposed PRBM model.

FEM-Based Modeling

As many continuum surgical robots have very complex geometries, the FEM method is used to realize high-fidelity modeling. A typical FEM-based modeling example is presented in Hu *et al.*, 2019. As can be seen in Fig. 2-29, although this snake-like surgical is also tendon-driven as in Fig. 2-27 and Fig. 2-28a), the complex helical structure makes it difficult to be simplified by the PCCM and PRBM method. Therefore, the authors used the commercial software Solidworks to analyze the continuum robotic structure. In the modeling, load applied to the tendon-driven section was simplified as a force couple on the top of the section. Fig. 2-29b) shows the
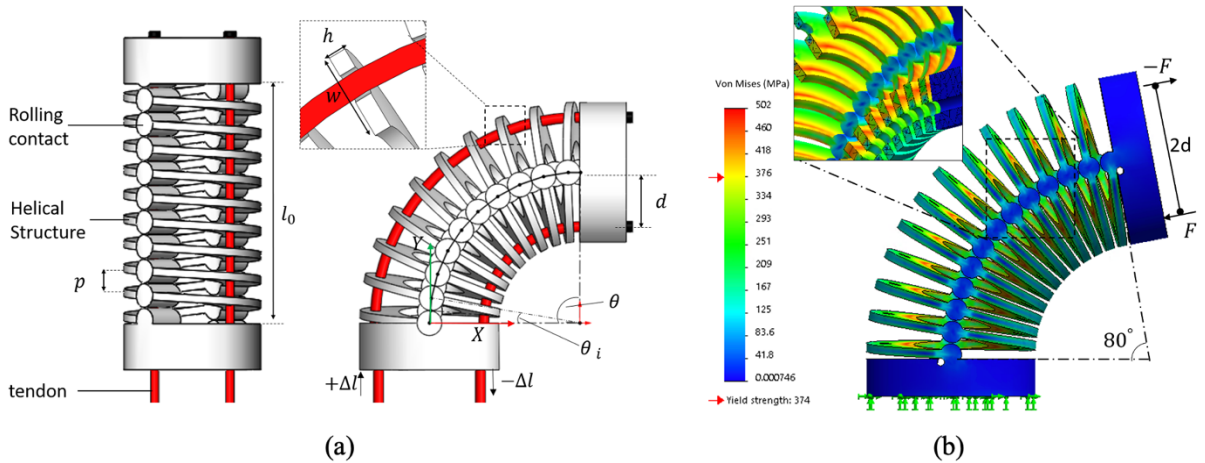
Figure 2-29: FEM-based modeling of a snake-like surgical robot (reproduced from Hu et al., 2019): a) Continuum robot with helical structure and rolling contact, b) FE model of the snake-like robot.

stress distribution of the surgical robot after being actuated by the tendons. Other FEM-based modeling examples of continuum robots can be found in Baek, Yoon and Kim, 2016; Bieze *et al.*, 2018.

2.3.2 Design of Compliant Surgical Forceps

Surgical forceps are an important for grasping, retracting and stabilizing tissues or organs in the surgery. Compliant surgical forceps have attracted great research attention as they are much easier to assemble and sterilize than the traditional rigid-link-based forceps. in this section, we will introduce the current synthesis methods for the compliant forceps.

Pseudo-Body Replacement Synthesis

Shuib *et al.*, 2007 proposed a disposable compliant forceps for the HIV patients (see Fig. 2-30a)). In that work, the design of the compliant forceps is inspired from the slide-crank mechanism (see Fig. 2-30b)) and hence, torsional springs are used to replace the rigid links to build a PRB model. The mechanism synthesis was performed by providing the load on the forceps jaw.

(a)   (b)
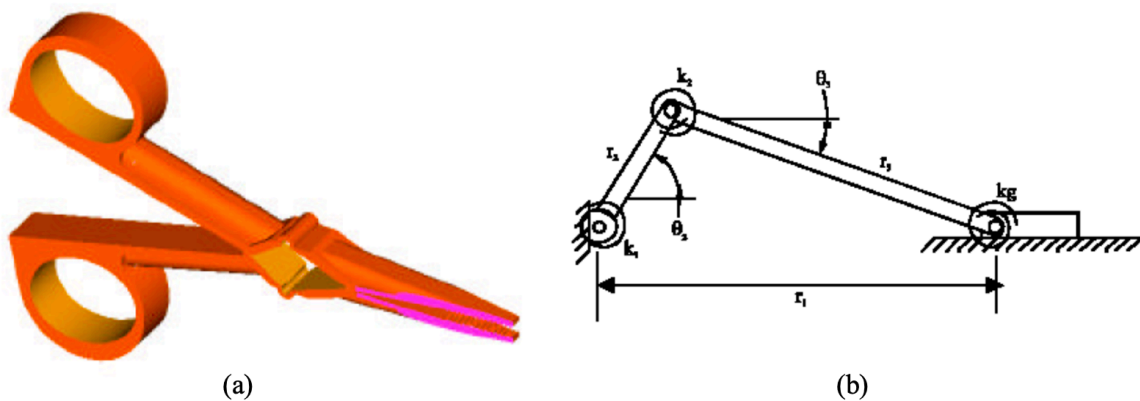
Figure 2-30: Design of a disposable compliant forceps using pseudo-body replacement synthesis (reproduced from Shuib et al., 2007): a) CAD model of the compliant forceps, b) slide-crank mechanism.
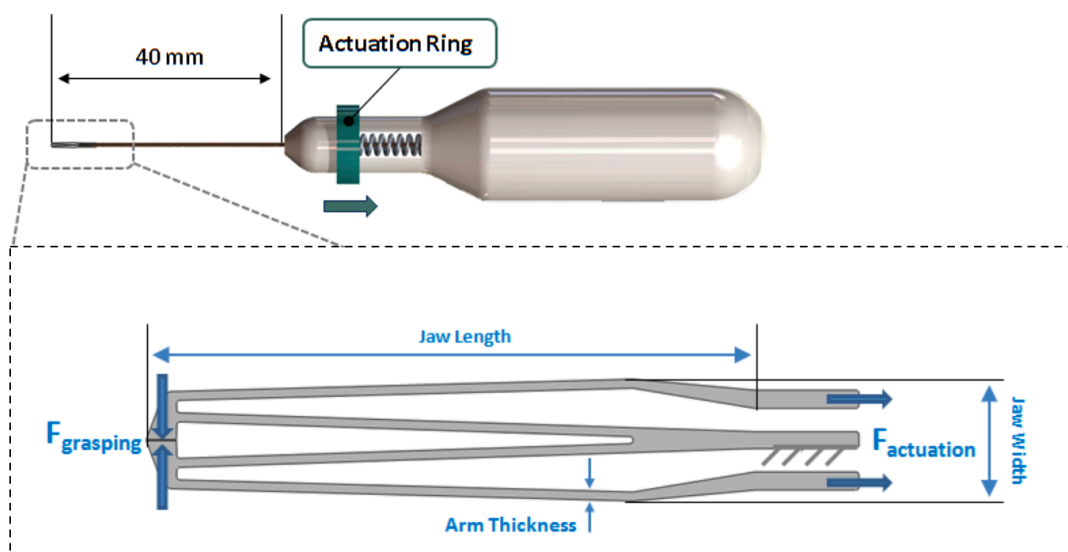
Figure 2-31: The force-sensing micro-forceps and its design parameters for size optimization (reproduced from Gonenc et al., 2013).

27

Size Optimization

Gonenc *et al.*, 2013 used a size optimization method to design a 3-DOF force-sensing micro-forceps for robot-assisted vitreoretinal surgery. The size optimization of the forceps jaws was performed under the goal of minimizing action force while preserving a minimum grasping force of 20 mN and considering laser cutting limits. The design parameters for the size optimization process are shown in Fig. 2-31. Sensitivity analysis was then performed to find the optimal parameters.

Shape Optimization

Lan and Wang, 2011 proposed a constant-force forceps for robot-assisted surgical manipulation. The constant-force mechanism was a compliant mechanism synthesized using Bezier-curve-based shape optimization (see Fig. 2-32). As can be seen in Fig. 2-32a), the end effector forceps was actuated by the constant-force mechanism via a cable. Similar works can be found in Lan, Wang and Chen, 2010; Chen and Lan, 2012.

(a)   (b)
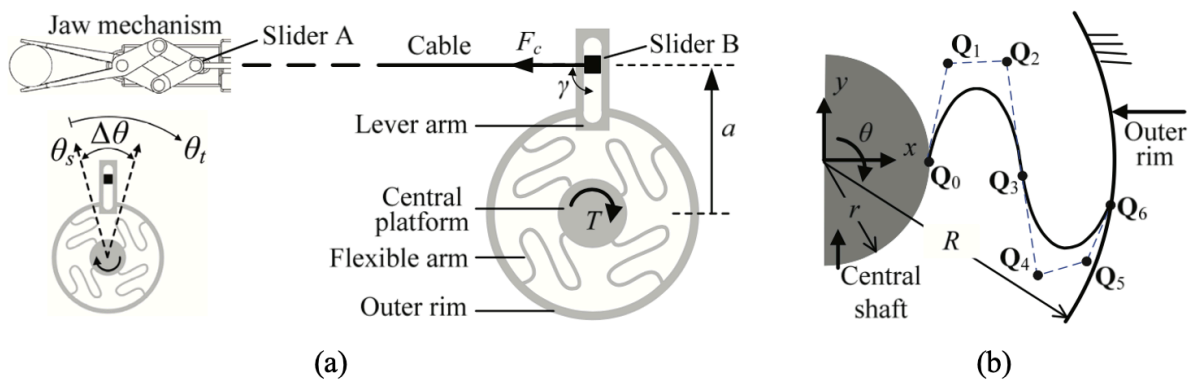
Figure 2-32: Design of a constant-force forceps using shape optimization (reproduced from Lan et al., 2011): a) The proposed contant-force mechanism, b) Bezier curve of the flexible beam with multiple control points.

Topology Optimization

Several research works were also conducted to use topology optimization methods to synthesize compliant forceps. Kota *et al.*, 2005 proposed a topology optimized compliant manipulator for positioning and manipulating kidneys in robot-assisted surgery. The manipulator was synthesized using a ground structure based topology optimization method (see Fig. 2). The final design is a three-jaw manipulator in order to increase the grasping stability (see Fig. 2). Another compliant forceps, also synthesized by a ground structure based method, was presented by

(a)   (b)
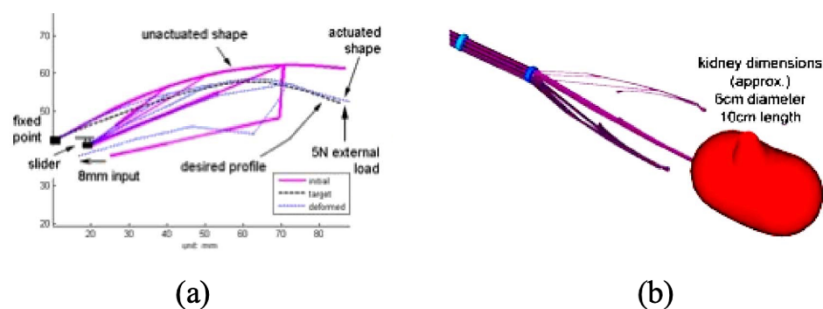
Figure 2-33: A topology optimized compliant kidney manipulator (reproduced from Kota et al., 2005): a) The synthesis result of a grasping jaw, b) CAD model of the final compliant manipulator.
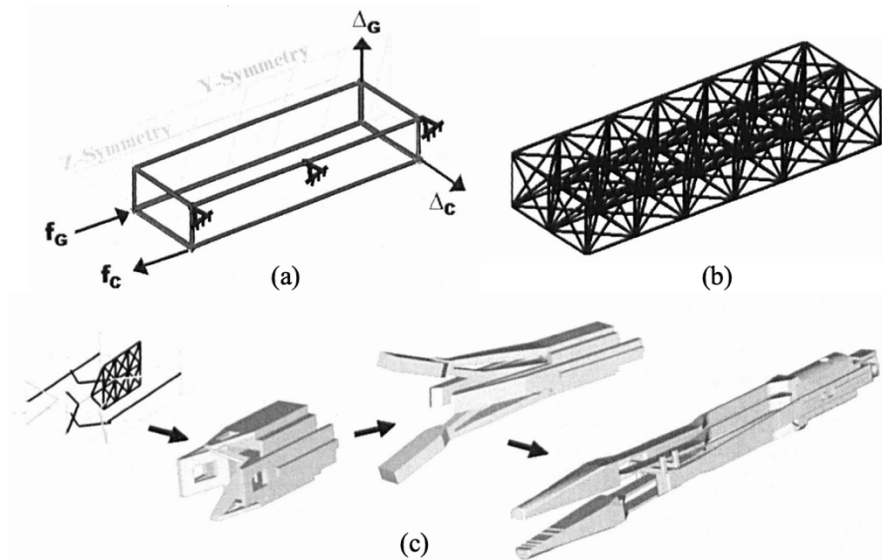
Figure 2-34: Design of a multifunctional MIS forceps using a topology optimization method (reproduced from Frecker, Dziedzic and Haluck, 2002): a) The design problem, b) The fully ground structure, c) Evolution of the forceps from synthesis result to final design.

Frecker, Dziedzic and Haluck, 2002. In that work, a multi-objective optimization formulation was proposed to achieve a multifunctional compliant forceps design. The realized forceps has successfully combined the cutting and grasping function. Another similar work that used topology optimization methods for the compliant forceps design can be found in de Lange, Langelaar and Herder, 2008.

2.3.3 Compliant Prosthetic Finger

Another important medical application of the compliant mechanisms is the compliant prosthetic finger. In Zheng *et al.*, 2016, a ground structure based topology optimization method was
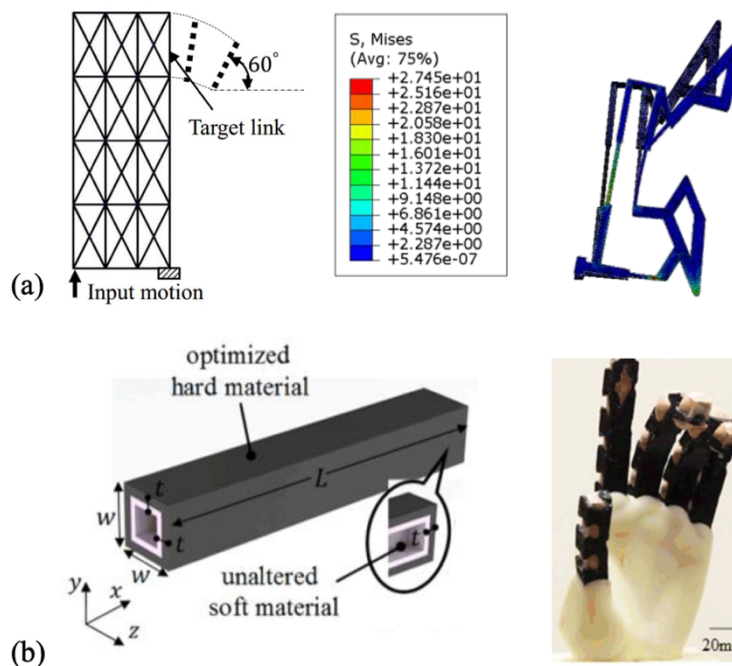
Figure 2-35: Topology optimization based design of compliant prosthetic fingers: a) Ground structure based design problem and design result (reproduced from Zheng et al., 2016), b) 3D continuum structure based design problem and design result (reproduced from Zhang et al., 2019).

proposed to synthesize a 2D fully compliant finger (see Fig. 2-35a)). The objective function is to minimize the distance between the desired deflection position of the target link and its FE simulated position. The design result is presented in Fig. 2-35a). To further develop the compliant finger, a 3D continuum structure based topology optimization method was used for the compliant finger design (Zhang *et al.*, 2019). The realized prosthetic finger, as is shown in Fig. 2-35b), consists of a soft and hard material layer and is actuated pneumatically.

2.4 Disadvantages of the State of the Art

Currently, the main-stream methods for the modeling of compliant mechanisms utilize simplification assumptions to incorporate the elastic features into the conventional rigid-link-based mechanism theory, such as the PRBM, BCM and PCCM method. The used assumptions are mainly based on the bending beam theory. Although the simplified models are proved to be close to the reality, they are inefficient to analyze complaint mechanisms with complex geometries and different actuation principles. This problem is also reflected in the synthesis methods. Using the pseudo-body replacement synthesis method, the design result is based on PRB models of specific mechanical structures, which means the design freedom is greatly constrained by the prescribed PRB models. Although the shape optimization can partly extend the design freedom using optimization-based algorithms, its performance is still limited since the topology of the mechanism should be manually determined. From this point of view, the FEM method and the continuum-structure-based topology optimization methods are more suitable for the general modeling and synthesis of compliant mechanisms since the FEM method is able to achieve high-fidelity modeling and handle geometries with any complex shape.

As is mentioned in Section 2.1.3, the current FE-analysis of compliant mechanisms is mainly performed using commercial software. Although the commercial FEM software can perform robust modeling, they have the disadvantage of being a closed black-box where the source code cannot be directly modified by the user. Hence, the implementation of new modeling methods or topology-optimization-based synthesis algorithms often requires the integration of different developing environments via additional interfaces, as in Liu and Chiu, 2017; Runge *et al.*, 2017; Chen *et al.*, 2019. In the current state of the art, although some studies have realized the FEM-based modeling and synthesis process by using one open-source platform (Sigmund, 2001; Talischi *et al.*, 2012; Liu and Tovar, 2014), it is still complicated to create very complex initial design domain for the continuum structure based topology optimization process.

As is mentioned in Section 2.3, although the topology-optimization-based design methods for general compliant mechanisms have been developed for many years, their applications to medical instruments are still rare. Even in the design cases that were described in Section 2.3, only a few applications (de Lange, Langelaar and Herder, 2008; Zhang *et al.*, 2019) utilized the continuum structure based topology optimization as design method. On the other hand, it is also necessary to further develop the continuum structure based topology optimization methods for designing task-specific or multifunctional compliant medical instruments.

3 FEM-Based Modeling of Compliant Medical Instruments

The focus of this dissertation is to develop FEM-based topology optimization methods for the automatic design of compliant medical instruments. In this chapter, the basic mechanics modeling framework, based on the FEM methods, is illustrated in detail. Firstly, the modeling tools for constructing 2D and 3D geometry models are introduced. Then, the basic linear algorithm for the 2D and 3D FEM as well as their implementation in MATLAB are presented. After that, the linear algorithms are extended so that large-displacement compliant mechanisms can be analyzed. Besides, tendon-driven mechanisms and contact problems are also taken into account for complicated modeling cases. At the end of this chapter, several bio-inspired compliant mechanisms are modeled using the proposed modeling framework to show its applications in compliant medical instruments.

3.1 Geometry Modeling Tool

There are two major representation schemata used in geometry modeling, which are Constructive Solid Geometry (CSG) and Boundary Representation (B-Rep) (Hoffmann, 1989). In the conventional computer-aided-design (CAD) software, the CSG method is always employed as the geometry modeling method, in which a 3D geometry is implicitly represented as a list of modeling steps. The created CSG-models are suitable for generating precision drawings and technical illustrations, but cannot be directly used for 3D-printing since the 3D-printable STL files are based on the B-rep method, in which an explicit description of the geometry surface is required for modeling. Since our topology optimized compliant instruments are usually fabricated using 3D printing technology, the geometry modeling tool we used is based on the B-Rep principle. Herein, we employed the Solid Geometry (SG) Library in MATLAB as the geometry tool, which was developed in our institute by Prof. Dr. Lueth (Lueth, 2015). The SG Library was implemented in MATLAB because MATLAB has integrated many powerful methods and toolboxes so that we can perform various simulations and analyzes in the same environment without additional data input and output.

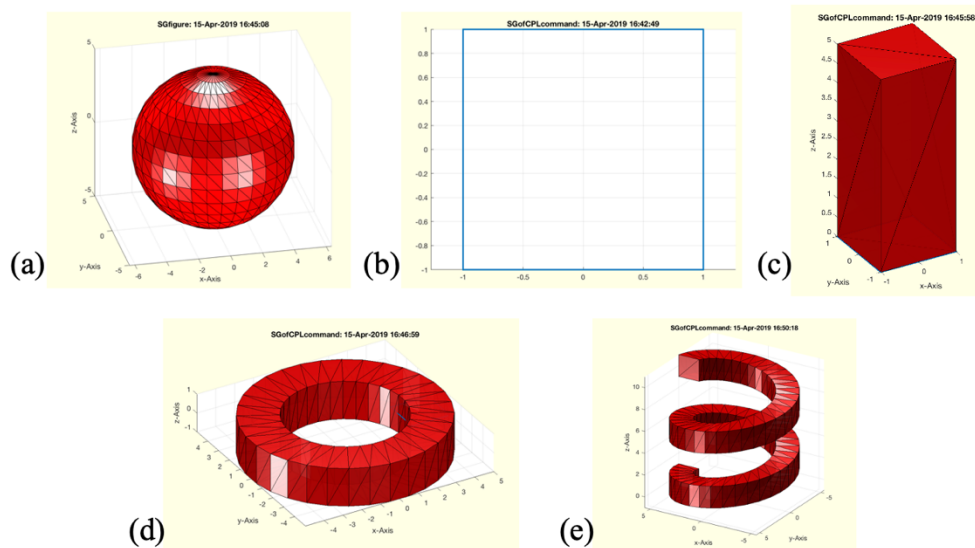**Figure 3-1: Geometry modeling examples of SG Library: a) A sphere with the radius of 5mm, b) A 2D square with the length of 2mm, c) A pillar with the height of 5mm, which is extruded by the square in b, d) A loop with the inner radius of 3mm, which is constructed by radial extrusion of the square in b, e) A helix with the pitch of 10mm, which is also constructed by radial extrusion of the square in b.**

In the surface-model-based STL file, the geometry information is stored as a list of oriented triangles tessellating the surface of a 3D solid, which has a redundant storage of the vertex information and thus inefficient for data processing (Botsch *et al.*, 2010). Therefore, we have employed the shared vertex data structure to implement the 3D geometry model, which is a struct in MATLAB composed of an unrepeated Vertex List (VL) and a triangular Facet List (FL). On the other hand, since 3D solids are usually created by vertical and radial extrusion of 2D contours, we have also implemented the 2D contour model, which is a MATLAB array containing the vertices along the contour boundary in a counterclockwise direction. For clearance, we named the 2D contour model and the 3D geometry model as Closed Polygon List (CPL) and Solid Geometry (SG), respectively. Fig. 3-1 shows several geometry examples that are created by SG Library.

The construction of a geometry model was originally realized by combining individual MATLAB functions into a script using the functional programming paradigm. However, the modeling process would be very complicated and inefficient if a complex geometry contains a large number of modeling steps. In order to simplify the modeling process of complex geometries, in which multiple modeling functions are involved, a modeling language, the Solid Geometry Coding Language (SGCL), is also developed. The SGCL language can encode the modeling steps of a geometry into a single string with an easy-to-learn syntax, and the modeling tool can interpret the coded string by calling the corresponding modeling functions. Table 3-1 provides an overview of some frequently used commands in the SGCL language. A modeling application is shown in Fig. 3-2, where the SG of a flexure hinge is extruded vertically by a CPL.

Table 3-1: Commonly Used Commands of the SGCL Language.

Command	Illustration
b (x) (y)	Create a rectangular CPL with length x and width y
c (d)	Create a circular CPL with diameter d
dupc (n_x) (n_y) (l)	Duplicate the current CPL for n_x times in x axis and n_y times in y axis, with a constant distance l
h (z)	Create a SG by vertically extruding the current CPL with a certain height z
move (x) (y) (z)	Translation of the current CPL or SG by a vector of (x, y, z)
enter	Put the current CPL or SG into stack and wait for another geometry for Boolean operation
+, -, &	Boolean operations (union, subtraction, intersection) of the current geometry and the geometry in the stack
rotx (θ)	Rotation of the current SG along the x axis by θ degree
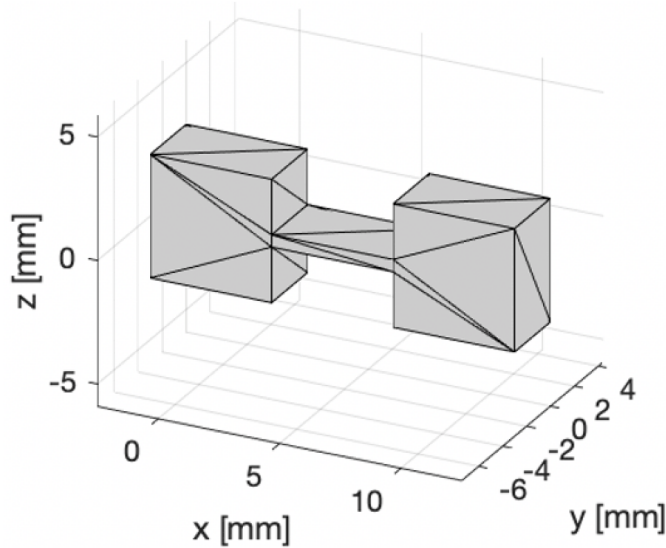
```
>> SGofCPLcommand('b 5,dupc 2 1 10,enter,b 6 0.5,move 5,+,h 3,rotx 90')
```

Figure 3-2: A flexure hinge created using the SGCL language. The code in the figure shows the modeling process.

3.2 2D Linear FEM Algorithm

There are generally two ways to perform FE-analysis in MATLAB, which are using the Partial Differential Equation (PDE) Toolbox or using self-implemented FEM code. Although the PDE Toolbox can perform robust linear FEM simulation, it has the same problem as the other commercial software because its core FEM code is also a black box. As is shown in Fig. 3-3, the PDE Toolbox takes a STL file or a SG as the geometry input to create a PDEModel object. With the other user-defined boundary conditions, the displacements of the geometry are calculated as output. However, the stiffness matrix of the geometry is implemented as a private property of the PDEModel object, which can be neither accessed nor modified. This is a remarkable disadvantage for the development of topology optimization algorithms because the evolution process is based on the iterative modification of the stiffness matrix. Therefore, we have used self-implemented code to perform the FE-analysis.
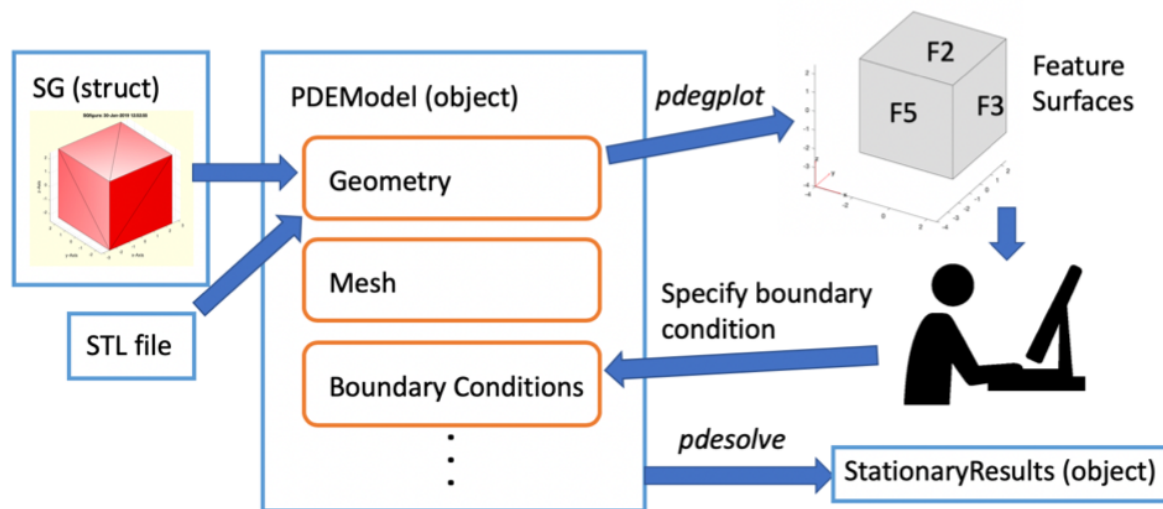
Figure 3-3: The workflow of the PDE Toolbox for performing FE-analysis.

The basics of the continuum mechanics was already described in Section 2.1.3. In this dissertation, we focus on the modeling of linear elastic materials. Using the FEM method to analyze a 2D solid, the first step is to mesh the 2D domain into small elements. A robust meshing algorithm is the built-in function `generateMesh` in MATLAB. Fig. 3-4a) shows a meshing example of a 2D flexure hinge.
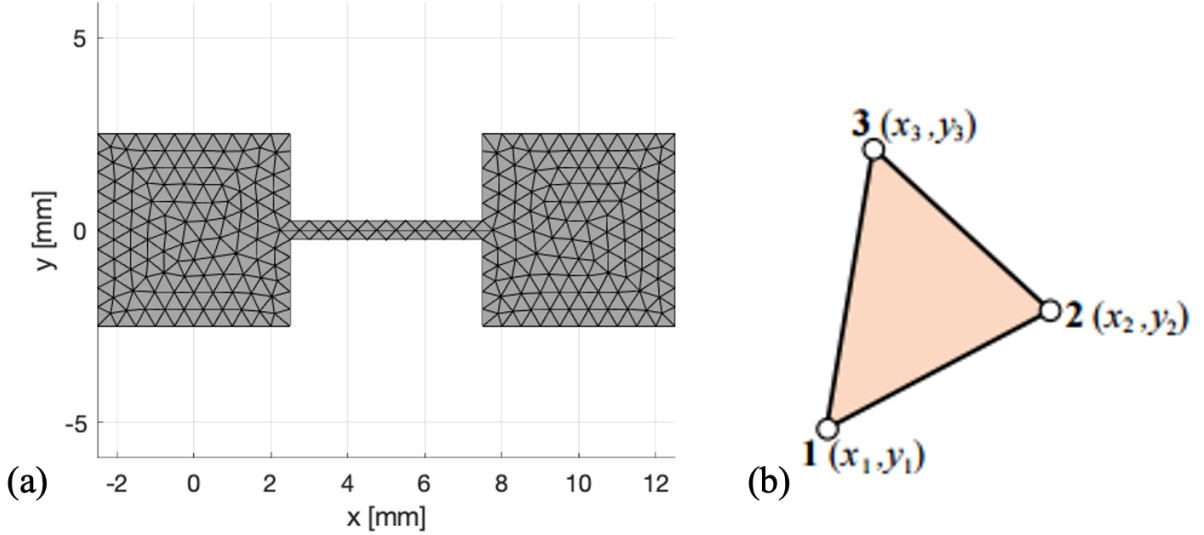
Figure 3-4: 2D meshing of a provided domain: a) A meshed 2D flexure hinge with the element size of 0.5mm, b) A triangular element.

Here, we introduce the numerical modeling methodology. As is shown in Fig. 3-4b), a triangular element is employed as the basic meshing element which has three nodes (x_1, y_1), (x_2, y_2), (x_3, y_3). The displacement and load vector ($\mathbf{u_e}$ and $\mathbf{F}_e$) of this element can then be written as:

$$\mathbf{u_e} = [u_{1,x} \quad u_{1,y} \quad u_{2,x} \quad u_{2,y} \quad u_{3,x} \quad u_{3,y}]^T \tag{3-1}$$

$$\mathbf{F}_e = [f_{1,x} \quad f_{1,y} \quad f_{2,x} \quad f_{2,y} \quad f_{3,x} \quad f_{3,y}]^T \tag{3-2}$$

The displacement of any point inside the element can be determined by linear interpolation:

$$\begin{bmatrix} u_x(x,y) \\ u_y(x,y) \end{bmatrix} = \begin{bmatrix} N_1(x,y) & 0 & N_2(x,y) & 0 & N_3(x,y) & 0 \\ 0 & N_1(x,y) & 0 & N_2(x,y) & 0 & N_3(x,y) \end{bmatrix} \mathbf{u_e} \tag{3-3}$$

$$N_1(x,y) = \frac{2(x_2 y_3 - x_3 y_2) + x(y_2 - y_3) + y(x_3 - x_2)}{2A} \tag{3-4}$$

$$N_2(x,y) = \frac{2(x_3 y_1 - x_1 y_3) + x(y_3 - y_1) + y(x_1 - x_3)}{2A} \tag{3-5}$$

$$N_3(x,y) = \frac{2(x_1 y_2 - x_2 y_1) + x(y_1 - y_2) + y(x_2 - x_1)}{2A} \tag{3-6}$$

where A is the area of the triangular element. Then, the strain and stress vector ($\boldsymbol{\varepsilon}_e$ and $\boldsymbol{\sigma}_e$) of this element can be obtained from the displacement vector:

$$\boldsymbol{\varepsilon_e} = \mathbf{B u_e} \tag{3-7}$$

$$\mathbf{B} = \begin{bmatrix} \boldsymbol{\varphi}_{1,1} & 0 & \boldsymbol{\varphi}_{2,1} & 0 & \boldsymbol{\varphi}_{3,1} & 0 \\ 0 & \boldsymbol{\varphi}_{1,2} & 0 & \boldsymbol{\varphi}_{2,2} & 0 & \boldsymbol{\varphi}_{3,2} \\ \boldsymbol{\varphi}_{1,2} & \boldsymbol{\varphi}_{1,1} & \boldsymbol{\varphi}_{2,2} & \boldsymbol{\varphi}_{2,1} & \boldsymbol{\varphi}_{3,2} & \boldsymbol{\varphi}_{3,1} \end{bmatrix}$$

(3-8)

$$\text{with} \qquad \boldsymbol{\varphi} = \begin{bmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix}^{-1} \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\boldsymbol{\sigma}_\mathbf{e} = \mathbf{E}\boldsymbol{\varepsilon}_\mathbf{e} = \mathbf{E}\mathbf{B}\mathbf{u}_\mathbf{e}$$

(3-9)

$$\mathbf{E} = \frac{E_0}{1-v^2} \begin{bmatrix} 1 & v & 0 \\ v & 1 & 0 \\ 0 & 0 & \dfrac{1-v}{2} \end{bmatrix}$$

(3-10)

where $\mathbf{B}$ and $\mathbf{E}$ are the strain-displacement matrix and the elasticity tensor, respectively. E_0 and v represent the Young's modulus and the Poisson's ratio of the elastic material. According to Equation (2-13) and (2-14), the equilibrium equation of the entire system can be formulated as:

$$\mathbf{KU} = \sum_e^N \mathbf{K}_\mathbf{e}\,\mathbf{u}_\mathbf{e} = \sum_e^N Ah\mathbf{B}^\mathbf{T}\mathbf{E}\mathbf{B}\mathbf{u}_\mathbf{e} = \sum_e^N \mathbf{F}_\mathbf{e} = \mathbf{F}$$

(3-11)

Before solving the equation system $\mathbf{KU} = \mathbf{F}$, boundary conditions should be applied to the system since the mechanical system will be otherwise singular. The boundary conditions can be divided into two types, the Dirichlet and Neumann boundary condition. The Dirichlet boundary condition represents the fixed displacement constraints while the Neumann boundary condition represents the load. For numerical implementation, the Neumann boundary condition can be realized by assembling the elementary load vector $\mathbf{F}_\mathbf{e}$ into the global load vector $\mathbf{F}$, while the Dirichlet boundary condition can be integrated into the equation system by introducing an additional equation system:

$$\mathbf{DU} = \mathbf{w}$$

(3-12)

where $\mathbf{D}$ is a constraint matrix searching the nodes in $\mathbf{U}$ which are relevant to the predefined displacement vector $\mathbf{w}$. Combining Equation (3-11) and (3-12), a new equation system can be obtained:

$$\begin{pmatrix} \mathbf{K} & \mathbf{D}^\mathbf{T} \\ \mathbf{D} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{U} \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \mathbf{F} \\ \mathbf{w} \end{pmatrix}$$

(3-12)

where $\boldsymbol{\lambda}$ is the Lagrange multiplier. By solving the Equation system (3-12), the displacement vector $\mathbf{U}$ can be calculated.

An important issue in the implementation of the FEM algorithm is to select the corresponding nodes to specify boundary conditions. In this dissertation, we have developed a method called Overlapping Region Concept (ORC) to make the specification of free-form boundary conditions more efficient and user-friendly. For the 2D FEM algorithm, the basic idea of the ORC method is to firstly create a user-defined CPL to occupy a certain 2D domain. Then we use the MATLAB function `inpolygon` to find out the nodes of the mesh which are inside the user-defined domain. Those nodes, which have created an overlapping region, will be used to specify

boundary conditions. In this way, any kinds of free-form boundary conditions can be applied to a geometry. Fig. 3-5 illustrates the principle of the ORC method graphically.

Dirichlet boundary condition (fixation)
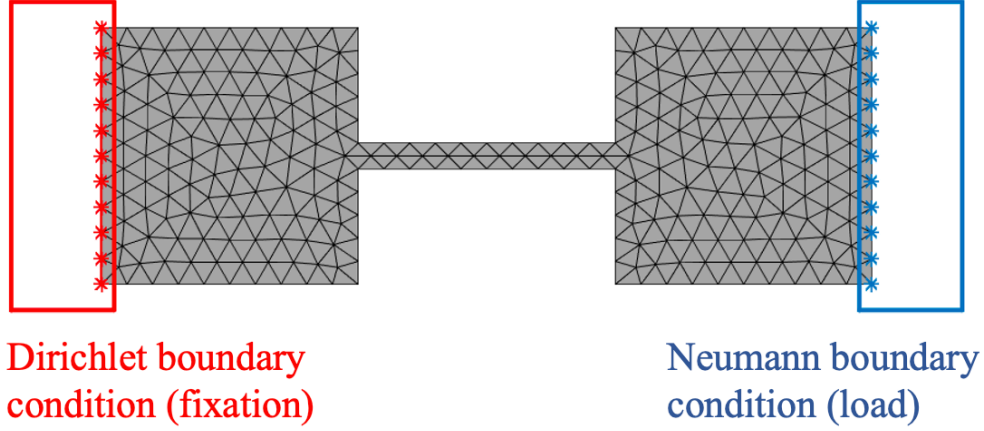
Neumann boundary condition (load)

Figure 3-5: Specification of boundary conditions using the ORC method. The red nodes indicate the nodes for applying Dirichlet boundary condition while the blue ones for the Neumann boundary condition.

3.3 3D Linear FEM Algorithm

Basically, the 3D linear FEM algorithm is an extension from the 2D version. In the 3D version, tetrahedral elements are used as basic meshing element (see Fig. 3-6). In order to adapt to the 3D problem, Equation (3-1), (3-2), (3-8), (3-10) and (3-11) should be rewritten as follows:

$$\mathbf{u_e} = [u_{1,x} \quad u_{1,y} \quad u_{1,z} \quad u_{2,x} \quad u_{2,y} \quad u_{2,z} \quad u_{3,x} \quad u_{3,y} \quad u_{3,z}]^T \tag{3-13}$$

$$\mathbf{F_e} = [f_{1,x} \quad f_{1,y} \quad f_{1,z} \quad f_{2,x} \quad f_{2,y} \quad f_{2,z} \quad f_{3,x} \quad f_{3,y} \quad f_{3,z}]^T \tag{3-14}$$

$$\mathbf{B} = \begin{bmatrix}
\boldsymbol{\varphi}_{1,1} & 0 & 0 & \boldsymbol{\varphi}_{2,1} & 0 & 0 & \boldsymbol{\varphi}_{3,1} & 0 & 0 & \boldsymbol{\varphi}_{4,1} & 0 & 0 \\
0 & \boldsymbol{\varphi}_{1,2} & 0 & 0 & \boldsymbol{\varphi}_{2,2} & 0 & 0 & \boldsymbol{\varphi}_{3,2} & 0 & 0 & \boldsymbol{\varphi}_{4,2} & 0 \\
0 & 0 & \boldsymbol{\varphi}_{1,3} & 0 & 0 & \boldsymbol{\varphi}_{2,3} & 0 & 0 & \boldsymbol{\varphi}_{3,3} & 0 & 0 & \boldsymbol{\varphi}_{4,3} \\
\boldsymbol{\varphi}_{1,2} & \boldsymbol{\varphi}_{1,1} & 0 & \boldsymbol{\varphi}_{2,2} & \boldsymbol{\varphi}_{2,1} & 0 & \boldsymbol{\varphi}_{3,2} & \boldsymbol{\varphi}_{3,1} & 0 & \boldsymbol{\varphi}_{4,2} & \boldsymbol{\varphi}_{4,1} & 0 \\
\boldsymbol{\varphi}_{1,3} & 0 & \boldsymbol{\varphi}_{1,1} & \boldsymbol{\varphi}_{2,3} & 0 & \boldsymbol{\varphi}_{2,1} & \boldsymbol{\varphi}_{3,3} & 0 & \boldsymbol{\varphi}_{3,1} & \boldsymbol{\varphi}_{4,3} & 0 & \boldsymbol{\varphi}_{4,1} \\
0 & \boldsymbol{\varphi}_{1,3} & \boldsymbol{\varphi}_{1,2} & 0 & \boldsymbol{\varphi}_{2,3} & \boldsymbol{\varphi}_{2,2} & 0 & \boldsymbol{\varphi}_{3,3} & \boldsymbol{\varphi}_{3,2} & 0 & \boldsymbol{\varphi}_{4,3} & \boldsymbol{\varphi}_{4,2}
\end{bmatrix} \tag{3-15}$$

$$\text{with} \quad \boldsymbol{\varphi} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ z_1 & z_2 & z_3 & z_4 \end{bmatrix}^{-1} \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{E} = \frac{E_0}{(1+v)(1-2v)} \begin{bmatrix}
1-v & v & v & 0 & 0 & 0 \\
v & 1-v & v & 0 & 0 & 0 \\
v & v & 1-v & 0 & 0 & 0 \\
0 & 0 & 0 & 1-2v & 0 & 0 \\
0 & 0 & 0 & 0 & 1-2v & 0 \\
0 & 0 & 0 & 0 & 0 & 1-2v
\end{bmatrix} \tag{3-16}$$

$$\mathbf{KU} = \sum_e^N \mathbf{K_e}\, \mathbf{u_e} = \sum_e^N V_e \mathbf{B^T E B u_e} = \sum_e^N \mathbf{F_e} = \mathbf{F} \tag{3-17}$$
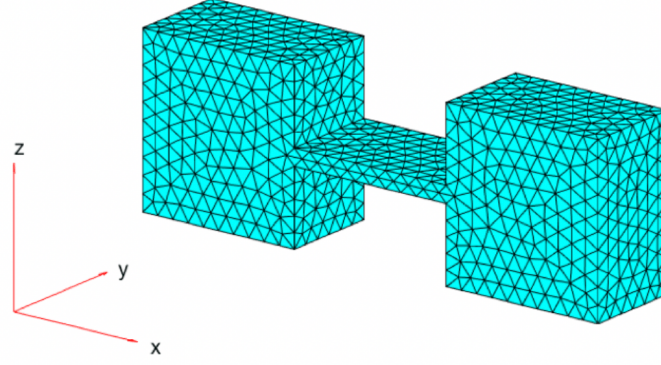
Figure 3-6: The meshed tetrahedral elements of the geometry in Fig. 3-2 with the maximum element length of 0.5mm.

where V_e is the volume of the tetrahedral element. Similar to the 2D version, the ORC method can also be applied to the 3D problem by using the function `SGisInterior` to detect the overlapping nodes (see Fig. 3-7).
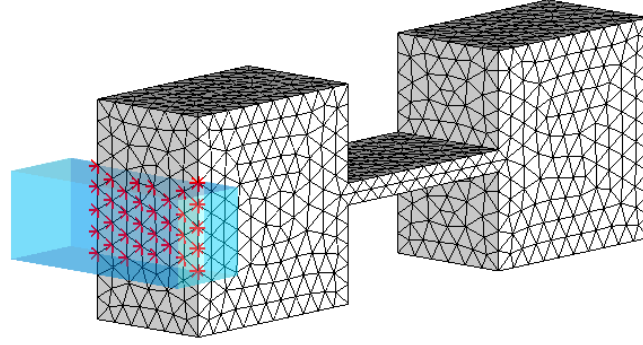

Figure 3-7: 3D application of the ORC method. The blue cuboid and the red nodes represent the user-defined domain and the detected overlapping region for applying boundary conditions, respectively.

3.4 Modeling of Large Displacements

For the structure with small displacement **U**, the Equation (3-11) and (3-17) are reasonable and **K** can be treated as a constant matrix since the geometry changes little by the small displacement. However, for the large-displacement compliant mechanisms, **K** is inconstant and dependent on the displacement **U** during the large deflections. In this case, the equation system in (3-11) and (3-17) becomes a non-linear system which cannot be easily solved by the elimination method. To cope with this problem, we have developed an incremental-load method to integrate the large-displacement feature into our modeling framework. We firstly divide **F** equally into N_{inc} small increments $\Delta\mathbf{F}$ and these incremental loads are applied to the continuum structure in N_{inc} steps. In this way, the linear FEM formulation in Equation (3-11) and (3-17) can be easily modified as in (3-18) to approximate the incremental displacement $\Delta\mathbf{U_i}$, since $\Delta\mathbf{U_i}$ is so small that the linear theory is still valid. $\mathbf{K_i}$ is the global stiffness matrix of the deformed geometry in the i-th step. The total displacement $\mathbf{U_i}$ and stress tensor $\boldsymbol{\sigma_i}$ can be obtained by accumulating $\Delta\mathbf{U_i}$ and $\Delta\boldsymbol{\sigma_i}$ as in Equation (3-19) and (3-20), respectively.

$$\mathbf{K_i}\Delta\mathbf{U_i} \approx \Delta\mathbf{F} \quad \text{with} \quad \Delta\mathbf{F} = \frac{\mathbf{F}}{N_{inc}} \tag{3-18}$$

$$\mathbf{U_i} = \mathbf{U_{i-1}} + \Delta\mathbf{U_i} \tag{3-19}$$

$$\boldsymbol{\sigma_i} = \boldsymbol{\sigma_{i-1}} + \Delta\boldsymbol{\sigma_i} = \boldsymbol{\sigma_{i-1}} + \mathbf{EB_i}\Delta\mathbf{U_i} \tag{3-20}$$

In this dissertation, we focus on modeling compliant mechanisms with incompressible materials. According to the law of mass balance, the volume of incompressible geometry should remain constant, even at large displacements. To achieve plausible simulation results, we have formulated the inequality in Equation (3-21) to make the sum $\epsilon(N_{inc})$ of the volume errors smaller than a prescribed value $\epsilon_{V,tol}$.

$$\epsilon(N_{inc}) = N_{inc} \cdot \epsilon_{V,1} < \epsilon_{V,tol} \tag{3-21}$$

$$\epsilon_{V,1} = \left| \frac{V(\Delta\mathbf{F}) - V_0}{V_0} \right| \tag{3-22}$$

$\epsilon_{V,1}$ in Equation (3-21) is illustrated by (3-22) that represents the volume error brought by the first incremental load, where V_0 and $V(\Delta\mathbf{F})$ are the volume of the original and deformed geometry, respectively. $V(\Delta\mathbf{F})$ is calculated based on the updated coordinates of the geometry model with the linear displacement. Assuming that the volume error of each step is almost the same by applying the identical $V(\Delta\mathbf{F})$, $\epsilon(N_{inc})$ can be approximated as the product of N_{inc} and $\epsilon_{V,1}$. Herein, we have employed the MATLAB function `solve` to determine the minimum N_{inc} that satisfies (3-21). Algorithm 1 shows the implemented algorithm of the proposed incremental-load method.

Algorithm 3-1: Algorithm of the Incremental-Load Method for Modeling Large Displacements.

---

1   Initialization of $\mathbf{K_0}$, $\mathbf{F}$, $\mathbf{B_0}$, and $\epsilon_{V,tol}$;

2   Calculate N_{inc} by solving $\epsilon(N_{inc}) < \epsilon_{V,tol}$;

3     $\Delta\mathbf{F} \leftarrow \dfrac{\mathbf{F}}{N_{inc}}$;

4     $\mathbf{U_0} \leftarrow \mathbf{0}, \boldsymbol{\sigma_0} \leftarrow \mathbf{0}$;

5   **for** $i = 1:N_{inc}$ **do**

6   **if** $i > 1$ **then**

7   Calculate $\mathbf{K}_i$ and $\mathbf{B}_i$ based on $\mathbf{U}_{i-1}$;

8   **else**

9             $\mathbf{K}_i \leftarrow \mathbf{K_0}, \mathbf{B}_i \leftarrow \mathbf{B_0}$;

10   **end if**

11    $\Delta\mathbf{U_i} \leftarrow \mathbf{K_i^{-1}} \cdot \Delta\mathbf{F}$;

12    $\mathbf{U_i} \leftarrow \mathbf{U_{i-1}} + \Delta\mathbf{U_i}$;

---

13        $\sigma_i \leftarrow \sigma_{i-1} + \mathbf{EB_i}\Delta\mathbf{U_i};$

14   **end for**

The large-displacement modeling of a compliant mechanism is graphically illustrated in Fig. 3-8, where the 3D flexure hinge in Fig. 3-2 is used as example and bent by a constant force. Firstly, user-defined geometries are created as fixed domains and loading domains for defining loading cases, as is shown in Fig. 3-8a). Using the ORC method, the nodes on the surface of the meshed geometry model, which are inside these user-defined domains, are detected for applying boundary conditions. Then, Algorithm 3-1 is used to calculate the large displacements. Fig. 3-8b) and Fig. 3-8c) show the modeling results by using the linear FEM ($N_{inc} = 1$) and the proposed incremental-load method ($N_{inc} > 1$), respectively. It can be noticed that, the linearly modeled geometry in Fig. 3-8b) swells greatly after one-step large displacement while the geometry volume of the non-linear model changes little during the incremental-load process. The
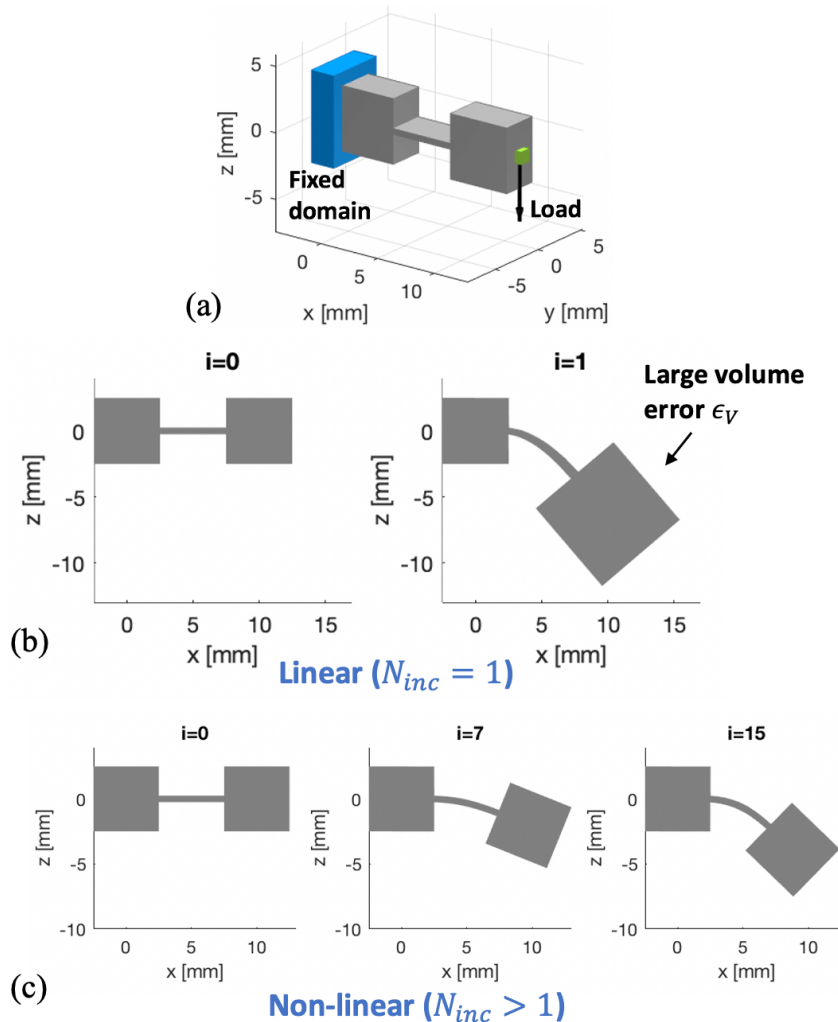
Figure 3-8: Large-displacement modeling of a compliant mechanism: a) Loading cases of a compliant mechanism. The flexure hinge in Fig. 3-2 is used for illustration. The surface nodes in the green domain are loaded by a force in z-axis and the ones in the blue domain are fixed, b) The linear FE-analysis. The deformation is presented and a large volume error can be observed, c) The non-linear FE-analysis using the proposed incremental-load method. The intermediate shapes (i = 0, 7, 15) during the incremental-load process $N_{inc} = 15$, $\epsilon_{V,tol} = 0.02$ are presented, which show little volume error.

advantage of the presented incremental-load method over the linear FEM is thus illustrated. On the other hand, the proposed method, although based on statics, can also be used to solve quasi-static problems when the speed of movement is considered very slow. In our modeling framework, the incremental-load method is developed as the basis of the non-linear FEM formulation.

3.5 Modeling of Tendon-Driven Mechanisms

Another important feature of our modeling framework is the modeling of tendon-driven mechanisms, as they are frequently used in the actuation of continuum medical robot systems, such as the snake-like soft robot in Fig. 2-27 and Fig. 2-28. Fig. 3-9 shows the typical structure of a tendon-driven compliant mechanism, which is comprised of a bendable continuum body and a actuation tendon. The continuum body consists of a series of thin-wall flexure hinges and connecting disks, while the tendon is attached at the free end of the continuum body and goes through the disks to actuate the bendable structure. The tendon brings mainly two kinds of forces to the compliant mechanism, which are the tensile force T at the attaching point and the bending forces $\{F_{b,k}\}_{k=1,2,...,N_P-1}$ applied on the contact points of the disks, as is shown in the diagram of Fig. 3-9. The friction forces on the disks are not considered in this paper. In our notation, b stands for "bending" and k is the index of the N_P contact points P_k between the tendon and the disks. As is graphically illustrated in Fig. 3-9, $F_{b,k}$ can be treated as the addition of the tensile forces before and after the contact point P_k. As the tensile force in the entire tendon is identical, $F_{b,k}$ can be calculated by using the following equations:

$$\mathbf{d}_{k-1,k} = \frac{\mathbf{x}_{P_k} - \mathbf{x}_{P_{k-1}}}{\left\|\mathbf{x}_{P_k} - \mathbf{x}_{P_{k-1}}\right\|_2} \tag{3-23}$$
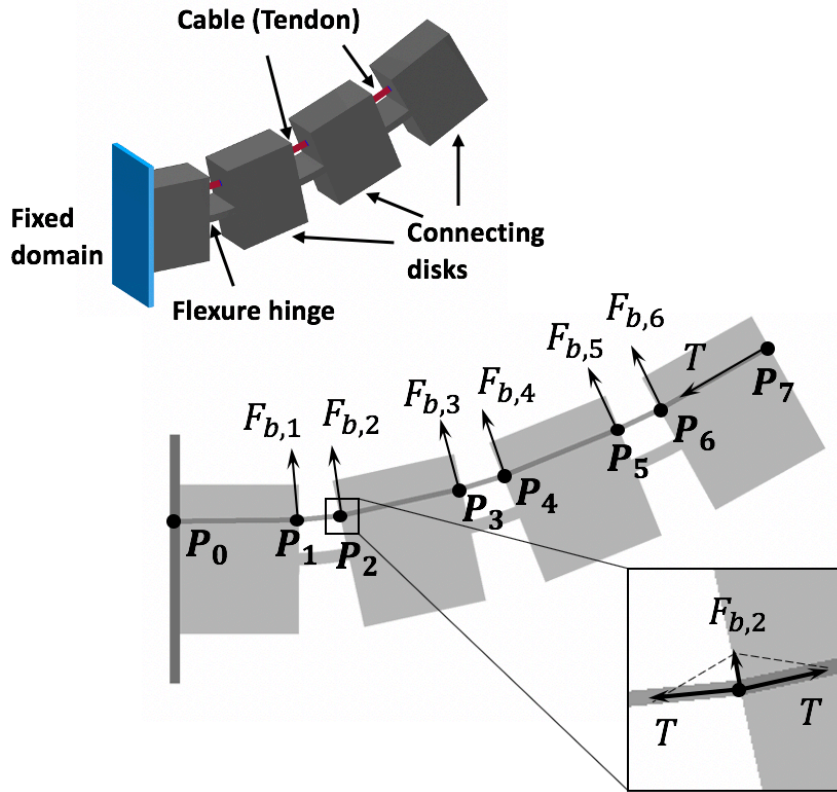

Figure 3-9: A compliant mechanism actuated by a single tendon. A schematic diagram is presented illustrating the principle of modeling the forces introduced by the tendon.

$$F_{b,k} = T \cdot \left\| \mathbf{d}_{k,k+1} - \mathbf{d}_{k-1,k} \right\|_2 \tag{3-24}$$

$$\mathbf{e}_{b,k} = \frac{\mathbf{d}_{k,k+1} - \mathbf{d}_{k-1,k}}{\left\| \mathbf{d}_{k,k+1} - \mathbf{d}_{k-1,k} \right\|_2} \tag{3-25}$$

where $\mathbf{x}_{P_k}$ is the coordinate of $\boldsymbol{P}_k$ and $\mathbf{d}_{k-1,k}$ indicates the normalized direction from $\boldsymbol{P}_{k-1}$ to $\boldsymbol{P}_k$. The force direction $\mathbf{e}_{b,k}$ of $F_{b,k}$ can be determined by Equation (3-25).

As can be seen in Equation (3-24) and (3-25), the magnitude and direction of the forces introduced by the tendon are correlated with the displacement of the compliant mechanism, which introduces an additional non-linearity $\mathbf{F} = \mathbf{F}(\mathbf{U})$ to the FEM formulation. From this point of view, the incremental-load method in Section 3.4 cannot be directly used for modeling the tendon as the load vector $\mathbf{F}$ was assumed constant in Algorithm 3-1. To cope with this problem, we have extended the non-linear FEM formulation in Algorithm 3-1 to realize the modeling of tendon-driven mechanisms. Firstly, the tendon is defined as a series of loading domains as input for detecting the nodes of $\boldsymbol{P}_k$. T is also predefined and incorporated into $\mathbf{F}$ for determining N_{inc} and the initial $\Delta \mathbf{F}$ (similar to line 2 and 3 in Algorithm 3-1). Then, during the incremental-load process, the magnitude of the tendon force changes (ΔT and $\Delta F_{b,k}$) are calculated in each step by:

$$\Delta T = \frac{T}{N_{inc}} \tag{3-26}$$

$$\Delta F_{b,k} = \Delta T \cdot \left\| \mathbf{d}_{k,k+1} - \mathbf{d}_{k-1,k} \right\|_2 \tag{3-27}$$

where the direction of ΔT and $\Delta F_{b,k}$ can be determined by Equation (3-23) and (3-25) in each step, respectively. After that, we update the $\Delta \mathbf{F}$ in line 10 of Algorithm 3-1 by accumulating the calculated ΔT and $\Delta F_{b,k}$ which are applied on the detected nodes of $\boldsymbol{P}_k$. With these modifications in Algorithm 3-1, the non-linear feature of the tendon forces is integrated into our non-linear FEM formulation.

3.6 Modeling of Contact Problems

As collisions between different parts often occur when a compliant mechanism undergoes large displacements, it is necessary to integrate the modeling of contact problems into our modeling framework for achieving plausible simulation results. The contact problems in continuum mechanics can be considered as a non-linearity of boundary conditions (Kim, 2014), which emerges when two or more surfaces penetrate each other. As can be seen in Fig. 3-10a), the contact-related non-linear boundary condition can be modeled as a pair of inverse forces applied on the contact boundaries to separate the penetrating surfaces. To model the contact problems by using FEM, an important issue is to the detect the discrete nodes on the contact boundaries and then calculate the generated contact forces applied on the detected nodes. Herein, we have employed the slave-master concept and the node-to-facet contact search method from Kim, 2014 for calculating contact forces. All contact problems in this paper are treated as frictionless. In this section, our solution for the 3D contact problems is described while its 2D version follows the same principle.

The diagram in Fig. 3-10b) provides a graphical illustration of the node-to-facet contact search method. Firstly, a pair of geometries, called the slave body and the master body, are chosen as
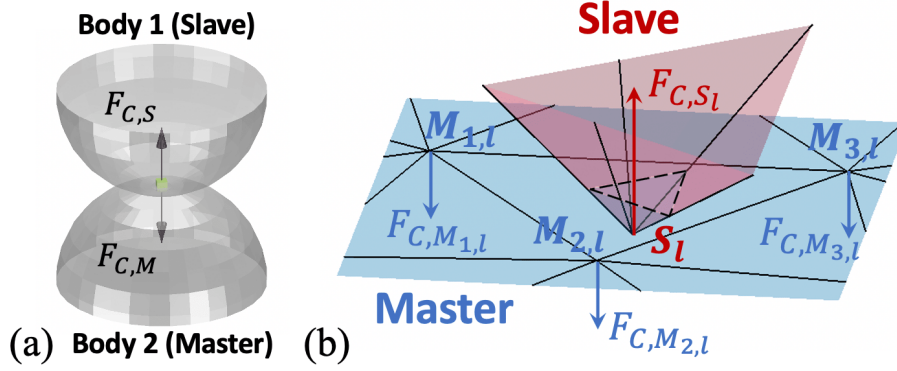
Figure 3-10: Graphical illustration of the modeling of contact problems: a) A typical contact problem of two continuum bodies. $F_{C,S}$ and $F_{C,S}$ are the generated contact force pair applied on the contact boundary (green domain), b) A diagram illustrating the modeling of contact forces using the slave-master concept and the node-to-facet contact search method.

potential contact domains. Then, during the incremental-load process, all nodes on the slave surface are checked in each step to see if they penetrate any facet of the master surface. A contact problem is detected if any penetration between the slave nodes and the master facets occurs. Here, we suppose that a contact problem consists of N_{cp} contact pairs, where the l-th contact pair is comprised of a slave node S_l and the corresponding master nodes of the penetrated facet $\{M_{j,l}\}_{j=1,2,3}$, as is shown in Fig. 3-10b). The contact forces generated by the l-th contact pair are calculated based on the principle that larger penetration needs larger contact force for elimination. Following this principle, the relationship between the slave contact force F_{C,S_l} and the penetration depth g_l is described by Equation (3-30), where K_C is a scaling factor. g_l is defined in Equation (3-29) as the normal distance between S_l and the master facet $\overline{M_{1,l}M_{2,l}M_{3,l}}$, where $\mathbf{e}_{n,l}$ is the normal direction of $\overline{M_{1,l}M_{2,l}M_{3,l}}$. According to Newton's third law of motion, the sum of the master contact forces is equal to F_{C,S_l} (see Equation (3-31)).

$$\mathbf{e}_{n,l} = \frac{(\mathbf{x}_{M_{2,l}} - \mathbf{x}_{M_{1,l}}) \times (\mathbf{x}_{M_{3,l}} - \mathbf{x}_{M_{2,l}})}{\left\|(\mathbf{x}_{M_{2,l}} - \mathbf{x}_{M_{1,l}}) \times (\mathbf{x}_{M_{3,l}} - \mathbf{x}_{M_{2,l}})\right\|_2} \tag{3-28}$$

$$g_l = (\mathbf{x}_{M_{1,l}} - \mathbf{x}_{S_l}) \cdot \mathbf{e}_{n,l} \tag{3-29}$$

$$F_{C,S_l} = K_C \cdot g_l \tag{3-30}$$

$$F_{C,M_{1,l}} = F_{C,M_{2,l}} = F_{C,M_{3,l}} = \frac{F_{C,S_l}}{3} \tag{3-31}$$

To ensure that the penetrating surfaces can be successfully separated by the calculated contact forces, we have developed a numerical process to calculate K_C, in which K_C is firstly assigned as the Young's modulus E_0 and then iteratively updated by multiplying the correction coefficient η in Equation (3-32) until the absolute value of the maximum penetration depth g_{l_m} is smaller than a prescribed tolerance ϵ_C with the applied contact forces. g_0 is the maximum penetration depth before applying the contact forces.

$$\eta = \frac{g_0}{g_0 - g_{l_m}} \tag{3-32}$$

The presented modeling method of contact problems has been implemented in Algorithm 3-2, which can be integrated into our modeling framework by extending line 10 in Algorithm 3-1.

Algorithm 3-2: Algorithm of the Modeling of Contact Problems (Extending Line 10 in Algorithm 3-1).

|	
1	$\Delta \mathbf{U_i} \leftarrow \mathbf{K_i^{-1}} \cdot \Delta \mathbf{F}$;
2	Calculate N_{cp};
3	**if** $N_{cp} > 0$ **then**
4	**for** $l = 1 : N_{cp}$ **do**
5	Calculate $\mathbf{e}_{n,l}$ and g_l;
6	$F_{C,S_l} \leftarrow K_C \cdot g_l$;
7	**for** $j = 1 : 3$ **do**
8	$F_{C,M_{j,l}} \leftarrow \frac{F_{C,S_l}}{3}$;
9	**end for**
10	**end for**
11	$g_{l_m} \leftarrow \max(g_l), g_0 \leftarrow g_{l_m}$;
12	$\eta \leftarrow 1$;
13	**while** $\left| g_{l_m} \right| \geq \epsilon_C$ **do**
14	$\Delta \mathbf{F_i} \leftarrow AddForce(\Delta \mathbf{F}, \eta \cdot F_{C,S_l}, \eta \cdot F_{C,M_{j,l}})$;
15	$\Delta \mathbf{U_i} \leftarrow \mathbf{K_i^{-1}} \cdot \Delta \mathbf{F_i}$;
16	Recalculate g_{l_m};
17	$\eta \leftarrow \frac{g_0}{g_0 - g_{l_m}}$;
18	**end while**
19	**end if**

3.7 Modeling Examples

In this section, the proposed FEM-based modeling framework was applied in the analysis of three kinds of compliant medical instruments, which are a snake-like continuum manipulator, a flytrap-like compliant forceps and a fish-fin-inspired adaptive finger. The simulation results were compared with the 3D-printed prototypes of the instruments to demonstrate the performance of the proposed modeling methods. All the calculations were executed on a computer with an Intel Core i7 CPU at 2.9 GHz.

3.7.1   Snake-Like Continuum Manipulator

In our institute, we are developing 3D printable robotic systems for minimally invasive surgery (MIS). The snake-like robots are used in our applications because of their high flexibility and tangible benefits to the patients. Fig. 3-11a) shows the prototype of a selective laser sintered (SLS) continuum manipulator, which is equipped with a snake-like robotic structure for dexterous bending movements and an adaptive compliant forceps for safe manipulation of sensitive tissues. The adaptive compliant forceps is designed using topology optimization methods, which will be illustrated in the following chapters. For this section, we focus on the modeling of the snake-like robotic structure. Unlike the tendon-driven mechanism in Fig. 3-9, the orientations of the neighboring flexure hinges in Fig. 3-11a) are always vertical to each other and the continuum structure is actuated by two tendons. In this way, the continuum structure can be bent in different directions. In this section, we use the presented modeling framework to simulate the snake-like robotic structure actuated by a single tendon.

(a)
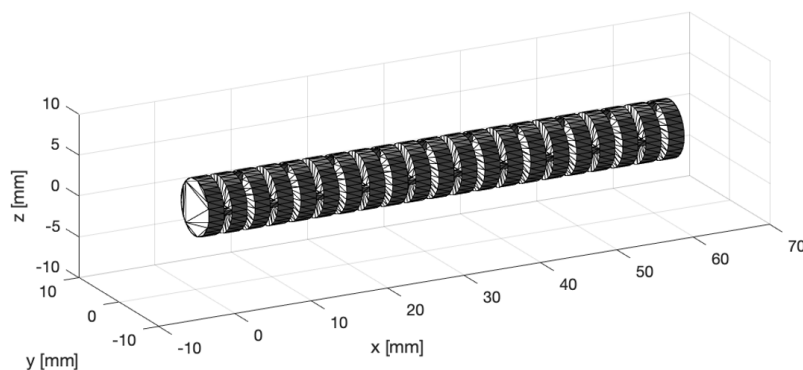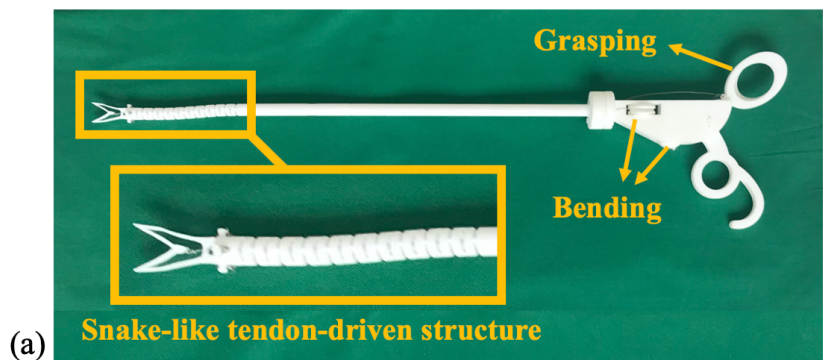

```
>> SGofCPLcommand(['c 7,h 2,dups 1 1 21 1,enter,b 0.9 0.5,'...
'move -3,dupc 4 1 2,h 1.8,move 0 0 1.8,dupg 1 1 10 6,'...
'dup,rotz 90,move 0 0 3,cat,+,roty 90,rotx 90'])
```

(b)

Figure 3-11: Snake-like continuum manipulator: a) A SLS-printed manipulator for robot-assisted MIS, b) Geometry modeling of the snake-like robotic structure.

Fig. 3-11b) shows the geometry model of the snake-like structure, which is realized using the SGCL language. The entire continuum manipulator is SLS-printed using polyamide (PA2200) as material. The PA2200 is treated as linearly elastic material in this case, and the Young's modulus and the Poisson's ratio were set to 1700 MPa and 0.3 for calculating **K**. The generated mesh was composed of 25319 nodes and 109620 elements with the maximum element size of 0.5 mm. In the simulation, the bottom of the snake-like structure was fixed while a tendon was attached to the free end and went through the connecting disks. The intersecting domains between the tendon and the disks were detected for applying tendon forces. The contact-related master and slave surfaces were located using the cyan and green domains in Fig. 3-12a) respectively to detect collisions between the disks. As the movement speed of the snake-like structure is considered slow in this paper, we have employed the proposed incremental-load method to analyze the quasi-static bending process. In order to achieve a large bending angle, a large tensile force T of 5 N is applied in the simulation to actuate the snake-like structure. With a prescribed $\epsilon_{V,tol}$ of 0.02, the simulation was performed with 50 incremental steps (240.3 seconds) and the results are presented in Fig. 3-12b). It can be noticed that, at the end of the simulation, the snake-like structure has achieved a large bending angle and the forceps of the manipulator has already passed the xz-plane of the coordinate system. From the simulation results we can see that, in the first part of the simulation (before the 30th step) the bent continuum
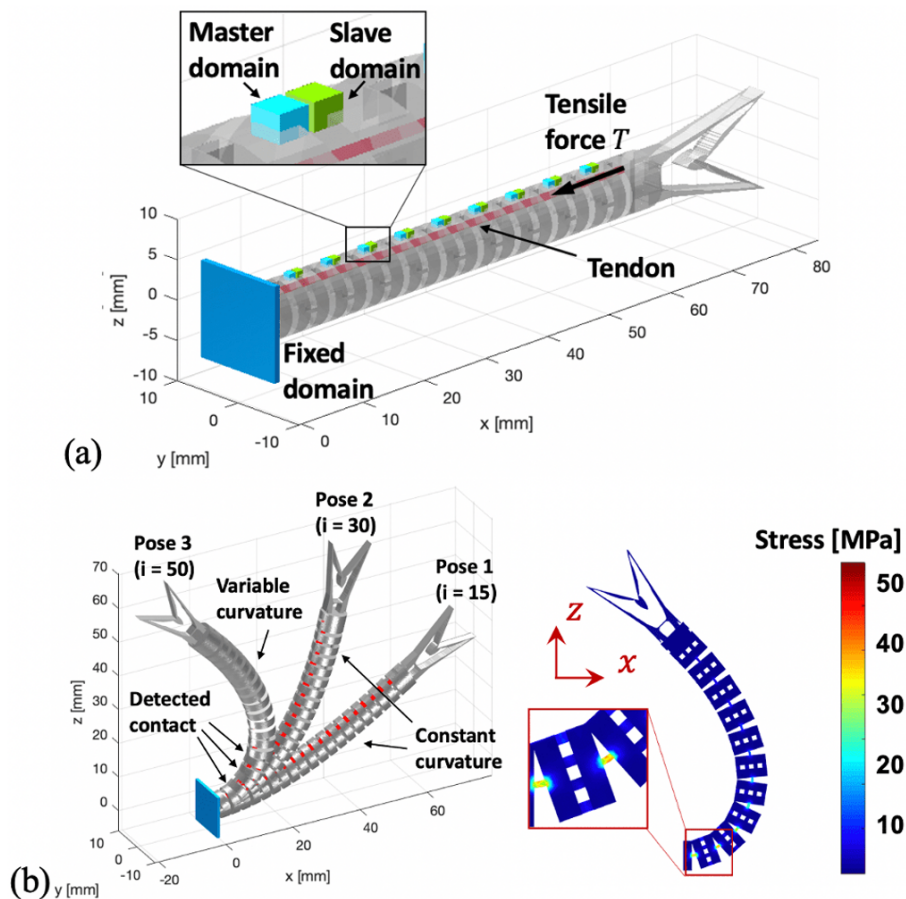

Figure 3-12: Modeling of the snake-like robotic structure in Fig. 3-11 and experimental evaluation: a) A schematic diagram showing the loading cases of the snake-like tendon-driven structure. The cyan and green domains are used to define the master and slave surfaces respectively, for detecting contact problems, b) Simulation of the snake-like structure. Intermediate poses (i = 15, 30, 50) of the deformed tendon-driven mechanism and the stress distribution of the 3rd pose are presented.
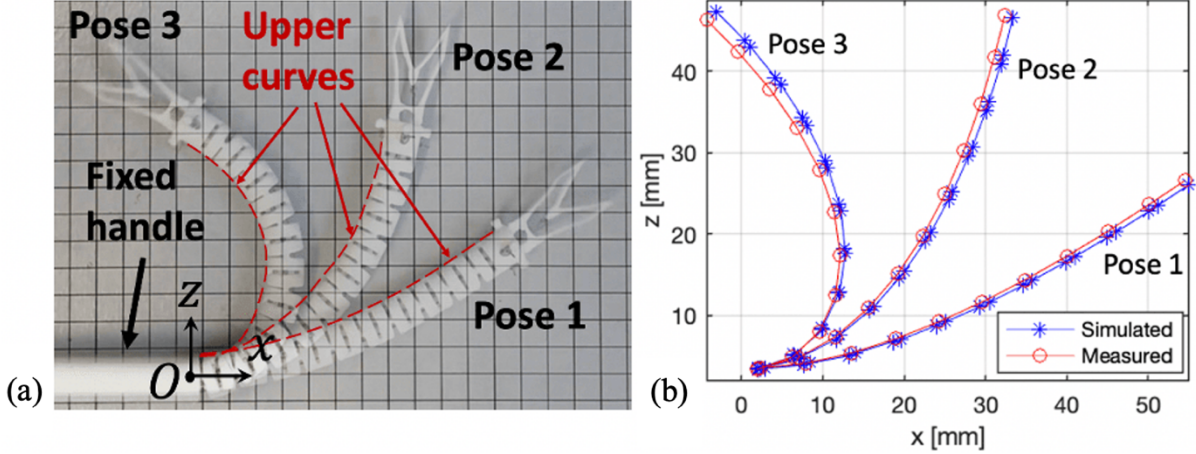
Figure 3-13: Comparison of modeling and experiment results: a) Measurement of the displacements of the snake-like structure, b) A comparison between the simulated (blue) and measured (red) upper curve of the 3 poses.

structure had a constant curvature and was contact-free, while in the last simulation steps variable curvatures and collisions between the disks emerged, which shows the limitations of the PCCM method in the large bending angles. On the other hand, the stress distribution in the third pose, as is presented in Fig. 3-12b), indicates that the most stresses that emerged in the bent continuum body were concentrated in the flexure hinges near the fixed bottom, which also shows that the bending curvature was not identical in the entire structure.

An experiment was also carried out to evaluate the plausibility of the simulated displacements. In the experiment, the handle of the continuum manipulator was fixed and the snake-like structure was slowly bent with pre-calculated tendon pulling lengths (1.67 mm, 3.58 mm and 5.67 mm) to achieve the 3 simulated poses in Fig. 3-12b). A digital microscope (Conrad DP-M17) was used to measure the upper curve of the continuum body during the bending movement, as is shown in Fig. 3-13a). A comparison between the simulated and measured curves is presented in Fig. 3-13b). The results show that the maximum error between the simulated upper curves and the reality is only 1.2 mm, which reflects the high accuracy of our FEM-based modeling framework.

3.7.2 Flytrap-Like Compliant Forceps

The second example of compliant medical mechanisms is a medical forceps inspired by Venus flytrap. The Venus flytrap is a small plant that catches insects with a trapping structure composed of two leaves (see Fig. 3-14a)). The trapping mechanism is realized by transforming the two leaves from a totally flat surface to a convex and a concave surface, which is highly flexible and efficient. Inspired by this interesting property, Edmondson *et al.*, 2013 have developed a laser-cut compliant forceps with foldable origami structures, where the flexure hinges were fabricated using paper or metallic glass. Based on the work in Edmondson et al., 2013, we have modeled the flytrap-like compliant forceps in this paper using our modeling framework and also fabricated it with the SLS printing technology. Fig. 3-14b) shows the realized basic geometry of the flytrap-like forceps. In order to achieve the clamping movement, the center panel of the structure was fixed and forces were applied on the two side panels (see Fig. 3-15a)). The panel thickness t_{panel} was 1 mm in this case. Since the polyamide has a high stiffness, the thickness of the folding area t_{fold} must be very thin so that the origami structure is bendable. To determine the optimal t_{fold}, we have developed a structural optimization algorithm to iteratively modify t_{fold} so that the strain and stress inside the forceps are mainly concentrated in
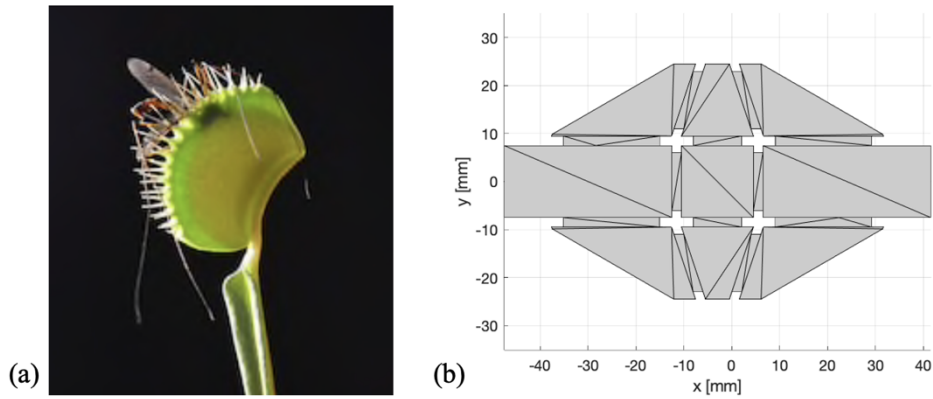
Figure 3-14: The Venus flytrap and the inspired forceps: a) The Venus flytrap, b) The created 3D geometry model of the flytrap-like forceps (z-axis view).

the folds. The algorithm can be illustrated by Equation (3-33) and (3-34). The stress concentration factor λ of the folding area is defined as in Equation (3-33), where $\bar{\sigma}_f$ and $\bar{\sigma}_{nf}$ are the average stresses in the prescribed folding area and in the rest of the forceps, respectively. The folds with the largest deformation, as they are marked in Fig. 3-15a), were selected for

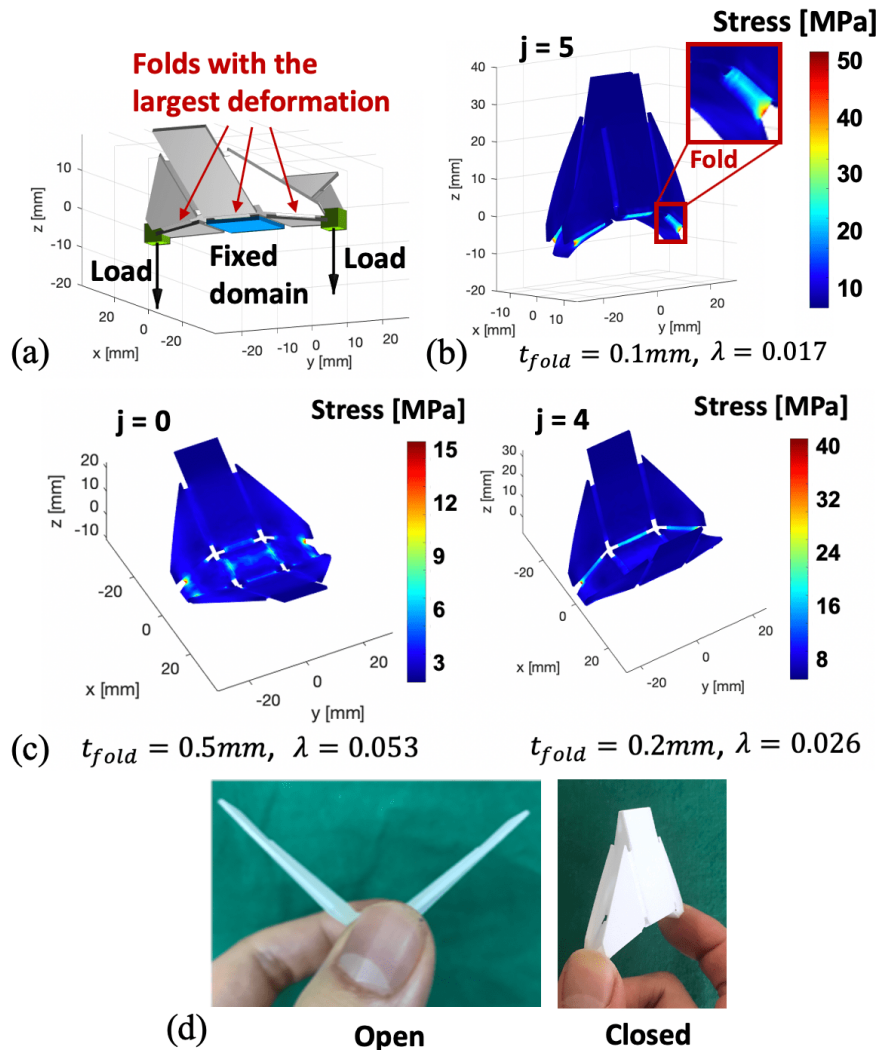**Figure 3-15: Modeling and structural optimization of the flytrap-like compliant forceps: a) A schematic diagram showing the loading cases of the flytrap-like compliant forceps, b) The calculated stress distribution of the optimized forceps, c) Stress distribution of the deformed forceps at some iterations of the optimization process, d) The SLS-printed forceps in the open and closed state.**

47

calculating $\bar{\sigma}_f$. The optimization objective is to achieve the minimum absolute difference between λ and the expected stress concentration factor λ_0. Since the SLS printer (EOS Formiga P100) has a minimum printable thickness t_{min} of 0.01 mm, the design variable t_{fold} should be greater than t_{min} while smaller than t_{panel}. The updating scheme of t_{fold} is formulated as Equation (3-34), where $t_{fold,j}$ is the fold thickness in the j-th iteration while $t_{fold,0}$ was set to 0.5 mm to start the optimization process.

$$\min_{t_{fold}} \left| \lambda(t_{fold}) - \lambda_0 \right|$$

$$s.t.:\ \lambda(t_{fold}) = \frac{\bar{\sigma}_{nf}}{\bar{\sigma}_f},\quad t_{panel} > t_{fold} > t_{min}$$

(3-33)

$$t_{fold,j+1} = t_{fold,j} \cdot \frac{1}{1 + \ln\frac{\lambda}{\lambda_0}},\qquad t_{fold,0} = 0.5mm$$

(3-34)

The optimization process of the flytrap-like forceps is presented in Fig. 3-15b) and Fig. 3-15c). The generated mesh in this case was comprised of 8632 nodes and 25613 elements with the maximum element size of 1 mm. λ_0 was set to 0.02. The optimization process converged at the 5th iteration (165.7 seconds in total) within a tolerance of 0.004 and the result was presented in Fig. 3-15b). According to the calculated stress distribution in Fig. 3-15b), the highest stress was in the predefined folds, which is consistent with the optimization goal. With the optimized t_{fold}, the SLS-printed flytrap-like forceps can successfully achieve the clamping movement (see Fig. 3-15d)), which also shows the reliability our modeling framework.

3.7.3 Fish-Fin-Inspired Compliant Finger

The third example of bio-inspired compliant mechanisms is a fish-fin-like adaptive finger (Aguib, 2011). As is shown in Fig. 3-16a), the biological fish fins are very flexible, allowing them to fit in almost any shape. Inspired by this interesting feature, Mahl, Hildebrandt and Sawodny, 2014 have developed an adaptive compliant gripper, which has an isosceles triangular structure with multiple parallel stabilizers between its elastic gripping jaws. As the adaptive gripper could be used to safely grasp organs or tumors of irregular shapes (see Fig. 3-17a)), we have printed a finger of the adaptive gripper to explore its potential for medical applications. Since the polyamide is too stiff to realize such a flexible finger, we employed the

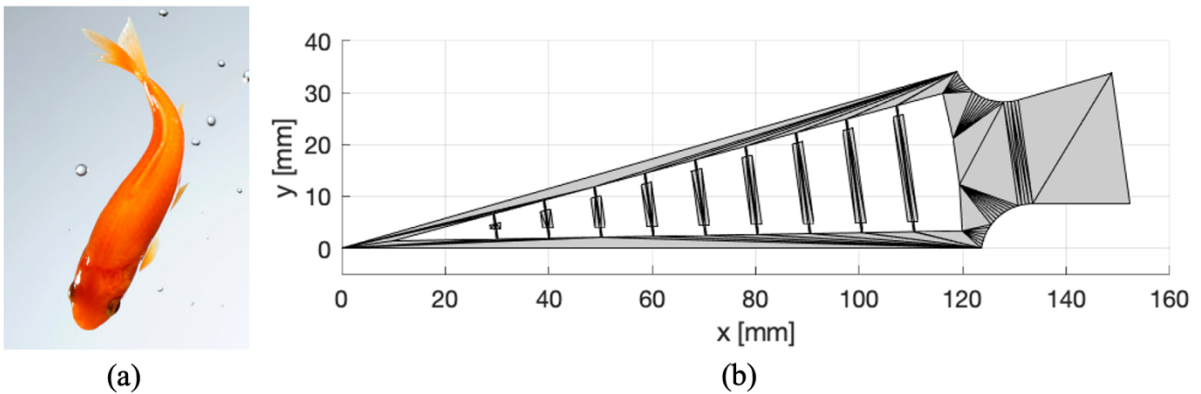(a)   (b)

Figure 3-16: Fish-fin-inspired compliant finger: a) The fish fin, b) The geometry model of the developed compliant finger (created by the SGCL language).
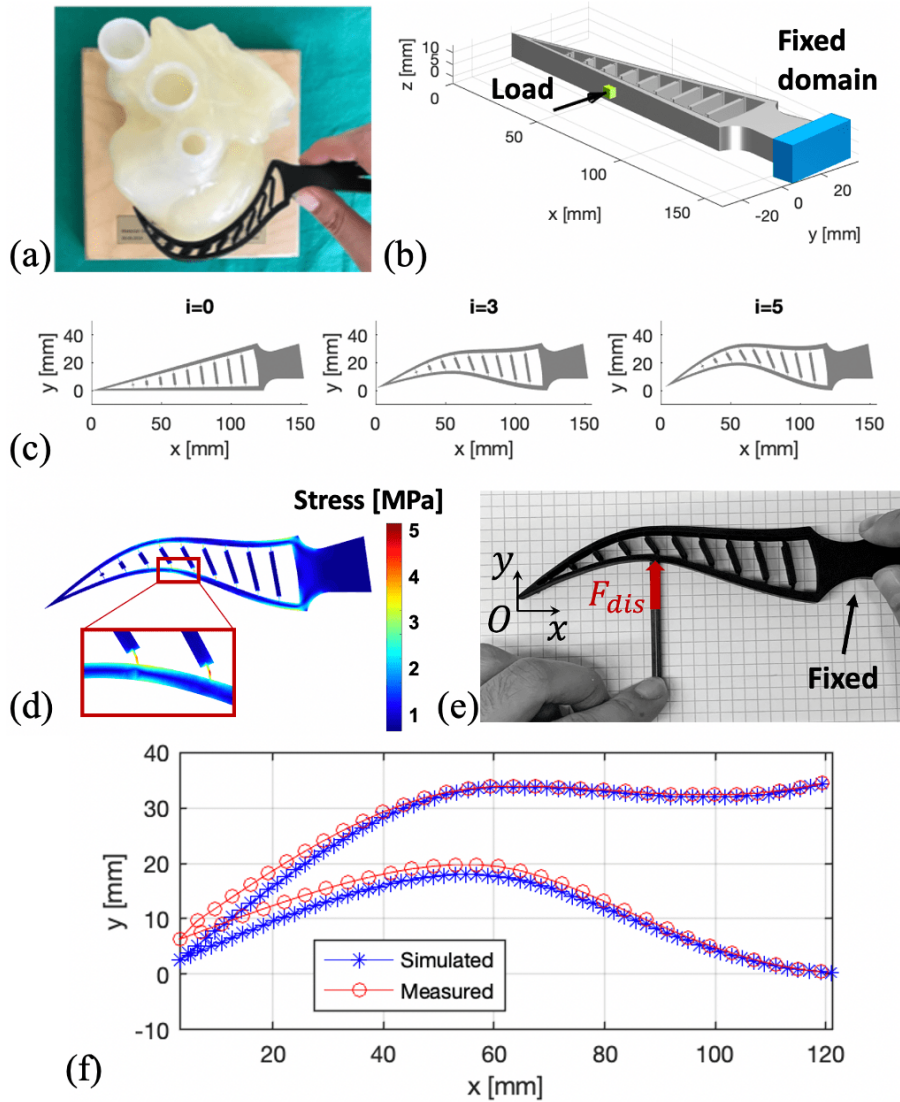
Figure 3-17: Modeling of the fish-fin-inspired adaptive finger: a) Using the adaptive finger to fit in the shape of a silicone heart model, b) A schematic diagram showing the loading cases of the adaptive finger, c) Simulation of the deformed adaptive finger. Intermediate deformed shapes (i = 0,3,5) are presented, d) The stress distribution of the deformed finger, e) Experiment for measuring the deformed shape of the SLA-printed finger, f) A comparison between the simulated (blue) and measured (red) curves of the elastic gripping jaws.

stereolithography (SLA) printer Form2 (Formlabs, USA) to fabricate the finger, using the Flexible Resin as material. The Flexible Resin is treated as linearly elastic material in this case.

Our FEM-based modeling framework was used to analyze the mechanical performance of the adaptive finger. Fig. 3-17b) shows the realized geometry of the finger. The Young's modulus of the Flexible Resin was set to 6 MPa for calculating **K**. The generated mesh was composed of 28083 nodes and 124460 elements with the maximum element size of 1 mm. In the simulation, a displacement-based force F_{dis} of 20 mm was applied on the middle part of the finger to mimic the effect of grasping an object, while the bottom of the finger was fixed. The simulation was performed with 5 incremental steps 48.6 seconds) and the calculated intermediate shapes are presented in Fig. 3-17c). From the simulated stress distribution in Fig. 3-17d), it can be noticed that the highest stress was concentrated in the largely deformed flexure hinges between the parallel stabilizers and the elastic gripping jaws of the finger. From the biological point of view, this observation is useful for analyzing the actuation principle of the fin-ray effect. Besides, we have also experimentally deformed the printed finger using a steel rod with the load

49

F_{dis} and measured the deformed curve of the elastic gripping jaws to validate the simulation results, as is shown in Fig. 3-17e). From the results reported in Fig. 3-17f) we can see that, the maximum error between the calculated deformation of the elastic jaws and the experimental result is only 2.5 mm, which also shows that the proposed incremental-load method is plausible for modeling large-displacement compliant mechanisms.

3.8 Conclusion of the Chapter

In this chapter, we presented our FEM-based framework in MATLAB to realize the mechanics modeling of compliant instruments. The modeling framework is based on a geometry modeling tool (SG Library) and contains a non-linear FEM formulation that achieves high-fidelity modeling of large displacements, tendon-driven mechanisms and contact problems. Several modeling examples have been presented to show the performance of the proposed methods in modeling 3D-printed compliant medical robots and devices. Simulation and experimental results have demonstrated the accuracy and plausibility of the presented framework. With the 2D and 3D modeling methods described in this chapter, we are able to further develop FEM-based topology optimization algorithms for the automatic synthesis of compliant medical instruments.

4 Topology Optimization Based Automatic Design

In this chapter, we focus on the second design task of the compliant medical instruments, the mechanism synthesis. The continuum-structure-based topology optimization methods are employed here to realize the synthesis. Firstly, we describe the general workflow of the proposed synthesis process. Then, the 2D-based synthesis algorithm and its implementation in MATLAB are illustrated in detail. A simple synthesis example is presented to demonstrate the performance of the 2D algorithm. Besides, we also introduce the 3D-extension of our method to explore its potential in 3D applications.

4.1 Workflow of the Design Process

As is already described in Section 1.2, the automatic design process we developed aims at simplifying the synthesis of compliant medical instruments using structural-optimization-based computer-aided algorithms. To achieve this goal, we present the general workflow of our automatic design method as in Fig. 4-1, which is applicable to both 2D and 3D design cases. Firstly, the design problem is defined by importing the initial design domain and the boundary conditions. The initial design domain doesn't need to include all design details of the instruments but should provide some basic information such as the maximum length or width. With the boundary conditions, the actuation principle or the movement constraints of the instruments can be defined. After defining the design problem, a structural optimization based synthesis process is activated, which is an iterative process. During the optimization process, a function representing the design objective is defined and FE-analysis is performed in each step to calculate the objective function. With the calculated value of the objective function, an updating scheme iteratively modifies the density of the material in the design domain until the structural optimization process converges. Finally, a post-processing algorithm is executed to realize a smooth and 3D-printable instrument.
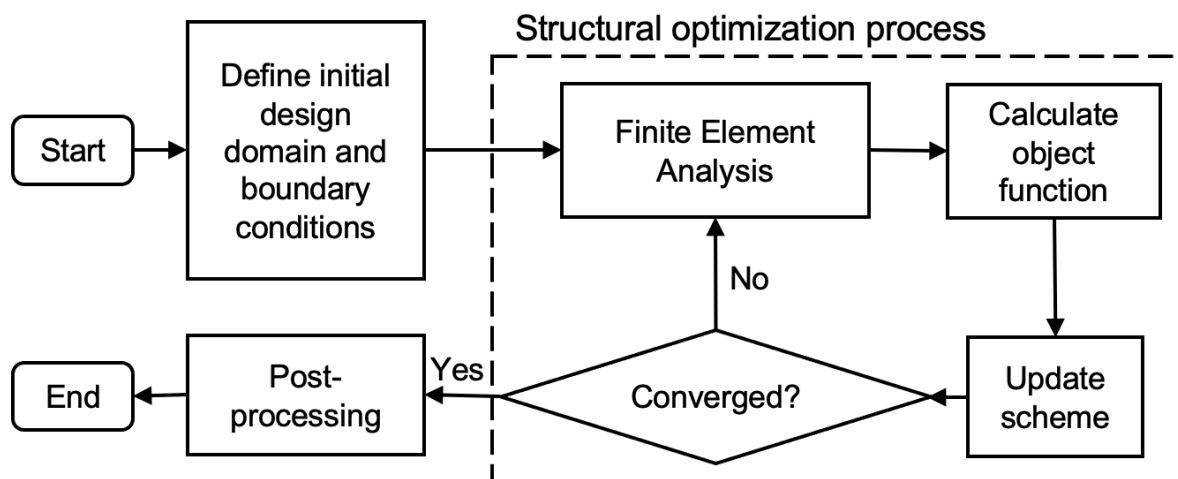

Figure 4-1: Flowchart of the developed structural optimization method for achieving automatic design of compliant medical instruments.

4.2 Automatic Design of 2D Compliant Medical Instruments

Since many medical instruments are based on 2D-structures, our design framework has provided the users the opportunity to automatically generate their design proposal in 2D and then extrude them into 3D products. In this section, we focus on the 2D automatic design algorithms.

4.2.1 2D Design Problem

As is described in Section 4.1, the first step of the automatic design is to define the design problem. Fig. 4-2 provides a schematic representation of the design problem and the possible boundary conditions that the user can define.
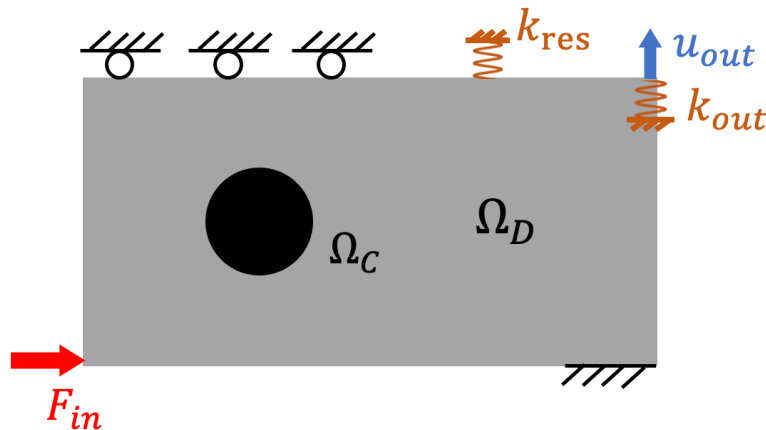

Figure 4-2: Overview of the design problem and boundary conditions of the 2D automatic design process.

Design Domain

The gray area Ω_D in Fig. 4-2 represents the design domain, from which the final design proposal derives. The densities of the material in the entire design domain are the design variable of the topology optimization process, which are initially equal to the prescribed volume fraction and will be iteratively modified to search for the best topology.

Fixation

Since the developed compliant instruments are usually mounted on certain fixed platforms in their applications, we have introduced fixation as boundary condition into the design problem. The displacements of the fixed region are constrained to 0 during the optimization process, which is similar to the setting of Dirichlet boundary condition in the FEM-based modeling. Fig. 4-3 shows the symbol for fixation.

Figure 4-3: Graphical symbol for fixation.

Sliding Mechanism

52

Since contact-aided mechanisms are frequently included contact in the design of compliant instruments, we have also provided the users the option to define sliding mechanisms in the design problem. On the other hand, as many compliant instruments have symmetrical structure, the sliding mechanism could be used as mirroring function to simplify the design problem. In this case, only half of the compliant instrument needs to be put into the design domain while the symmetrical line is defined by the sliding mechanism. Fig. 4-4 shows the symbol for the sliding mechanism.

Figure 4-4: Graphical symbol for sliding mechanism.

Load

The load is a kind of boundary condition that is defined by the user in order to actuate the compliant instrument. In our automatic design framework, we have provided two kinds of load, the force-based load and the displacement-based load. The force-based load is a Neumann boundary condition as is described in Section 3.2. In this case, the magnitude of the applied actuation force is prescribed. On the other hand, the displacement-based load is a Dirichlet boundary condition, in which the displacement of the force-applying region is predefined. The two kinds of load can be used to fulfill different actuation requirements. Fig. 4-5 shows the symbol for the load in the design problem.
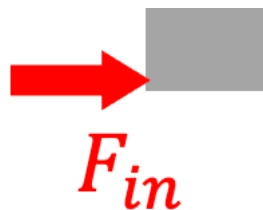

Figure 4-5: Graphical symbol for the input load.

Spring

Since the developed compliant instruments have interaction with the environment, we have introduced the spring to imitate the external resistance. The regions in the design domain, where a spring is applied, will be loaded with forces that are proportional to their displacement. In our framework, the spring is implemented as enhanced material stiffness. Fig. 4-6 shows the graphical symbol of the spring in the design problem.

Figure 4-6: Graphical symbol for the spring.

Predefined Solid Area (Geometry Constraint)

Although the fully automatic design process can greatly ease the work of the user, it is sometimes still desirable that some certain areas in the design domain, such as the end-effector and the handle of the instruments, could be preserved as specific shape during the entire optimization process. To fulfill this requirement, we have introduced the predefined solid area $\boldsymbol{\Omega_C}$ into the design problem, which is actually a set of constrained regions whose density are always 1 and cannot be modified in the design process. Fig. 4-7 shows the graphical symbol of the predefined solid area in the design problem.

Figure 4-7: Graphical symbol for the predefined solid area.

Objective Movement

Similar to the conventional rigid-link-based mechanism synthesis, the basic objective of the compliant mechanism synthesis is also to realize a mechanism that achieves specific movements with the provided actuation. Therefore, we have introduced the objective movement to define the specific motion direction that the user wants to achieve. The basic motion direction in the 2D optimization algorithm can be either in x-axis direction or in y-axis direction. Through the combination of the x- and y-axis objective movements in the objective function, the user can achieve a motion in any direction. Fig. 4-8 shows the graphical symbol of the objective movement.

Figure 4-8: Graphical symbol for the objective movement.

4.2.2 2D Topology Optimization Algorithm

In this section, we will introduce the mathematical principles behind the 2D topology optimization algorithms of the automatic design. As is mentioned before, the goal of the 2D topology optimization method, used to synthesize the compliant instruments, is to find an optimal void-solid (0 or 1) material distribution $\boldsymbol{\rho}$ of the 2D design domain which will allow to achieve the maximum displacement of the objective movement with the given input load. The optimization problem is described by the following objective function $f(\boldsymbol{\rho})$:

$$\max_{\boldsymbol{\rho}}: \quad f(\boldsymbol{\rho}) = \sum_{l=1}^{N_l} \omega_l \cdot u_{out,l} = \sum_{l=1}^{N_l} \omega_l \cdot \mathbf{L_l} \cdot \mathbf{U_l} \qquad (4\text{-}1)$$

$$\text{subject to:} \quad \mathbf{K_l U_l} = \mathbf{F_l}, \quad l = 1, \dots, N_l \qquad (4\text{-}2)$$

$$: \quad V(\boldsymbol{\rho}) = \sum_{e=1}^{N} v_e \rho_e \leq V_0 g \qquad (4\text{-}3)$$

$$: \quad E_e = E_0 \rho_e^p, \quad 0 < \rho_{min} \le \rho_e \le 1 \qquad (4\text{-}4)$$

$$: \quad \sum_{l=1}^{N_l} \omega_l = 1 \qquad (4\text{-}5)$$

Since the design of compliant instruments sometimes includes more one design task, the objective function of our optimization algorithm also has a multi-objective formulation. As in Equation (4-1), N_l design tasks can be defined with independent loads $\mathbf{F_l}$ und objective movements $u_{out,l}$. Different weighting factors ω_l are multiplied with the corresponding $u_{out,l}$ so that the multiple design tasks have different importance in the objective function. As is shown in Equation (4-5), the sum of the weighting factors ω_l is 1. $\mathbf{K_l}$ and $\mathbf{U_l}$ are the global stiffness matrix and displacement vector in the l-th design task, respectively. Since the optimization is an iterative process and the large-displacement FEA has a high computational cost, we use the linear approximation ($\mathbf{K_l U_l} = \mathbf{F_l}$) in each step to reduce the computation time. $\mathbf{L_l}$ is a sparse vector mainly composed of zeros except on the elements related to the degree of freedom (DOF) for the output displacement $u_{out,l}$ of the l-th design task.

$\boldsymbol{\rho}$ (design variable) is a list of element densities and the continuum design domain is discretized into triangular elements by the FEM method introduced in Chapter 3. E_e of the element is correlated to its density ρ_e by employing the Solid Isotropic Material with Penalization (SIMP) method (Bendsøe, 1989). The relationship between E_e and ρ_e can be illustrated by Fig. 4-9. E_0 and p are respectively the Young's modulus of the solid material and the penalization parameter, while ρ_{min} is a prescribed minimum density used to avoid singularity in the FEA process.
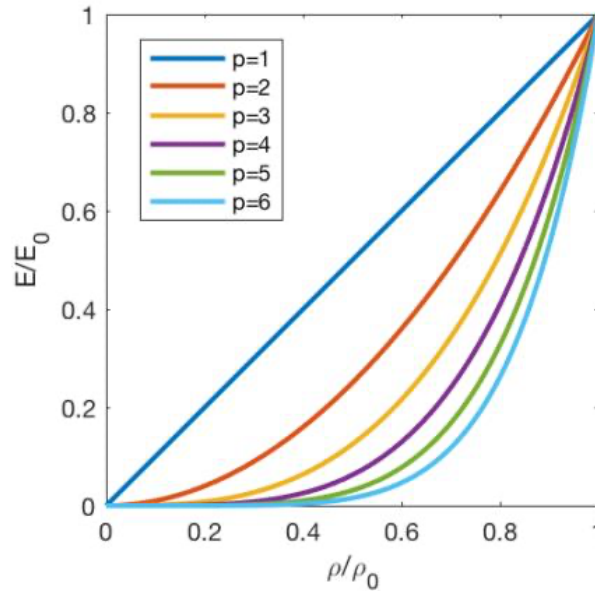

Figure 4-9: Diagram for illustrating the SIMP method ($\rho_e = \rho/\rho_0$).

v_e and v_0 are the material volume of an element and the entire initial design domain, respectively. The parameter g defines a prescribed volume fraction which controls the final volume of the design result.

The topology optimization problem in Equation (4-1) has been solved by using the Optimality-Criteria (OC) method proposed in Bendsoe and Sigmund, 2003. The updating scheme of the design variables $\boldsymbol{\rho}$ can be written as:

$$\rho_e^{new} = \begin{cases} \rho_e^-, & if \ \rho_e B_e^\eta \leq \rho_e^- \\ \rho_e B_e^\eta, & if \ \rho_e^- < \rho_e B_e^\eta < \rho_e^+ \\ \rho_e^+, & if \ \rho_e^+ \leq \rho_e B_e^\eta \end{cases} \tag{4-6}$$

$$\rho_e^- = \max(\rho_{min}, \rho_e - m) \tag{4-7}$$

$$\rho_e^+ = \min(1, \rho_e + m) \tag{4-8}$$

$$\rho_e^0 = g \quad in \ \boldsymbol{\Omega_D} \tag{4-9}$$

$$B_e = \frac{-\dfrac{\partial f}{\partial \rho_e}}{\lambda_1 \dfrac{\partial V}{\partial \rho_e}} = \frac{-\dfrac{\partial f}{\partial \rho_e}}{\lambda_1 v_e} \tag{4-10}$$

where m is a move limit of the design variable ρ_e in each step and η is a prescribed parameter for damping. ρ_e^{new} is the updated design variable. The initial design variable ρ_e^0 in the design domain $\boldsymbol{\Omega_D}$ is set to the prescribed volume fraction g. B_e in Equation (4-10) can be obtained from the optimality condition problem which considers both the objective function $f(\boldsymbol{\rho})$ and the volume constraint condition ($V(\boldsymbol{\rho}) \leq V_0 g$). The Lagrangian multiplier λ_1 can be determined by a bisection algorithm when solving the optimality condition problem (Sigmund, 2001). The sensitivity function $\frac{\partial f}{\partial \rho_e}$ with respect to ρ_e in Equation (4-10) can be formulated as:

$$\frac{\partial f}{\partial \rho_e} = \frac{\partial f}{\partial E_e} \frac{\partial E_e}{\partial \rho_e} = pE_0 \rho_e^{p-1} \frac{\partial f}{\partial E_e} \tag{4-11}$$

$$\frac{\partial f}{\partial \mathbf{E}} = -\sum_{l=1}^{N_l} \boldsymbol{\lambda}_{2,l}^{\mathbf{T}} \frac{\partial \mathbf{K_l}}{\partial \mathbf{E}} \mathbf{U_l} \tag{4-12}$$

$$\mathbf{K_l} \boldsymbol{\lambda}_{2,l} = \mathbf{L_l}, \quad l = 1, \dots, N_l \tag{4-13}$$

where $\boldsymbol{\lambda}_{2,l}$ can be calculated by solving the adjoint problem represented by the system of linear equations in Equation (4-13). $\mathbf{E}$ is the vector containing all Young's modulus E_e of the elements and $\mathbf{L_l}$ is defined as in Equation (4-1).

The design variables $\boldsymbol{\rho}$ reach a completely void-solid (0 or 1) material distribution, when the objective function f converges within a certain tolerance ε. A checkerboard prevention filter is also implemented, with mesh-size-based filter radius, to ensure a reasonable and continuous topology of the design result. In the post-processing algorithm, we extract the boundary (a CPL) of the final design proposal and use a bio-inspired shape optimization method to smooth it. After that, we extrude the 2D contour into a 3D solid in the B-rep format. The aim of this step is to realize the optimization results compatible with 3D printing technology.

4.2.3 Implementation in MATLAB

The implementation of the 2D optimization algorithm in MATLAB is based on functional programming, which is similar to the programming paradigm in Chapter 3. Algorithm 4-1 provides an overview of the implemented code.

Algorithm 4-1: Overview of the Implemented 2D Topology Optimization Algorithm in MATLAB.

1 Input of the multi-objective design problem using a MATLAB cell `fem_cell`;

2 Calculate the filtering matrix **P** using the function `TriFilter`;

3 Initialization of `opt` which contain the key data for optimization;

4 Search for the indices **s** of the elements inside the predefined solid area $\mathbf{\Omega}_C$;

5 $N_{iter} \leftarrow 0$;

6 **while** $N_{iter} < N_{max}$ && $\Delta\rho_{max} > \varepsilon$

7 $\qquad N_{iter} \leftarrow N_{iter} + 1$;

8 $\qquad$ Calculate the objective function $f_l, l = 1, \dots, N_l$ using the function `ObjFnc`;

9 $\qquad$ Calculate design sensitivities $\left(\dfrac{\partial f_l}{\partial \rho_e}\right)_{filtered} = \mathbf{P}^T \cdot \left(\dfrac{\partial f_l}{\partial E_e}\dfrac{\partial E_e}{\partial \rho_e}\right)$;

10 $\qquad$ Calculate the sum of sensitivities $\dfrac{\partial f}{\partial \rho_e} = \sum_{l=1}^{N_l} \omega_l \cdot \left(\dfrac{\partial f_l}{\partial \rho_e}\right)_{filtered}$;

11 $\qquad$ Update the design variables $\boldsymbol{\rho}$ using the function $\texttt{UpdateScheme}\left(\dfrac{\partial f}{\partial \rho_e}, \mathbf{s}\right)$;

12 $\qquad$ Calculate the maximum change $\Delta\rho_{max}$;

13 **end while**

14 Extract the boundary CPL from $\boldsymbol{\rho}$ and smooth it with the function `CPLofzsmooth`;

15 Extrude the extracted CPL with the height h to get a 3D-printable SG;

Input `fem_cell`

The input `fem_cell` is actually a package (cell) of all design tasks of the to be designed compliant instrument, where a single design task `fem_single` is a MATLAB struct created by the function `femofCPL`. Table 4-1 shows the main fields of `fem_single`. As is mentioned in Section 3.2, the fields `Node` and `Element` are created by the function `generateMesh`. The fields `Fixed`, `Load` and `Spring` are calculated using ORC method introduced in Section 3.2. In our implementation, the first column of `Fixed`, `Load` and `Spring` are the indices of the selected nodes for applying the boundary conditions, while the second and the third columns contain the specified values in x- and y-axis respectively. Another important field, the `DOut`, represents the output direction of the objective movement, where the last element of the (4×1) array is the weighting factor of the single design task.

Table 4-1: Main fields of the `fem_single` struct.

Field of `fem_single`	Description

Node	An (NNodes × 2) array of the node vertices of the mesh
NNode	Number of the nodes
Element	An (NElem × 3) array of the facet list of the meshed triangles
NElem	Number of the meshed triangular elements
E0	Elastic modulus of the solid material
nu0	Poisson's ratio of the solid material
Fixed	An (NFixed × 3) array representing the boundary conditions of fixation
Load	An (NLoad × 3) array representing the boundary conditions of load
Spring	An (NSpring × 3) array representing the boundary conditions of spring
DOut	An (4 × 1) array representing the objective movement and the corresponding weighting factor

Input Opt

Besides fem_cell, another important input is Opt, a struct containing the user-defined optimization parameters and the design variable during the optimization process. Table 4-2 shows the main fields of Opt.

Table 4-2: Main fields of the Opt struct.

Field of Opt	Description
rho	An (NElem × 1) array containing the density of each element
P	A (NElem × NElem) filtering matrix
VolFrac	Volume fraction
Tol	Tolerance of the optimization process
MaxIter	The maximum iteration number
Move	Move limit of the design variable in each step
Eta	Damping parameter in the update scheme
Cons	A CPL for determining the predefined solid area

Filtering Algorithm

A numerical problem that arises during the continuum-structure-based optimization process is the so-called checkerboard density distribution. An example for topology optimization (Bourdin, 2001) is presented in Fig. 4-10. The checkerboard problem should be avoided in order to achieve a continuous and 3D-printable structure.

Figure 4-10: A topology optimized statically stable structure without filter (reproduced from Bourdin, 2001).

From the numerical point of view, the phenomenon of checkerboard density distribution is caused by the meshed discrete finite elements. To cope with this problem, we have introduced a numerical filter to concentrate the density into connected elements during the optimization process. The proposed filter is mathematically a matrix **P** that is multiplied with the density vector $\boldsymbol{\rho}$:

$$\mathbf{P}_{l,k} = \frac{\max\left(1 - \frac{|\mathbf{x}_l^* - \mathbf{x}_k^*|}{R}, 0\right)}{\sum_{k \in S(l)}\left(1 - \frac{|\mathbf{x}_l^* - \mathbf{x}_k^*|}{R}\right)} \tag{4-14}$$

where R is a user-defined filtering radius that can be interpreted as filtering resolution. $\mathbf{x}_l^*$ and $\mathbf{x}_k^*$ are the centroid coordinates of the l-th and k-th elements. $S(l)$ denotes the set of indices of the k-th element whose centroid falls within radius R of the centroid of the l-th element, i.e., $|\mathbf{x}_l^* - \mathbf{x}_k^*| \le R$. The mechanism of the filter is to concentrate the density into connected elements within the range of radius R. The implemented function of the density filter is called `TriFilter`, whose algorithm is presented in Algorithm 4-2.

Algorithm 4-2: The Implemented Filtering Algorithm.

---

1   Input of the filter radius R and the coordinates of the mesh **x**;

2   **for** $i = 1: N_{el}$ **do**

3        $\mathbf{x}_i^* \leftarrow \frac{x_{i,1} + x_{i,2} + x_{i,3}}{3}$;

4   **end for**

5   **for** $k = 1: N_{el}$ **do**

6   **for** $l = 1: N_{el}$ **do**

7	$d_{l,k} \leftarrow \|\mathbf{x}_l^* - \mathbf{x}_k^*\|_2;$
8	**end for**
9	**end for**
10	Assemble the entire filtering matrix $\mathbf{P}$;

Postprocessing

In postprocessing, we aim to extract the 2D boundary from the final density distribution and then create a 3D-printable surface model. Since the data structure of our triangle-based density distribution is different from the pixel-based distribution, the conventional iso-contour method (Wenger, 2013) cannot be directly used to extract the 2D boundary contour. To cope with this problem, we propose a triangular-mesh-based method to extract the boundary. A shape optimization based algorithm is also incorporated to smooth the extracted boundary.

The first step of the proposed boundary-generation method is to search for the elements whose density fulfills the requirement of a solid element ($\rho_e \geq \rho_{cut} = 0.9$). The cut-off value of the boundary density ρ_{cut} is chosen as 0.9 instead of 1 because in the real synthesis examples not all densities of the solid elements can reach 1. After that, the MATLAB function `tetBoundaryFacet` is used to calculate the outer boundary contour (CPL) of the searched solid elements, which are made of the edges of the triangular elements.

However, the calculated boundary CPL is usually unsmooth since the connection of the original triangle edges often creates sharp angles which are unsuitable for 3D printing. Therefore, in the second step, we calculate the sharpness α_i of each angle and use Equation (4-18) to smooth the sharp angles. The diagram in Fig. 4-11 is used to graphically illustrate the shape optimization method.

$$\mathbf{e}_{i,i+1} = \frac{\mathbf{x}_{p_{i+1}} - \mathbf{x}_{p_i}}{\left\|\mathbf{x}_{p_{i+1}} - \mathbf{x}_{p_i}\right\|_2} \tag{4-15}$$

$$\mathbf{n}_i = \frac{\mathbf{e}_{i-1,i} + \mathbf{e}_{i,i+1}}{\left\|\mathbf{e}_{i-1,i} + \mathbf{e}_{i,i+1}\right\|_2} \tag{4-16}$$

$$\alpha_i = \frac{\mathbf{e}_{i-1,i} \cdot \mathbf{n}_i + \mathbf{e}_{i,i+1} \cdot \mathbf{n}_i}{2} \tag{4-17}$$
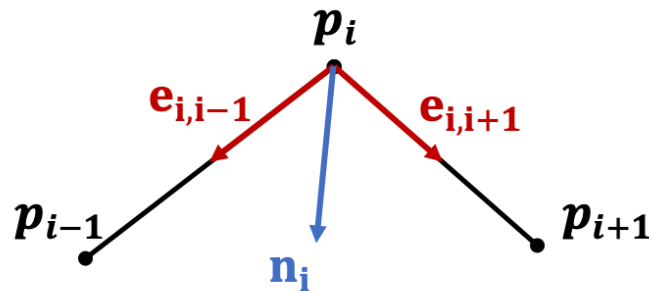

Figure 4-11: Schematic diagram for illustrating the shape optimization method for smoothing the 2D boundary contour.

$$\Delta \mathbf{x}_{\mathbf{p_i}} = c \cdot \alpha_i \cdot \mathbf{n_i} \tag{4-18}$$

$\mathbf{e_{i,i+1}}$ is the normalized tangential vector from $\boldsymbol{p_i}$ to $\boldsymbol{p_{i+1}}$, while $\mathbf{n_i}$ indicates the normalized normal vector of node $\boldsymbol{p_i}$. $\Delta \mathbf{x}_{\mathbf{p_i}}$ shows the iterative movement of the coordinate of $\boldsymbol{p_i}$ in the shape optimization to modify the 2D contour, where c is a scaling factor to control the optimization speed. The implemented algorithm of the shape optimization method is presented in Algorithm 4-3. Since the proposed optimization process is similar to the biological self-optimization process, it can also be seen as a bio-inspired method.

Algorithm 4-3: The Bio-Inspired Shape optimization Algorithm for Smoothing the 2D Boundary Contour.

1   Initialization of the extracted CPL using the function `tetBoundaryFacet`;

2   Calculate the coordinates $\left\{ \mathbf{x}_{\mathbf{p_i}} \right\}_{i=1,\dots,N_b}$ of the nodes on the boundary;

3      $\alpha_{mean} \leftarrow 1$;

4   **while** $\alpha_{mean} > 0.5$ **do**

5   **for** $i = 1 : N_b$ **do**

6          $\mathbf{e_{i,i+1}} \leftarrow \dfrac{\mathbf{x_{p_{i+1}}} - \mathbf{x_{p_i}}}{\left\| \mathbf{x_{p_{i+1}}} - \mathbf{x_{p_i}} \right\|_2}$;

7          $\mathbf{n_i} \leftarrow \dfrac{\mathbf{e_{i-1,i}} + \mathbf{e_{i,i+1}}}{\left\| \mathbf{e_{i-1,i}} + \mathbf{e_{i,i+1}} \right\|_2}$;

8          $\alpha_i \leftarrow \dfrac{\mathbf{e_{i-1,i}} \cdot \mathbf{n_i} + \mathbf{e_{i,i+1}} \cdot \mathbf{n_i}}{2}$;

9          $\Delta \mathbf{x}_{\mathbf{p_i}} \leftarrow c \cdot \alpha_i \cdot \mathbf{n_i}$;

10   **end for**

11         $\mathbf{x_p} \leftarrow \mathbf{x_p} + \Delta \mathbf{x_p}$;

12         $\alpha_{mean} = \dfrac{\sum_{i=1}^{N_b} \alpha_i}{N_b}$;

13   **end while**

14   Create a new CPL from the optimized $\mathbf{x_p}$;

The last step of the postprocessing is to extrude the smoothed boundary into a 3D solid with a user-defined height. Herein, we employ the function `SGofCPLz` (Lueth, 2013) from the SG Library to realize the extrusion.

4.2.4   2D Design Example

To demonstrate the performance of the proposed 2D topology optimization method, an example is presented in this section for synthesizing a basic compliant gripper mechanism. Firstly, a CPL, as is shown in Fig. 4-12, is created by the SGCL language to define the design domain. Then, other design inputs are provided as is shown in Fig. 4-13. In this example, only one design task is included in the design problem, which is to maximize the objective movement u_{out}. The left side of the design domain is fixed, while a sliding mechanism is placed on the upper side of the design domain to realize a symmetrical design. A displacement-based load f_{in} is applied to actuate the compliant mechanism. The main design parameters are listed in Table 4-3. The filtering radius is chosen equal to the size of the mesh element to achieve a continuous geometry for the synthesis result.
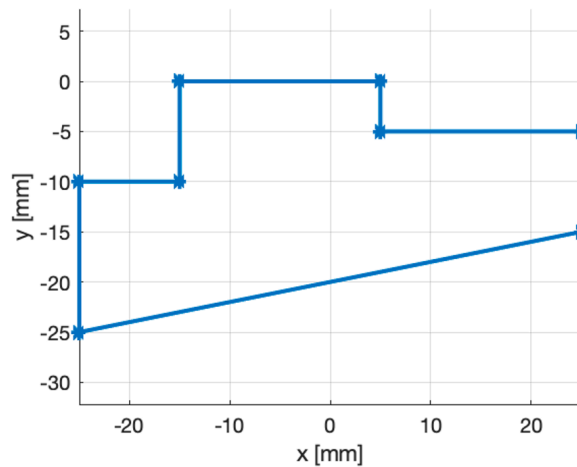

```
>> SGofCPLcommand(['b 50 25,move 0 −12.5,enter,b 20,'...
      'move −25,−,enter,b 40 10,move 25,−,enter,'...
      'b 50 10 50,move 0 −20,−'])
```

Figure 4-12: A CPL for defining the design domain.
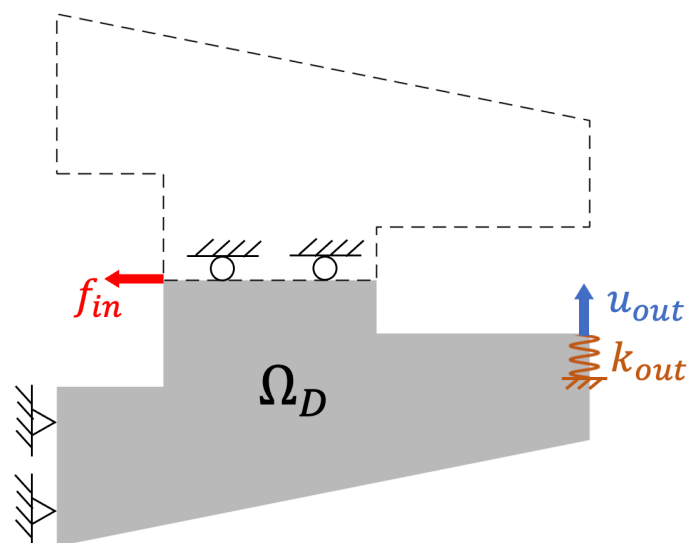
Figure 4-13: Schematic representation of the design problem.

Table 4-3: The main design parameters for the synthesis of the compliant gripper.

Parameter	Symbol	Value
Actuation load	f_{in}	$3\ mm$
Spring constant at the output port	k_{out}	$0.75\ N/mm$
Elastic modulus	E_0	$1700\ MPa$
Poisson's ratio	v	0.3
Maximum element size in FEA	h	$0.25\ mm$
Volume fraction	g	0.3
Penalization number	p	3
Damping factor	η	0.3
Move limit	m	0.05
Convergence tolerance	ε	0.01
Filtering radius	R	$0.25\ mm$

Fig. 4-14 and Fig. 4-15 present respectively the value of the objective function and the evolutionary process of the density distribution during the automatic shape synthesis. The topology optimization process reached its convergence at the 164th iteration. The entire process took 41.32 s. As can be seen in Fig. 4-14, the maximum output displacement u_{out} (4.97 mm) was reached at the last iteration. It can also be noticed that the value of the objective function has remained the same at the latter part of the optimization process.
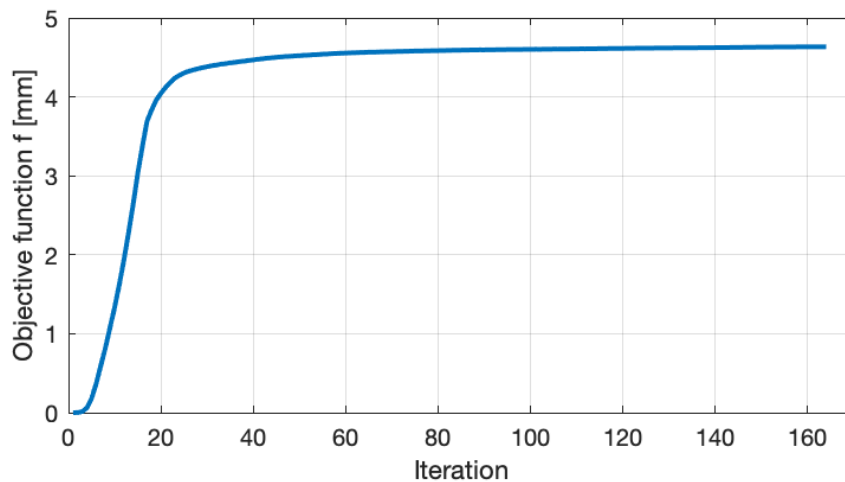
Figure 4-14: The trend of objective function during the optimization process.
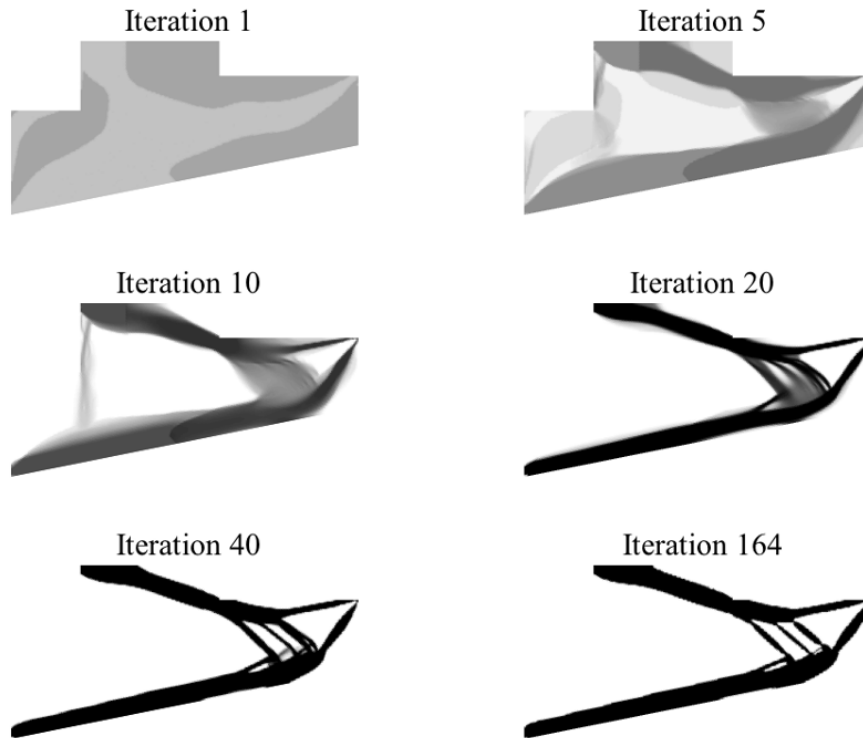
Figure 4-15: The evolution process during some iterations. The final topology of the compliant gripper is reached at the 164th iteration.

In order to show the performance of the filtering algorithm, we have also run the optimization process again without the filter. The achieved synthesis result is shown in Fig. 4-16. Although the process converged within much fewer iterations (53 iterations) and shorter calculation time (13.44 s), the optimized density distribution contained many checkerboard area, which is problematic for generating the 3D-printable model and fabricating the prototype.

As is mentioned in Section 4.2.3, the boundary CPL of the optimized result is derived using the triangular-mesh-based method. The extracted CPL and its duplicate are merged to get the full gripper. FE-analysis is also performed to evaluate the performance of the synthesized compliant gripper. The 2D large-displacement FEM method from Section 3.4 is used and Fig. 4 -17 shows the simulated stress distribution. In the FE-analysis, the left two bottoms of the realized
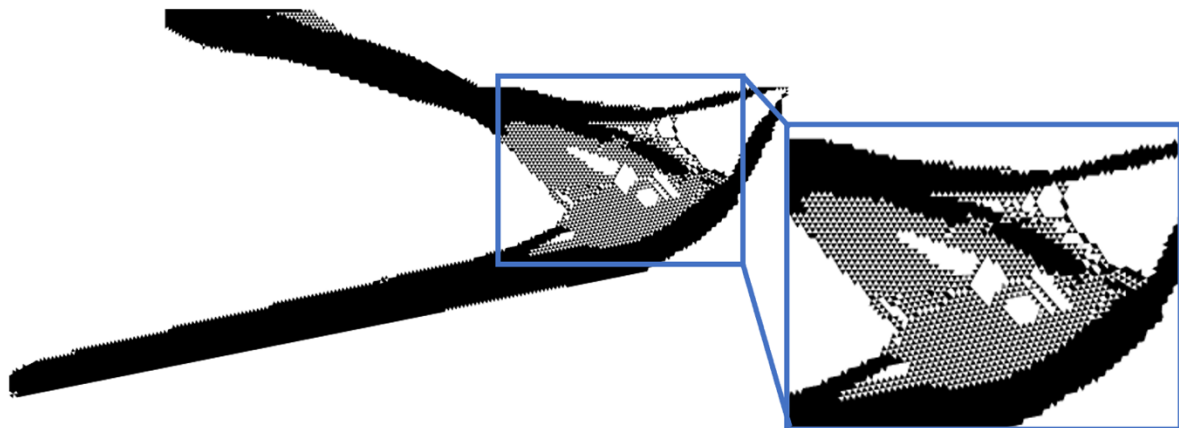
Figure 4-16: The unfiltered optimized density distribution, where the checkerboard phenomenon has emerged.

mechanism are fixed and a displacement-based load f_{in} of 3 mm is applied on the middle part, as is shown in Fig. 4-17. From the stress distribution we can see the highly stressed area where the largest deformations occur.
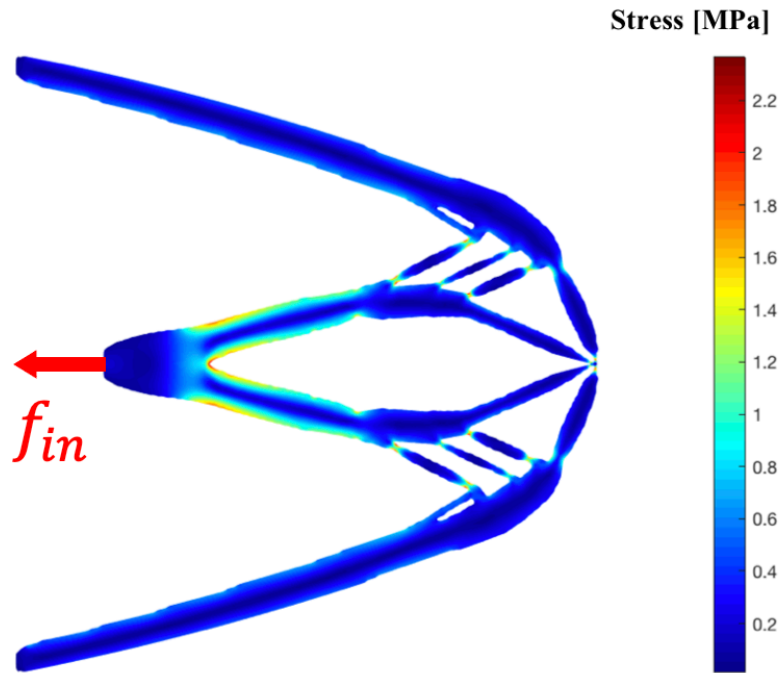

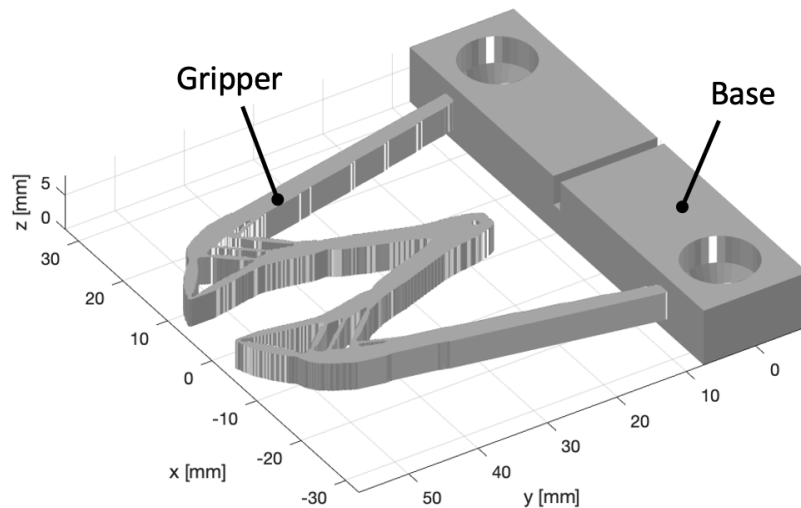**Figure 4-17: Stress distribution in the compliant gripper under the actuation force f_{in}.**

Figure 4-18: 3D surface model of the extruded gripper and the connected base.

The optimized 2D boundary contour of the gripper is then extruded into a 3D surface model with the height of 5 mm. A base is also created to fix the gripper, as is shown in Fig. 4-18. The compliant gripper is selective laser sintered with polyamide (PA2200) to test its grasping performance. As is presented in Fig. 4-19, the printed compliant gripper can be successfully closed with the dragging force along the symmetrical line, which has verified the feasibility of our automatic design framework.
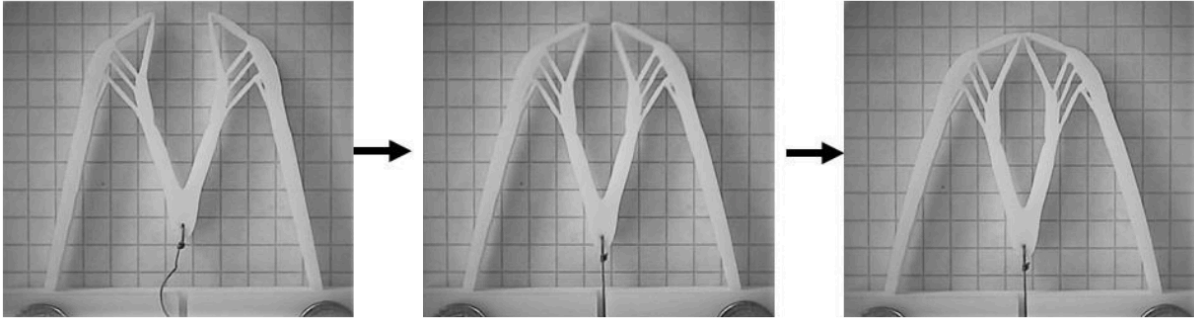
Figure 4-19: Test of the performance of the automatically synthesized compliant gripper. A cable is used to actuate the 3D-printed gripper.

4.3 Extension to 3D Topology Optimization

To deeply explore the 3D potential of the automatic design method, we have also integrated the fully 3D topology optimization algorithm into our design framework. In this section, we will describe the 3D-based algorithm in detail. Several synthesis examples are also presented to demonstrate its performance of synthesizing 3D compliant instruments.

4.3.1 3D Design Problem

The design domain of the 3D problem is a cuboid defined in a Cartesian coordinate system which can be divided into $n_x \times n_y \times n_z$ cubic elements with adjustable edge length l_e, as is shown in Fig. 4-20. The voxel-model-based design domain is used instead of a tetrahedral-mesh-based solid (SG) of any shape because the voxel-model-based mesh is more efficient in 3D FE-calculation and postprocessing. The white cubes in Fig. 4-20 are an example representing the user-defined geometrical constraints that should be kept void throughout the entire design process. The user can define the geometrical constraints by using a 3D surface model created by the SG Library that has an overlapping region with the prescribed design domain, which follows the principle of the ORC method.
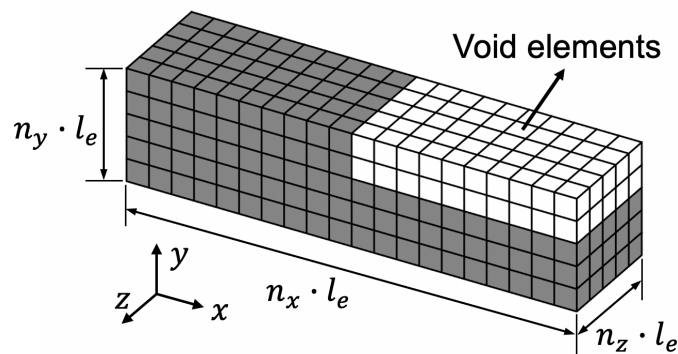

Figure 4-20: The voxel-model-based design domain for the general 3D design problem.

Fig. 4-21 is a schematic diagram illustrating the design problem for synthesizing a 3D compliant gripper, which is used as an example to illustrate a general 3D design problem. Basically, the input settings are similar to the 2D version. Since some fully 3D instruments have symmetric structures, our design framework allows the user to specify the top or side surface of the design domain as symmetric surface so that the computational cost for synthesizing the entire gripper structure can be greatly reduced. The user can define the actuation mechanism by specifying the actuation force F_a and the fixation area on the design domain, as is shown in Fig. 4-21. In order to realize the grasping movement of the compliant gripper, a grasping point $\boldsymbol{P_{out}}$ should

also be chosen from the nodes of the design domain whose displacement u_{out} is to be maximized in the optimization process.
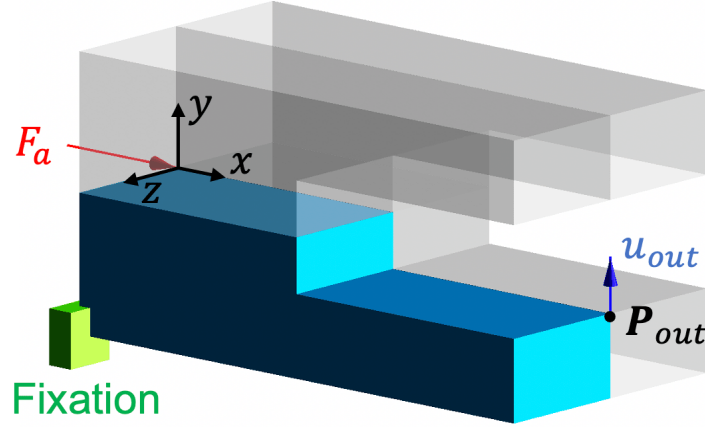

Figure 4-21: Schematic representation of the boundary conditions and design objective in the 3D design problem. In this case, an example of compliant gripper is presented for illustration purpose.

4.3.2   Modified Optimization Algorithm and Implementation in MATLAB

The aim of the proposed 3D topology optimization method is to find an almost solid-void (1-0) density distribution $\boldsymbol{\rho}$ (a voxel model) of the elements in the 3D design domain to achieve the maximum of the multi-design-objective. Basically, the 3D optimization algorithm is similar to the 2D version (see Equation from (4-1) to (4-13)). A major difference is that the 3D design variable $\boldsymbol{\rho}$ is a voxel-model-based MATLAB array whose size is $n_x \times n_y \times n_z$. Some important physical values, such as the displacement vector $\mathbf{U_l}$, are also extended with the z-axis components to adapt to the 3D formulation.

Since the voxel model is used to discretize the 3D design domain, the numerical filter described in Section 4.2.3 should also be modified. Herein, we have introduced a non-linear grayscale filter (Groenwold and Etman, 2008) to maintain the numerical stability of the 3D-based algorithm. The update scheme in Equation (4-6) is modified as follows:

$$\rho_e^{new} = \begin{cases} \rho_e^-, & if \ \rho_e B_e^\eta \le \rho_e^- \\ \left(\rho_e B_e^\eta\right)^q, & if \ \rho_e^- < \rho_e B_e^\eta < \rho_e^+ \\ \rho_e^+, & if \ \rho_e^+ \le \rho_e B_e^\eta \end{cases} \tag{4-18}$$

where q is the non-linear factor. In out implementation, q is set to 2 which is suitable for the SIMP-based topology optimization.

Similar to the 2D-based algorithm, $\boldsymbol{\rho}$ reach an almost solid-void (1-0) distribution when the optimization process converges. After that, we duplicate $\boldsymbol{\rho}$ symmetrically according to the symmetric planes to get the entire voxel model $\boldsymbol{\rho_{full}}$ of the compliant instrument. To extract a 3D surface model from $\boldsymbol{\rho_{full}}$, the marching cubes algorithm proposed in Lorensen and Cline, 1987 is used. As can be seen in Fig. 4-22a) and Fig. 4-22b), the extracted 3D surface model is sometimes split into several independent objects after using the marching cubes method. However, the broken parts should be treated as thin flexure hinges in the real design of soft grippers. The reason of this problem is that some neighboring solid elements in $\boldsymbol{\rho_{full}}$ are only connected by one single-shared edge (see Fig. 4-22a)) and the marching cubes algorithm considers those elements as unconnected when generating a surface model (see Fig. 4-22b)). To cope with this
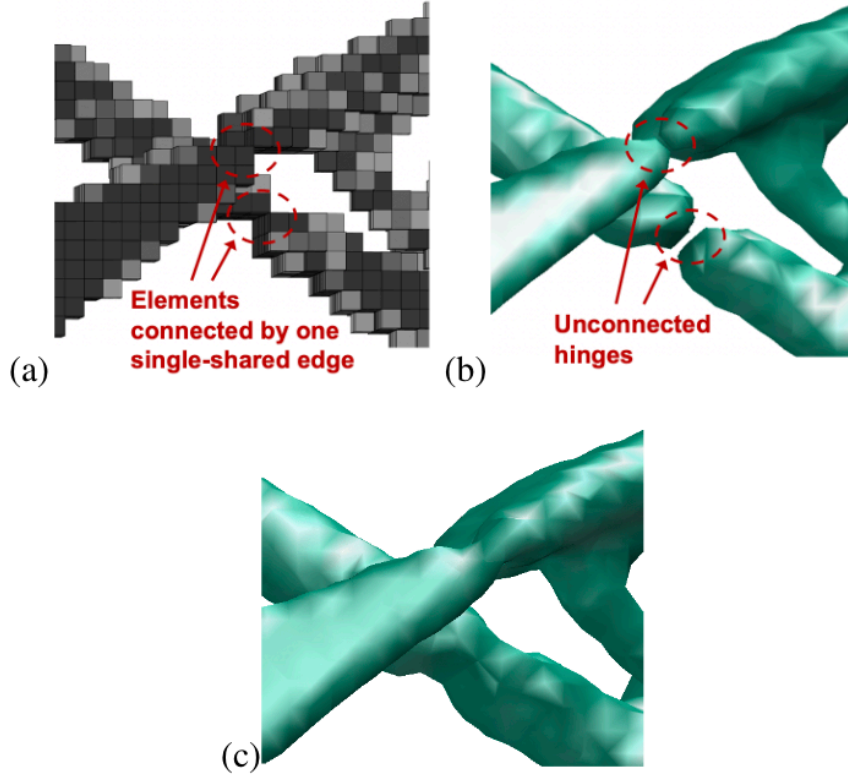
Figure 4-22: The problem of unconnected hinges in the extracted 3D surface model: a) The optimized voxel model of ρ which contains single-shared edges, b) The extracted surface model with unconnected hinges, c) The repaired surface model.

problem, we have developed an algorithm (see Algorithm 4-4) in our framework to repair the voxel model so that the missed flexure hinges can be regenerated in the extracted 3D surface model. The basic idea of the repair algorithm is to go through all connecting edges in the provided voxel model $\boldsymbol{\rho_{full}}$ using the neighborhood-element matrix **M** and detect all single-shared edges. The two void elements ($\rho_e < \rho_{iso}$) beside the detected single-shared edge are then thickened so that a thin flexure hinge will be generated at the detected place in the surface model. Fig. 4-22c) shows that the unconnected parts of the extracted surface model can be successfully reunited using the proposed repair algorithm.

Algorithm 4-4: Algorithm of Repairing the Unconnected Hinges in the Voxel Model

---

1   Initialization of $\boldsymbol{\rho_{full}}$ and ρ_{iso};

2   Determine the size of $\boldsymbol{\rho_{full}}$: $\{n_x, n_y, n_z\}$;

3   **for** $i = 1{:}n_z$ **do**

4   **for** j= $1{:}n_y - 1$ **do**

5   **for** $k = 1{:}n_x - 1$ **do**

6                $\mathbf{M} \leftarrow \boldsymbol{\rho_{full}}(k{:}k+1, j{:}j+1, i)$;

7   Determine the number of the solid elements n_{solid} in **M**;

8	**if** $n_{solid} = 2$ && $	\det(M)	> 0.5$ **then**
9	**if** $M(1,1) > \rho_{iso}$ **then**
10	$\rho_{full}(k+1,j,i) \leftarrow \rho_{iso} + 0.01;$
11	$\rho_{full}(k,j+1,i) \leftarrow \rho_{iso} + 0.01;$
12	**else**
13	$\rho_{full}(k,j,i) \leftarrow \rho_{iso} + 0.01;$
14	$\rho_{full}(k+1,j+1,i) \leftarrow \rho_{iso} + 0.01;$
15	**end if**
16	**end if**
17	**end for**
18	**end for**
19	**end for**

4.3.3   3D Design Examples

In this section, we demonstrated the performance of the proposed 3D topology optimization method by synthesizing two compliant grippers with different jaw shapes.

Soft Gripper With Flat Gripping Jaws

The first design example is to synthesize a soft gripper with flat gripping jaws. As is shown in Fig. 4-23a), the design domain was comprised of $80 \times 20 \times 20$ cubic elements with $l_e = 0.5\ mm$. The flat jaw shape was defined by a geometrical constraint (white region in Fig. 4-23a), which remained void during the entire synthesis process. Fig. 4-23b) shows the loading cases and optimization objective in the first design problem. Since the gripper is supposed to
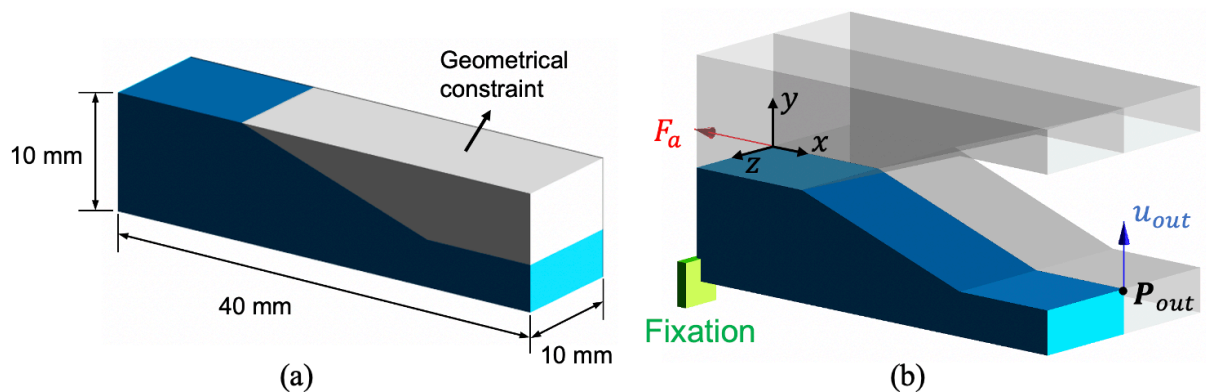
Figure 4-23: Design problem of the soft gripper with flat gripping jaws: a) The design domain. The blue and white region represent the active and constrained design domain respectively, b) Schematic representation of the boundary conditions and design objective in the design problem.

have a symmetric structure, we defined the surface on the x-y plane and the x-z plane as symmetric surfaces and all the boundary conditions were also mirrored. A pulling force F_a of 0.2 N was applied on the design domain to actuate the soft gripper while the single design objective was to maximize the displacement of $\boldsymbol{P_{out}}$. The distance between $\boldsymbol{P_{out}}$ and the bottom of the design domain was 4 mm. Since the soft gripper is SLA-printed with the Durable Resin (Formlabs, USA), a linearly elastic material, the Young's modulus E_0 and Poisson's ratio ν were set to 1000 MPa and 0.3 for calculating $\mathbf{K_e}$. The volume fraction g was set to 0.15 to constrain the volume of the synthesized gripper. The automatic synthesis process is presented in Fig. 4-24. As is shown in Fig. 4-24a), the objective function u_{out} converged at the 98th iteration, reaching its maximum of 3.92 mm. The evolution process of $\boldsymbol{\rho}$ during some iterations is also graphically illustrated in Fig. 4-24b). It can be noticed that the most changes of u_{out} and $\boldsymbol{\rho}$ have already taken place in the first 20 iterations while the rest calculations were performed to achieve convergence. Fig. 4-25 shows the extracted 3D surface model of the full gripper after post-processing. It can be seen that the obtained model is a combination of a 2D topology optimized gripper structure (from the z-axis view) and an optimized statically stable structure (from the y-axis view), which is advantageous over the purely 2D topology optimized grippers. The entire design process took 450.76 seconds.
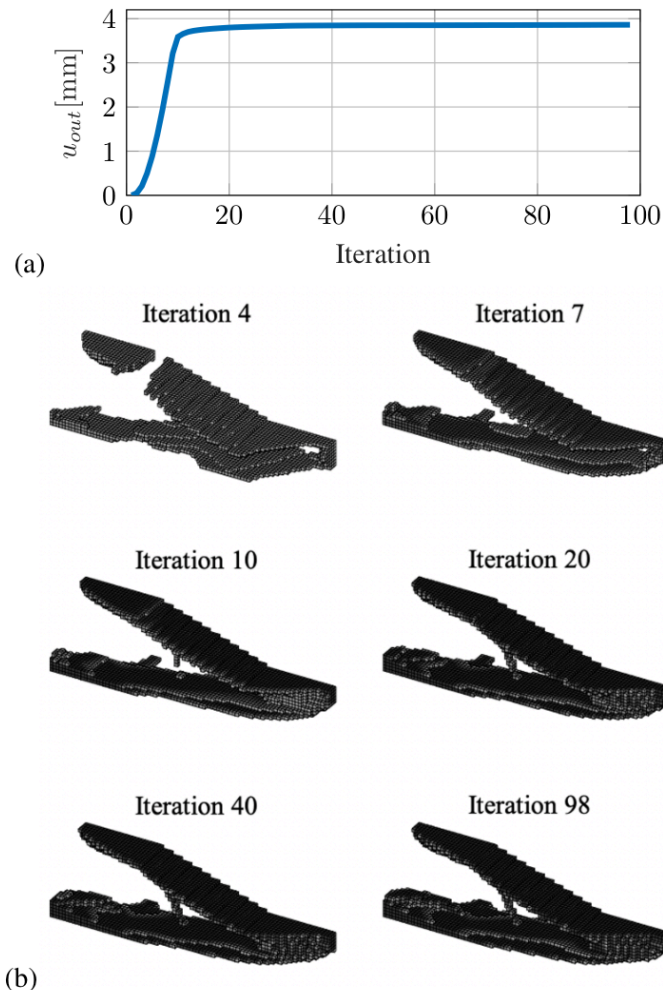

(a)

(b)

Figure 4-24: Automatic synthesis of the soft gripper with flat gripping jaws: a) The value of u_{out} in the optimization process. u_{out} converged at the 98th iteration, b) The distribution of ρ in some iterations. The elements with $\rho_e > 0.5$ are plotted and the black color represents solid element.
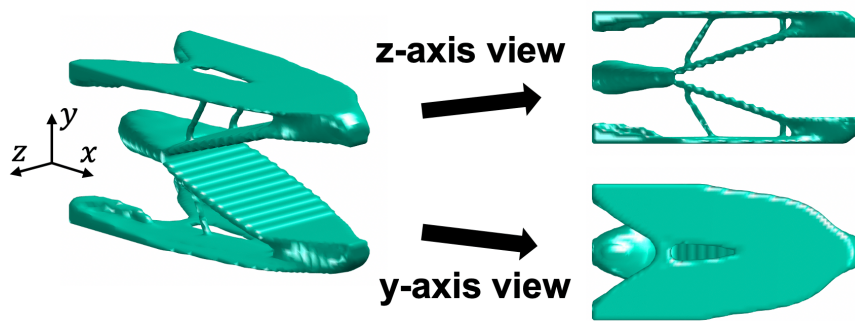
Figure 4-25: The post-processed 3D surface model of the synthesized soft gripper.

Soft Gripper With 3D Concave Gripping Jaws

In the second design example, we have defined another kind of geometrical constraint in the design domain (see Fig. 4-26) in order to achieve 3D concave gripper jaws in the synthesis result. The concave shape of the gripper jaw is intended to solve the slipping problem when grasping smooth convex objects. The corresponding design problem is illustrated in Fig. 4-26. Similar to the first example, the design domain was also based on a $80 \times 20 \times 20$ voxel model with the element length of $l_e = 0.5\ mm$. The distance between the objective point $\boldsymbol{P_{out}}$ and the bottom of the design domain was 3 mm while the volume constraint g was set to 0.12. All the other design settings, such as the material properties and the actuation force, were the same as in the first example. All the boundary conditions were also mirrored according to the symmetric surfaces.
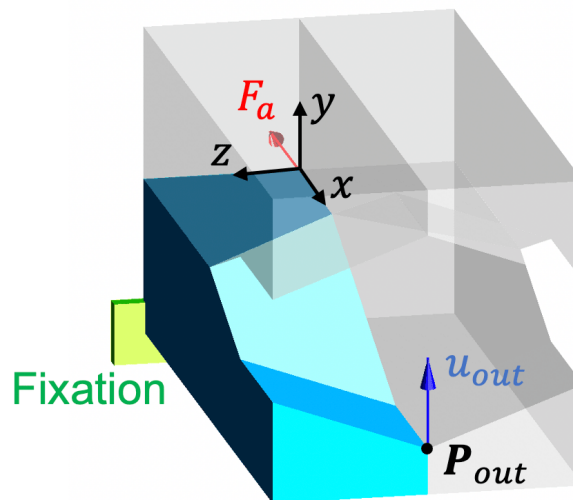
Figure 4-26: The design domain and the schematic representation of the second design problem.

Fig. 4-27a) shows the trend of u_{out} during the optimization process. It can be seen that synthesis process converged at the 94th iteration, reaching a maximum u_{out} of 4.05 mm. From the trend of u_{out} and the evolution process of the design variable $\boldsymbol{\rho}$ (see Fig. 4-27b), we can see that the main structure of the gripper has already emerged in the 30th iteration. Fig. 4-28 presents the full model of the soft gripper after post-processing. It can be noticed that the 3D concave gripping jaws and the statically stable structure (from the y-axis view) are successfully incorporated into the final design. The entire design process, including the optimization and post-processing, took 464.18 seconds.
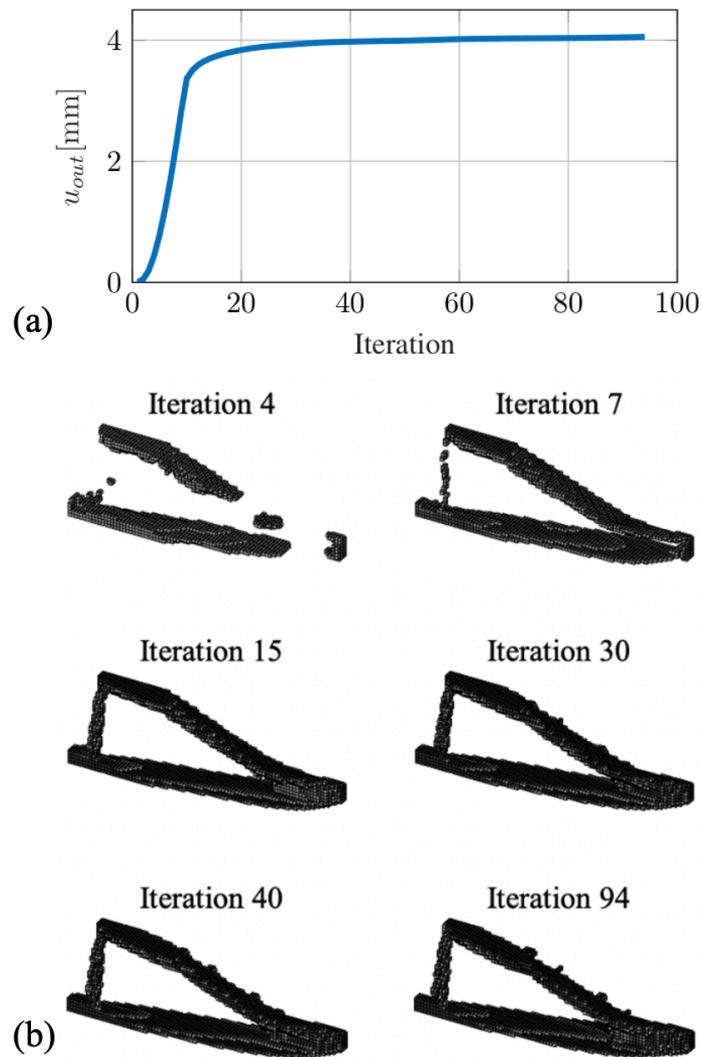
Figure 4-27: Automatic synthesis of the soft gripper with 3D concave gripping jaws: a) The value of u_{out} in the optimization process. u_{out} converged at the 94th iteration, b) The distribution of ρ in some iterations. The elements with $\rho_e > 0.5$ are plotted and the black color represents solid element.
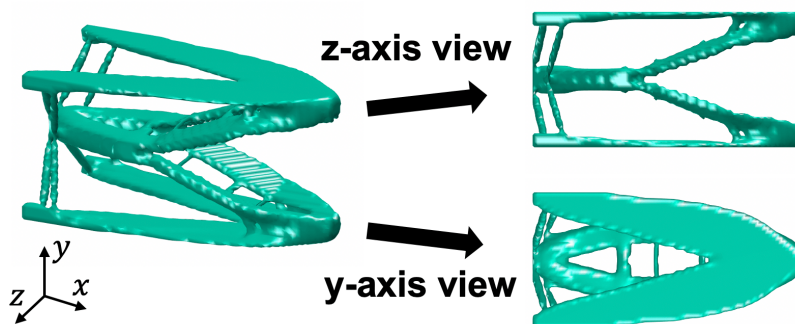

Figure 4-28: The post-processed 3D surface model of the synthesized soft gripper.

Evaluation of the Grasping Performance

Experimental tests were conducted to evaluate the adaptable grasping performance of the two synthesized soft grippers. The experimental setup is presented in Fig. 4-29. The soft grippers and the fixation base were both SLA-printed, using the Durable Resin and the White Resin
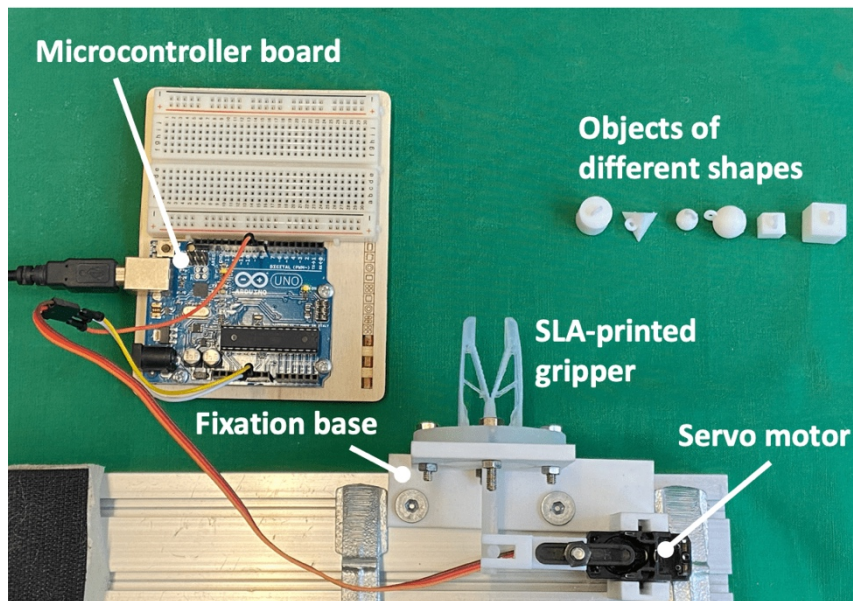
Figure 4-29: Experimental setup. The SLA-printed soft gripper was pulled by a servo motor to grasp rigid objects of different shapes.

(Formlabs, USA) respectively. A servo motor (Fischertechnik, Germany), controlled by a microcontroller (Arduino UNO), was used to actuate the gripper. The generated pulling force was 1 N at zero speed, which is sufficient to achieve the full closure of the gripper jaws. In the experiment, rigid objects of different shapes (sphere, cube, tetrahedron, cylinder) were grasped by the two soft grippers.

Experiment results are presented in Fig. 4-30. We can see that, in all grasping tests, the gripping jaws were deformed adaptively so that the objects with different shapes can be held tightly, which verified the feasibility of the 3D-based design framework. It can also be noticed that the gripper from the second synthesis example can enclose the convex objects better than the first one due to the wrapping effect of the incorporated 3D concave jaws. On the other hand, as can be observed in Fig. 4-30, some fatigue cracks have emerged in the branch-like thin structures during the experiment. After performing FE-analysis on the two soft grippers (see Fig. 4-31), we can see that an important cause of the cracks is the large stress concentrated in the thin structure. As these cracks had little influence on the adaptable grasping performance of the soft grippers, the branch-like thin structures in Fig. 4-30 could be considered as auxiliary structures that emerged in the optimization result but contributed little to the design objective.
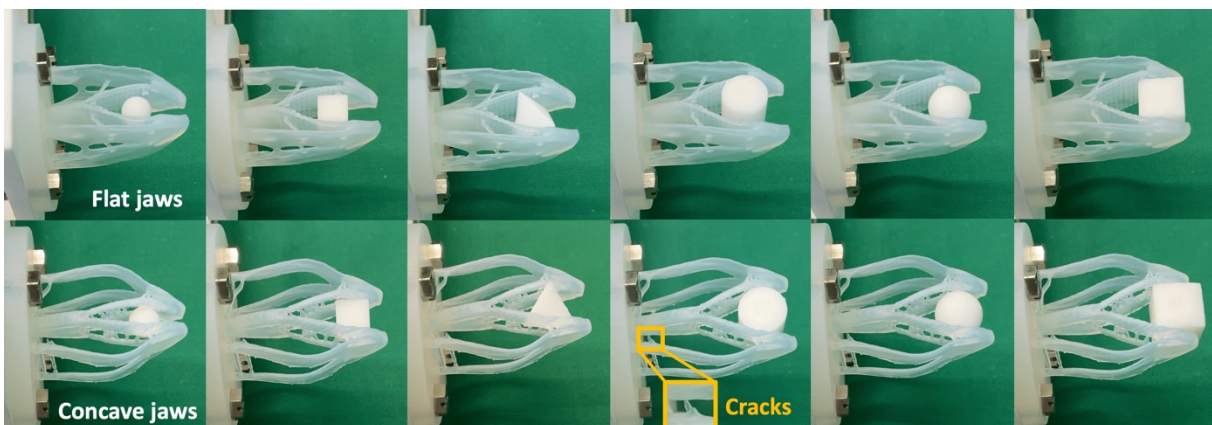

Figure 4-30: Experimental tests showing the adaptable grasping performance of the two soft grippers. The grasped objects from left to right: small sphere, small cube, tetrahedron, cylinder, big sphere, big cube.
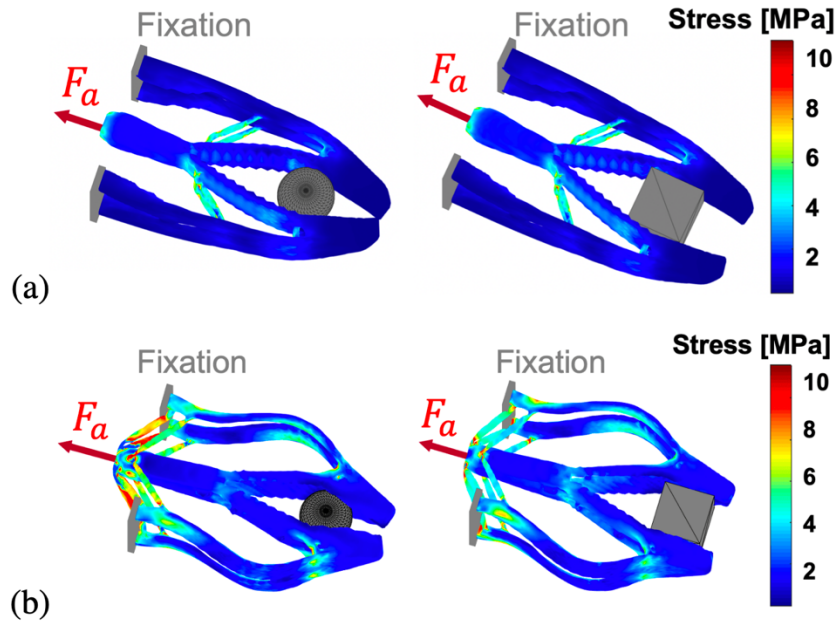
Figure 4-31: FEM-simulated stress distribution of the synthesized soft grippers grasping a spherical and a cubic object: a) Simulation of the soft gripper with flat gripping jaws ($F_a = 1\ N$), b) Simulation of the soft gripper with 3D concave gripping jaws ($F_a = 1\ N$).

4.4 Conclusion of the Chapter

In this chapter, we described the topology optimization algorithms that we used in our framework to realize the automatic synthesis of compliant instruments. The 2D and 3D version of the optimization algorithms were presented, with detailed illustration of the mathematical principle and implementation in MATLAB. Besides, several synthesis examples were also presented, which verified the feasibility of our topology optimization based design framework.

5 Analysis of Design Parameters

As is mentioned in Chapter 1, the automatic design framework can make the synthesis of compliant instruments much easier, since the design concepts are improved and refined by computer algorithms instead of the engineers themselves. Nevertheless, as the users still have to define some design parameters before executing the optimization algorithm, it is worthwhile to analyze the influence of such parameters on the synthesis result, in order to provide the users a guideline for choosing parameter values. In this chapter, the parameters, such as the mesh size, the volume faction and the geometrical constraints, will be analyzed. Besides, the difference of the 2D and 3D topology optimized compliant instruments will also be investigated.

5.1 Design Results Based on 2D and 3D Algorithms

To compare the design results of 2D and 3D based optimization methods, we solved the 2D design problem described in Fig. 4-13 using the 3D topology optimization algorithm. Fig. 5-1 presents the extended 3D design problem, where the blue domain indicates the 3D design domain excluding the void geometrical constraints. The geometry of the design domain is created by extruding the 2D design domain in Fig. 4-12 with the height $n_z \cdot l_e$. The red and blue arrow represent the actuation force and the objective movement, respectively. The upper (x-z plane) and right (x-y plane) side of the design domain are mirrored to realize a symmetrical gripper. In the 3D synthesis, l_e was set to 1 mm while n_z was chosen as 10 and 20 in different cases for comparison.
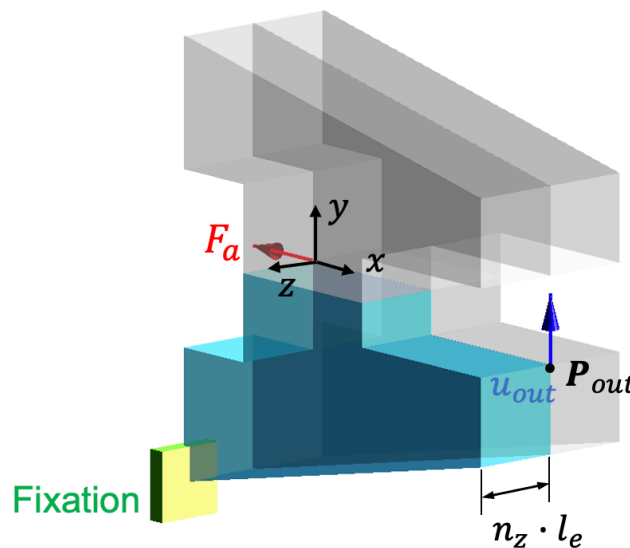

Figure 5-1: A 3D design problem for synthesizing a compliant gripper.

Case 1: $n_z = 10$

In the first synthesis case ($n_z = 10, l_e = 1mm$), $F_a = 3mm$ was displacement-based, as in the 2D example. The volume fraction was also set to 0.3. From the evolution process of the density distribution in Fig. 5-2, we can see that the optimization process converged at the 56th iteration. The post-processed full gripper is presented in Fig 5-3. It can be noticed that the shape of the realized gripper in z-axis view is similar to the 2D optimized gripper (see Fig. 4-18), while the gripper feet became statically more stable (observed from the y-axis view). The entire design process took 133.4 seconds.
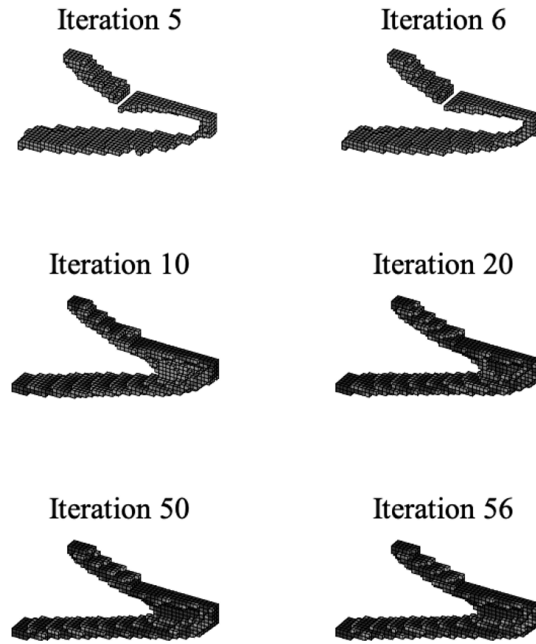
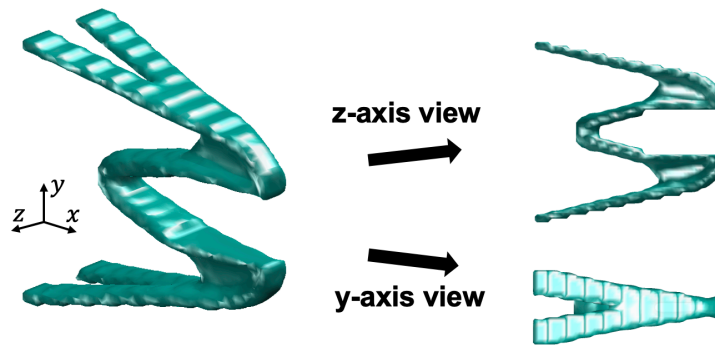Figure 5-2: Evolution of the density distribution during the optimization process ($n_z = 10$).

Figure 5-3: The realized compliant gripper for the synthesis case 1.

Case 2: $n_z = 20$

In the second case, we extended the z-axis dimension of the 3D design domain to 20 mm ($n_z = 20$, $l_e = 1mm$). Fig. 5-4 shows the automatically synthesized gripper. It can be seen that the z-
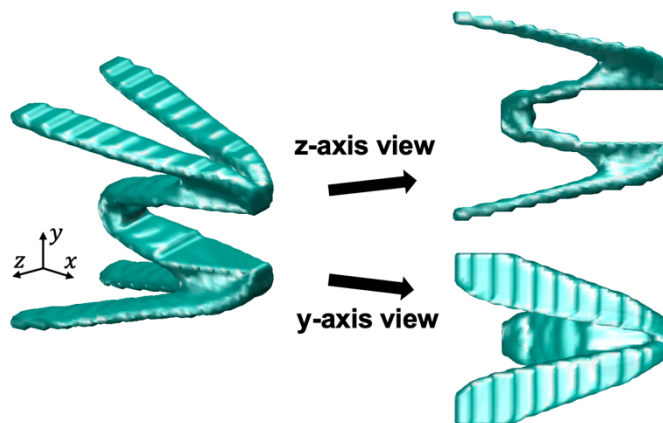

Figure 5-4: The realized gripper in case 2. The y-axis and z-axis view of the gripper are also presented.

axis view of the gripper is similar to that in case 1, and the shape in the y-axis view is an elongated version of the stable structure in Fig. 5-3. The entire design process took 357.6 seconds.

Comparing the 2D and 3D topology optimized grippers in Fig. 4-18, Fig. 5-3 and Fig. 5-4, we can see that, the synthesized 2D gripper structure is still maintained in the 3D design. In the meantime, a triangle-like structure has emerged in the y-axis view to increase the mechanical stability of the entire gripper structure. The width of the triangle-like structure is correlated to the z-dimension of the 3D design domain. Nevertheless, it should be noticed that, the elongated z-dimension could also increase the calculation time greatly.

5.2 Mesh Size

The mesh size of the finite element also plays an important role in the topology optimization process, since it has a great impact on the design details. In this section, we firstly investigate the impact of mesh size on the 2D optimization algorithm. Herein, the 2D design example from Section 4.2.4 was taken for consideration. The side length of the triangular element was set to 0.2, 0.3, 0.4 and 0.5 for comparison. Fig. 5-5 shows the optimized density distribution of different cases.
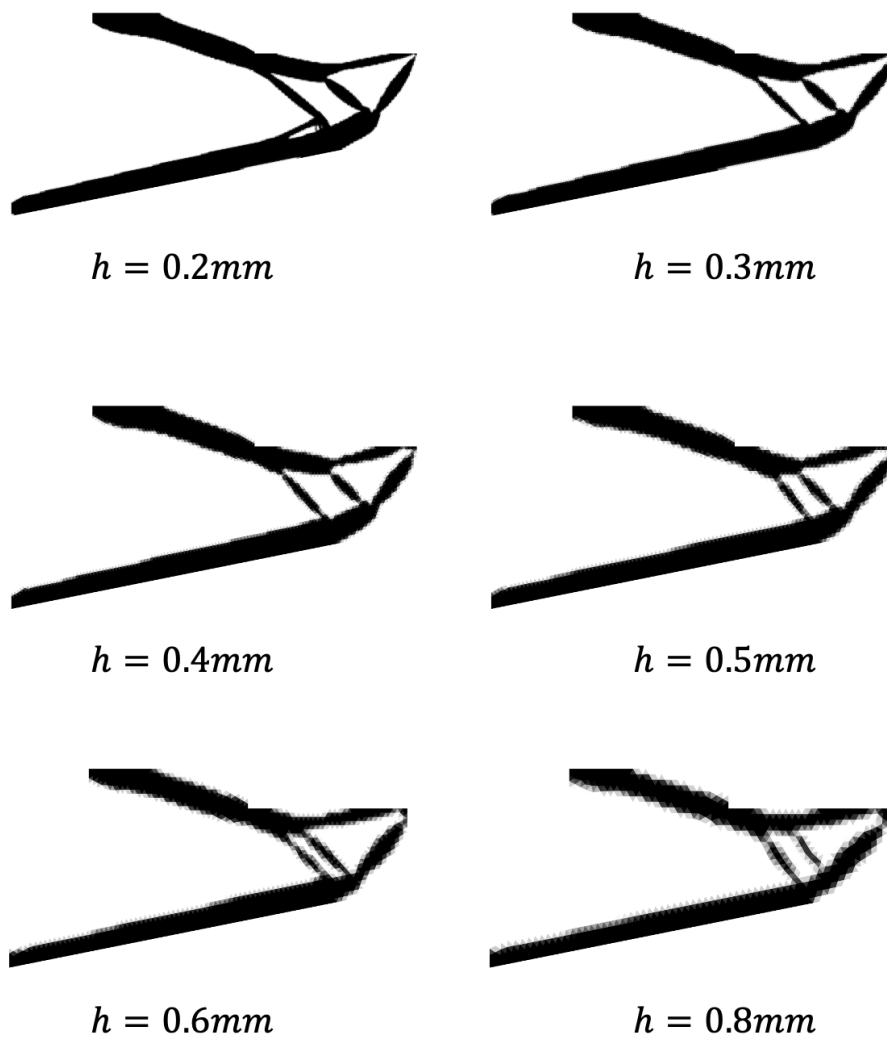

$h = 0.2mm$ $h = 0.3mm$

$h = 0.4mm$ $h = 0.5mm$

$h = 0.6mm$ $h = 0.8mm$

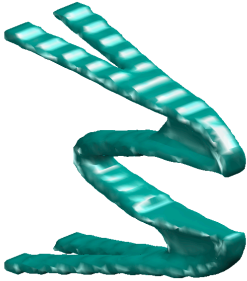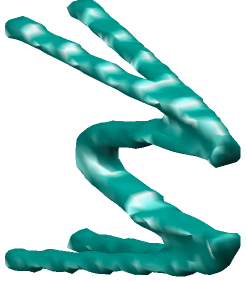**Figure 5-5: Optimized density distribution of different cases (h = 0.2, 0.3, 0.4, 0.5, 0.6 and 0.8).**

As shown in Fig. 5-5, although the topologies of the optimized results are almost the same, the resolution became worse with increasing element size. On the other hand, the iterations required for reaching convergence are similar for different cases, while the computation cost is much higher for the mesh with smaller elements (see Table 5-1).

Table 5-1: Calculation time and iteration number for 2D optimization problem with different mesh sizes.

Mesh size h [mm]	0.2	0.3	0.4	0.5	0.6	0.8
Calculation time per iteration [s]	0.163	0.074	0.041	0.025	0.015	0.010
Iteration number	168	162	167	170	161	164

We have also analyzed the impact of mesh size on the 3D topology optimization algorithm. The 3D design problem in the first case of Section 5.1 was taken for consideration. For comparison, $\{n_x, n_y, n_z\}$ and l_e were set to $\{30,15,6\}$ and 1.67 mm, respectively. Table 5-2 presents the design results of the two cases ($l_e = 1mm$ and $l_e = 1mm$). Again, the realized grippers also had similar topologies, while the refined mesh resulted in better shape quality. Since the computational cost of a 3D FEA strongly depends on the mesh size, the synthesis of a 3D design problem with smaller element size is extremely slow.

Table 5-2: Comparison of the synthesis results of the 3D design problem with different mesh sizes.

Mesh size l_e [mm]	1	1.67
Realized shape		
Calculation time per iteration [s]	1.742	0.310
Iteration number	56	54

5.3 Volume Fraction

The volume fraction is originally introduced to control the final volume of the synthesized compliant instruments. However, different values of the volume fraction could also lead to design

results with different performance. In this section, we will investigate the impact of volume fraction on the realized compliant instruments.

For the 2D design problem, we took the example from Section 4.2.4 again for analysis. In the synthesis, the volume fraction g was set to 0.1, 0.2, and 0.4 for comparison. Fig. 5-6 shows the synthesized density distribution with different volume fractions. It can be seen that, smaller g could lead to complicated network with thin branch-structure in the final topology. Although the realized mechanism with the small volume fraction could also achieve the objective movement, the high stress in the thin structure could lead to fatigue or fracture, which is undesirable.
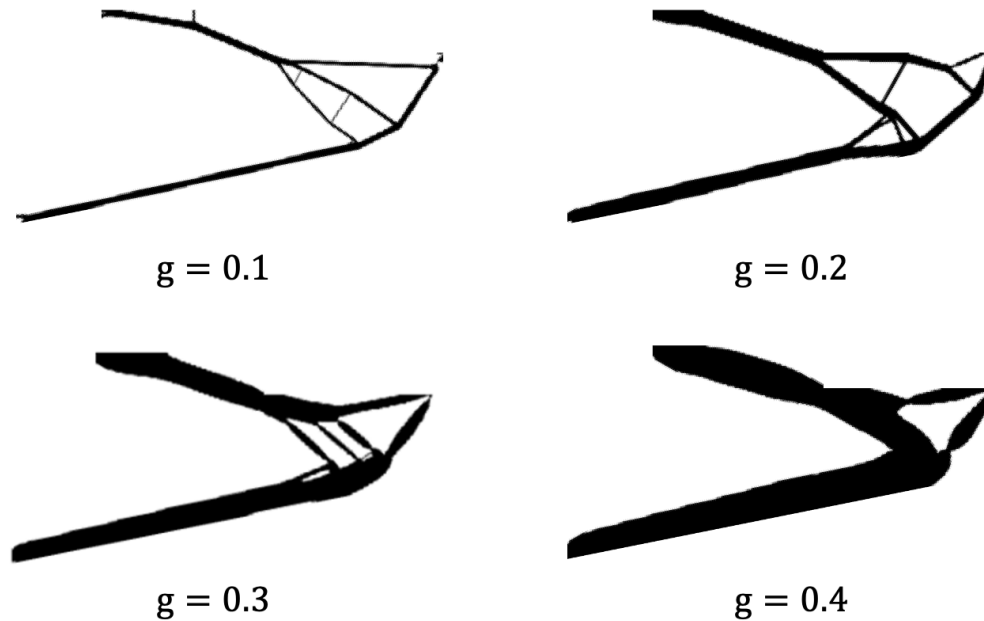

$$g = 0.1 \qquad\qquad g = 0.2$$

$$g = 0.3 \qquad\qquad g = 0.4$$

Figure 5-6: The synthesized structure with different volume fractions.

Since the example in Section 4.2.4 realizes a distributed compliant mechanism, we took another synthesis example to analyze the impact of volume fraction on the lumped compliant mechanism. Fig. 5-7 shows the design problem of the second example, in which a compliant crimper should be designed. f_{in} and u_{out} indicate the actuation force and the objective movement, respectively. The bottom of the design domain is fixed. Ω_C depicts the predefined solid area. In the automatic synthesis, the volume fraction was set to 0.3, 0.4, 0.5 and 0.6 for comparison. Fig. 5-8 presents the synthesized crimper shapes.
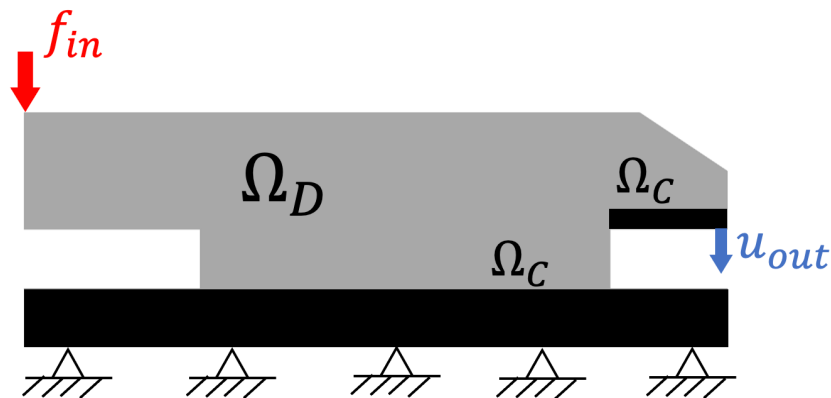

Figure 5-7: The design problem for a 2D compliant crimper.
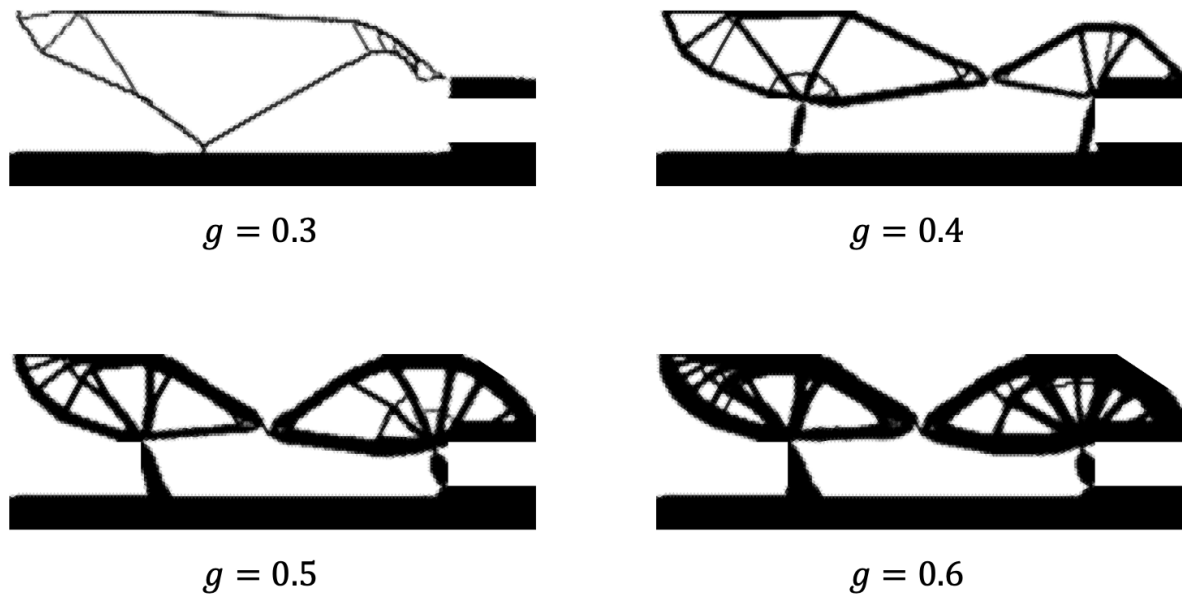
Figure 5-8: The synthesized compliant crimpers with different volume fractions (g = 0.3, 0.4, 0.5, 0.6).

It can be noticed that too small volume fraction ($g = 0.3$) could lead to a non-functional mechanism. After g exceeds a certain value (0.4), the basic flexure-hinge structures of a compliant crimper were established, while the increasing g (0.5 and 0.6) led to a higher infilling degree of the emerged porous structure. Therefore, for the lumped compliant instrument, it is important for the user to find out the threshold value of g to realize a functional mechanism.

For the 3D design problem, the first case of Section 5.1 was used for analysis. In the synthesis, volume fraction was set to 0.1 and 0.3 for comparison. Similar to the results in Fig. 5-6, smaller g had introduced thin branch-structures in the final topology, which could lead to mechanical fatigue or even fraction when large loads are applied.
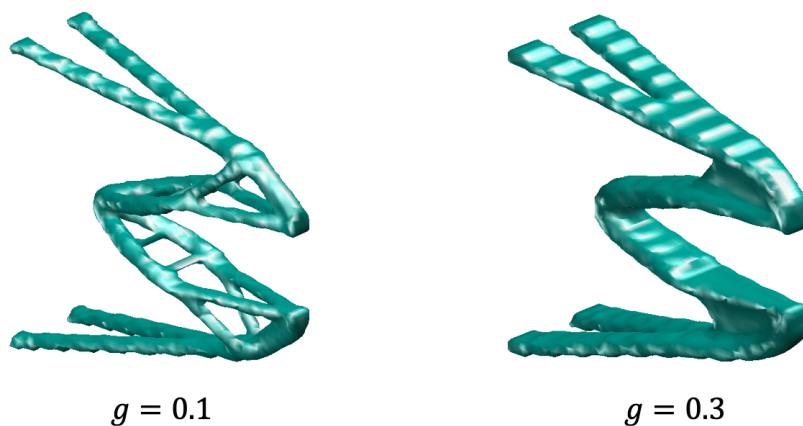

Figure 5-9: The synthesized 3D compliant gripper with different volume fractions (g = 0.1, 0.3).

Therefore, for both 2D and 3D topology optimization algorithms, the volume fraction should be chosen above the threshold value in order to synthesize a functional compliant instrument. The threshold value can be determined by using a trial-and-error method.

5.4 Geometrical Constraints

In this section, the influence of the geometrical constraints will be analyzed. As is described in Section 4.2.1, the geometrical constraints are predefined areas in the design domain, which remain solid through the entire synthesis process. The geometrical constraints are usually used to realize end effectors of the compliant instruments with user-defined shape.

To investigate the influence of geometrical constraints on the 2D synthesis results, the design problem in Section 4.2.4 was modified by adding a constraint on the gripper jaw (see Fig. 5-10). From the synthesis results presented in Fig. 5-11, it can be seen that the shape of the realized gripper jaw cannot be controlled if no geometrical constraint is applied.
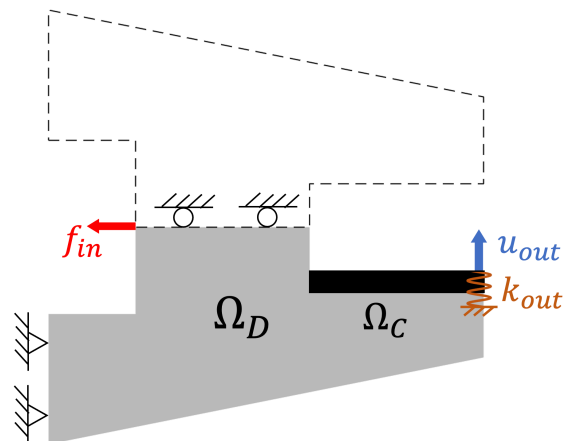

Figure 5-10: The design problem of a 2D compliant gripper with geometrical constraint on the gripper jaw.

(a)   (b)

Figure 5-11: Comparison of the 2D synthesis results without and with geometrical constraints: a) Without geometrical constraint, b) With geometrical constraint.

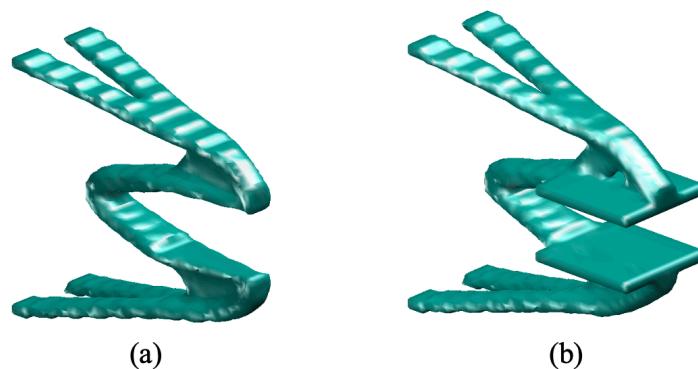(a)   (b)

Figure 5-12: Comparison of the 3D synthesis results without and with geometrical constraints: a) Without geometrical constraint, b) With geometrical constraint.

Fig. 5-12 compares the synthesized 3D grippers without and with geometrical constraint on the gripper jaw. Similar to the result presented in Fig. 5-11, the shape of the realized gripper jaw can be controlled if an additional geometrical constraint is applied.

5.5 Conclusion of the Chapter

In this chapter, several important design parameters were investigated with respect to their influence on the synthesis results of the proposed 2D and 3D topology optimization methods. It can be concluded that, the 3D synthesis results have inherited the main features from their 2D versions while the 3D-extension could optimize their mechanical stability of third dimension. For the selection of the mesh size, the user should find the balance between the design resolution and the computation cost. It is also important to figure out the minimum value of the volume fraction for realizing a functional compliant instrument. Besides, the geometrical constraint is also helpful when the user wants to realize a predefined shape of a certain area of the compliant instrument.

6 Applications to the Design of Compliant Medical Instruments

This chapter presents the design of several compliant medical instruments by using our automatic design framework. All the calculations were executed on a computer with an Intel Core i7 CPU at 2.9 GHz. The realized instruments were also 3D-printed in order to evaluate their performance.

6.1 Disposable Compliant Forceps for Open Surgery

Forceps is one of the most basic surgical instruments for open surgical procedures, such as holding needles during suturing and stabilizing tissues during dissection. To prevent the spread of infectious diseases from re-used and poorly sterilized forceps, disposable forceps are widely used in various surgical applications (Sopwith, Hart and Garner, 2002). In this section, we present the design of a disposable open surgical forceps by using our 2D topology optimization method.

6.1.1 Design Problem

Fig. 6-1 shows a 2D geometry (CPL) constructed by the SGCL language for modeling the initial design domain. The derived design problem is presented in Fig. 6-2. In the diagram in Fig. 6-2, f_{in} is the gripping force on the forceps handle while f_{clamp} simulates the clamping force on the forceps tip for grasping objects. The stiffness of the clamped tissue is modeled as a linear spring with spring constant k_{out} at the tip of the forceps. Since the forceps is SLS-printed by using the polyamide PA2200, the elastic modulus E_0 and Poisson's ratio v of PA2200 are used in the FE-Analysis. The objective of the design problem is to maximize the displacement u_{out} of the forceps tip in the positive y-direction with the predefined loading cases.
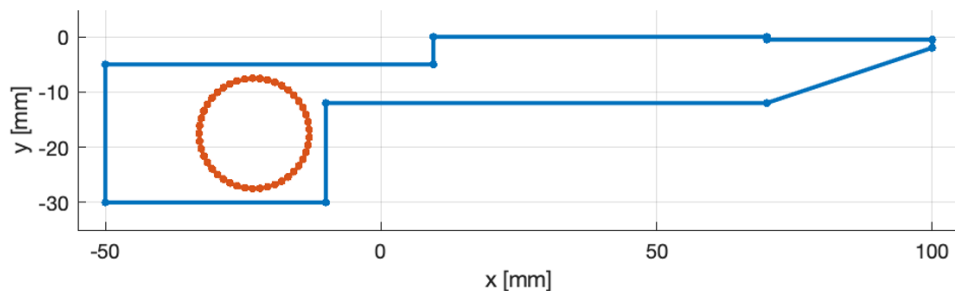

Figure 6-1: A 2D geometry as the design domain for the forceps. The blue and orange curves depict the exterior and interior boundary of the design domain, respectively.
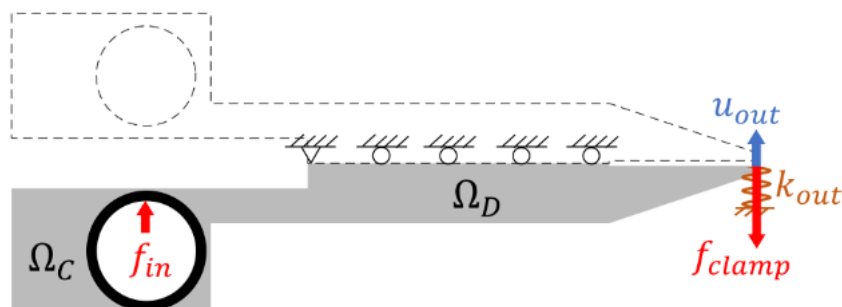

Figure 6-2: A schematic diagram illustrating the design problem for the disposable forceps.

6.1.2 Automatic Shape Synthesis

This section shows the automatic synthesis process of the proposed disposable compliant forceps. The value of several important parameters for the automatic synthesis are listed in Table 6-1. The clamping force and spring constant on the forceps tip were chosen according to the study results in Trobec and Gersak, 1997; Stoll and Dupont, 2006, respectively.

Table 6-1: Several important parameters for the automatic synthesis process.

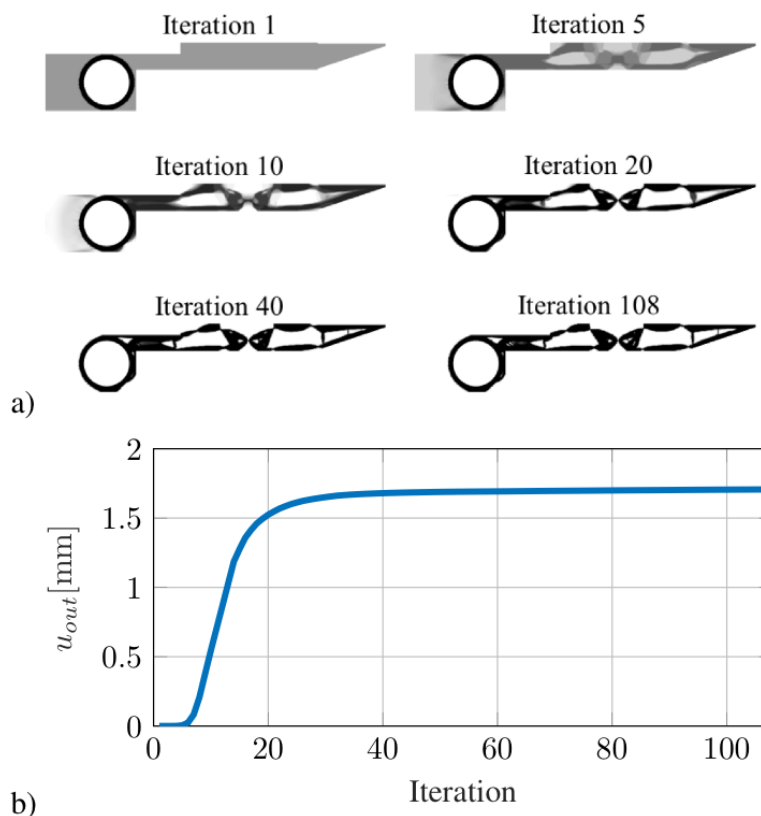| Parameter	Symbol	Value
Gripping force on the handle	f_{in}	$2\ N$
Clamping force on the tip	f_{clamp}	$2\ N$
Spring constant at the output port	k_{out}	$0.75\ N/mm$
Elastic modulus	E_0	$1700\ MPa$
Poisson's ratio	v	0.3
Maximum element size in FEA	h	$0.4\ mm$
Volume fraction	g	0.45

Figure 6-3: Automatic shape synthesis of the proposed disposable compliant forceps: a) Evolutionary process of the density distribution in the design domain during the topology optimization, b) Value of the objective function u_{out} during the automatic synthesis process.

Fig. 6-3a) presents the evolutionary process of the density distribution during the automatic shape synthesis. The topology optimization process reached its convergence at the 108th iteration. The entire process took 57.35 s. As can be seen in Fig. 6-4b), the maximum output displacement u_{out} (1.718 mm) was reached at the last iteration. Fig. 6-4a) shows the 3D model of the realized compliant forceps after post-processing. The thickness of the forceps in z-axis direction was 5 mm. Fig. 6-4b) shows the SLS-printed disposable forceps.
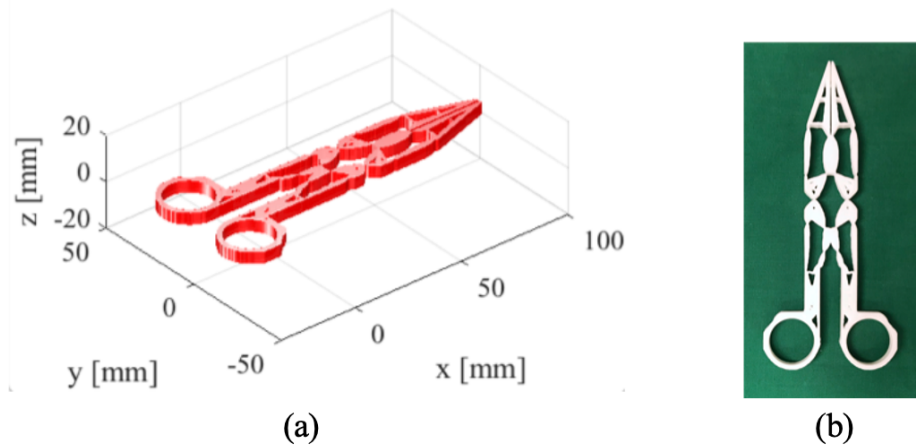

(a)   (b)

Figure 6-4: Realized open surgery forceps: a) The post-processed 3D surface model, b) The SLS-printed forceps.

6.1.3   FEM-Based Simulation

In this section, two FE-Analyzes were performed to evaluate the opening and clamping capability of the disposable compliant forceps. The FEM method described in Chapter 3 was used. A symmetrical load $F_1 = 2N$ was applied on the forceps handle in Fig. 6-5a) to simulate the opening movement of the compliant forceps while a load with the same magnitude $F_1 = 2N$ but the opposite direction was applied in Fig. 6-5b) to evaluate the clamping capability. The displacement and von Mises stress of the loaded forceps are shown in Fig. 6-5.
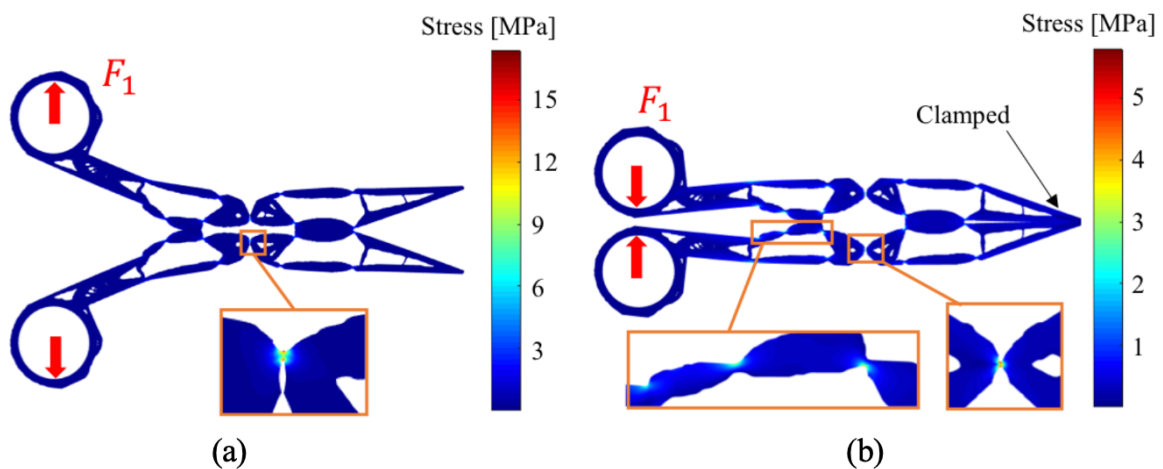

(a)   (b)

Figure 6-5: FE-Analysis of the opening and clamping movements of the proposed compliant forceps: a) Displacement and stress of the compliant forceps when a symmetrical load F_1 (red arrow) was applied on the handle to open the forceps, b) Displacement and stress of the compliant forceps when a load with the same magnitude F_1 (red arrow) was applied to close the forceps tips.

The FEA results in Fig. 6-5 demonstrated that the proposed compliant forceps could success-fully achieve opening and clamping movements with the given load F_1. In Fig. 6-5b), the dis-tance between the two pairs of the forceps handle was shortened to close the tips. The calculated clamping force on the forceps tip was 2.45 N. From the stress distribution, it can be noticed that, in both loading cases the stress was not equally distributed in the entire forceps but con-centrated in some parts of thin flexure hinges. During the clamping movement, the maximum stress in the forceps was lower than that in the opening position. This can be attributed to the boundary conditions employed in the topology optimization process, which was specifically designed for the clamping movement (see Fig. 6-2).

6.1.4 Experiments

A series of experiments were conducted to validate the clamping capability of the proposed disposable compliant forceps.

Tests of Clamping Capability

As is shown in Fig. 6-5b), the clamping force on the forceps tip is correlated to the gripping force on the handle, while the deformation of the two handle pairs can be used to characterize the gripping force. Hence, we measured the deformation of the forceps handle pairs as well as the resulting clamping force in a series of loading tests, to experimentally evaluate the clamping capability of the proposed forceps. The experimental setup is shown in Fig. 6-6. A weight F_G was employed to pull the two forceps tips apart through cables, which emulated the clamping resistance. At the same time, we applied gripping force F_{grip} on the two handle pairs to achieve closure of the forceps tips. The digital microscope of Conrad was used to measure the distance d_h between the two handle pairs (see Fig. 6-6) at the moment when the two forceps tips just touched each other. In this case, the clamping force F_{clamp} was equal to the dragging force F_{cable} in the cable. The deformation Δd of the two handle pairs and the clamping force F_{clamp} can be formulated as:
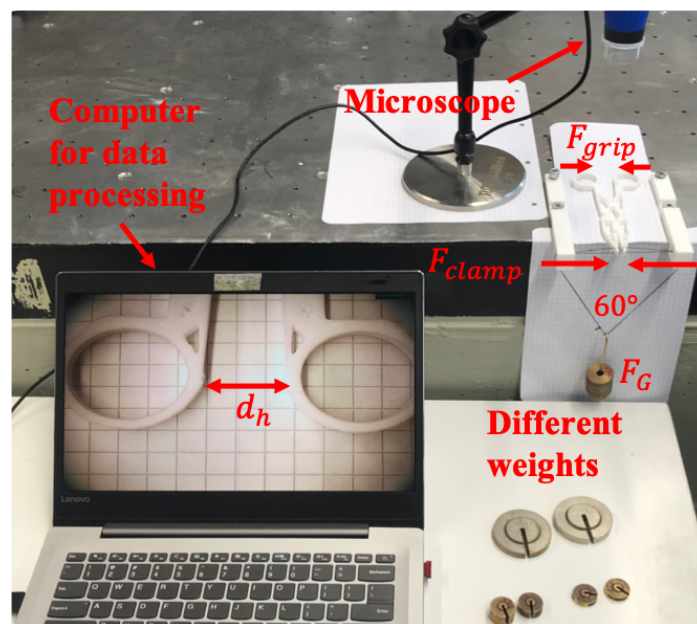

Figure 6-6: Experimental setup for testing the clamping capability of the proposed forceps. A microscope was used to determine the deformation of the forceps handle while different weights were used to emulate the clamping resistance.
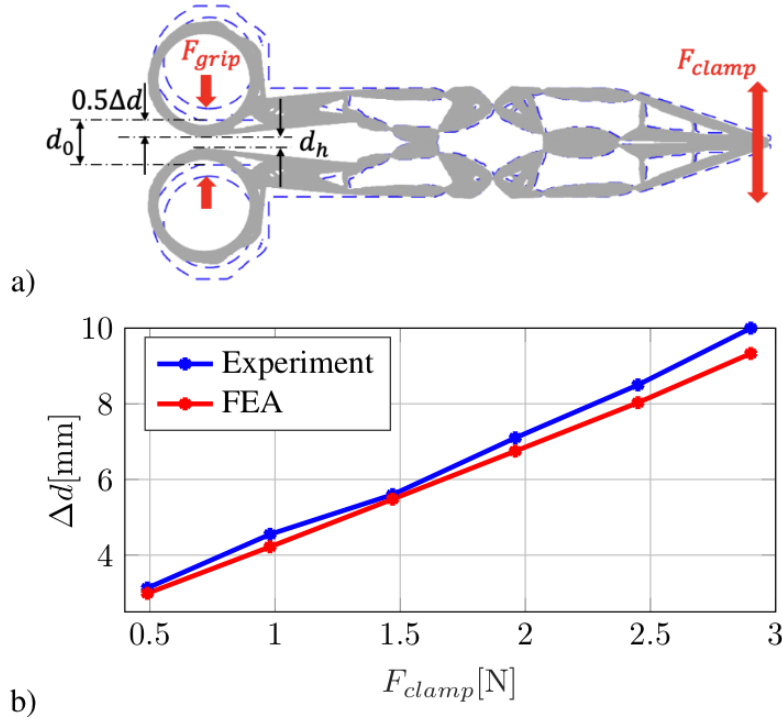
a)

b)

Figure 6-7: Schematic representation and results of the experiments: a) A schematic diagram which illustrates the loading cases of the forceps in the experiments, b) Diagrams of the relationship between F_{clamp} and Δd. The blue one was measured by experiments while the red one was calculated by FEA.

$$\Delta d = d_0 - d_h \qquad (6\text{-}1)$$

$$F_{clamp} = F_{cable} = \frac{\sqrt{3}}{3} F_G \qquad (6\text{-}2)$$

where $d_0 = 10mm$ is the original distance between the two handle pairs. F_{clamp} in Equation (6-2) is determined according to the law of sines. The schematic diagram in Fig. 6-7a) illustrates the loading cases and deformation of the forceps in the experiments. Apart from the experimental measurements, a series of FE-simulations were also performed to evaluate the handle deformation Δd. The same set of clamping forces as in the experiments were applied in the FEA for calculation. The experimental and FEA results are presented in Fig. 6-7b). It can be noticed that, the experimental results were very close to those of the simulations, which shows the plausibility of our design. On the other hand, Δd and F_{clamp} showed almost linear correlations in both experimental and FEA results. The maximum clamping force measured in the experiments was 2.90 N, which was achieved at $\Delta d = d_0 = 10mm$ when the two forceps handle pairs were completely closed. The maximum clamping force of the proposed forceps is already sufficient for the most clamping tasks in cardiovascular surgery according to Trobec and Gersak, 1997.

Clamping in Open Surgical Dissection

To test the clamping performance of the proposed forceps in open surgical tasks, an experiment of surgical dissection was performed in this section on a silicone heart. The surgical instruments and the silicone heart used in the experiment are shown in Fig. 6-8a). The disposable compliant forceps was used for stabilizing the phantom tissue while conventional surgical scissors were

used for cutting the tissue. As can be seen Fig. 6-8b), the forceps could successfully clamp and stabilize the phantom tissue during the dissection task, which shows the reliability of the proposed forceps.
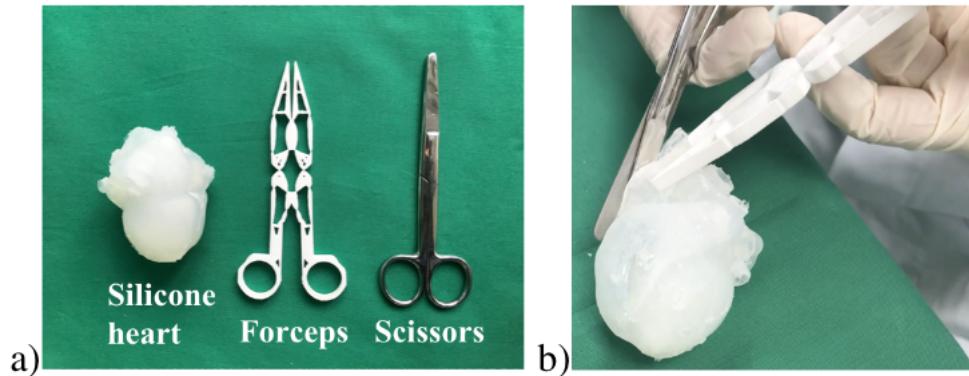

Figure 6-8: Open surgical dissection on a silicone heart: a) The disposable forceps, the conventional scissors and the silicone heart for the dissection experiment, b) Dissection on the phantom tissue.

6.2 Compliant Forceps for Minimally Invasive Surgery

In this section, two compliant MIS-forceps were synthesized using our 2D topology optimization method. Different initial design domains and actuation concepts were used in the design. A 3D-printed continuum manipulator with a synthesized forceps is also presented to demonstrate the application of the proposed method in robot-assisted MIS.

6.2.1 MIS-Forceps Actuated by a Single Tendon Force

Design Problem and Synthesis Result

For the first MIS-forceps, the design problem from Fig. 6-9b) was used, where a single tendon force is applied to actuate the forceps. The design objective is to achieve the maximum displacement u_{out} of the forceps tip. Fig. 6-9a) shows the initial design domain and the corresponding optimization problem. In the synthesis, the maximum mesh size was set to 0.1 mm for performing FEM analysis. f_{in} and k_{out} were chosen as 0.5 N and 0.5 N/mm respectively, as k_{out} is used to mimic the gripping resistance of the soft tissues (Stoll and Dupont, 2006).
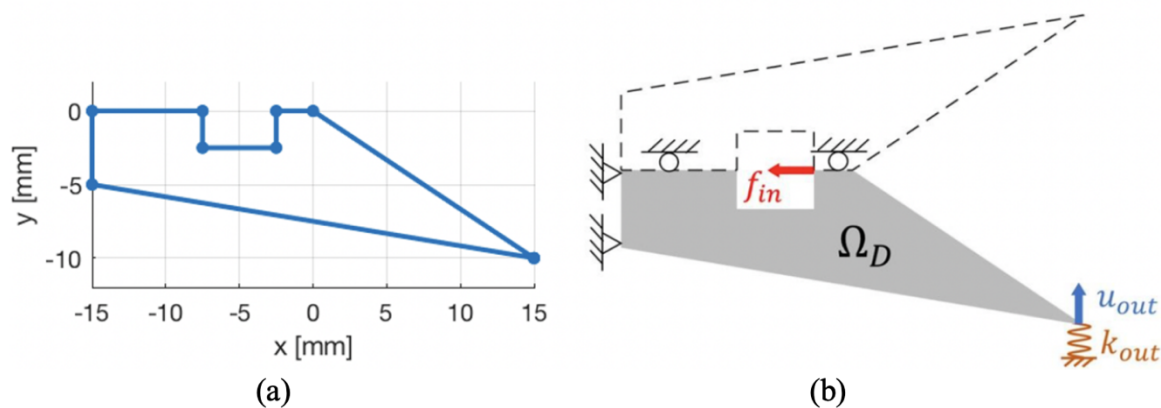

Figure 6-9: The design problem of the first MIS-forceps: a) The initial design domain (CPL), b) The schematic diagram of the formulated optimization problem.

Since the forceps is supposed to be SLS-printed by using polyamide (PA2200), E_0 was set to 1700 MPa. The volume constraint was 0.5. Fig. 6-10a) shows the evolution process of $\boldsymbol{\rho}$ in the design domain, where the black area represents the solid material. The final topology is reached at the 101st iteration. The convergence time was 47.33 s. Fig. 6-10b) shows the trend of the objective function u_{out} over the iterations. It can be noticed that u_{out} reaches its maximum, 3.4 mm, at the end of the synthesis. By extracting the boundary CPL of the optimized density distribution in Fig. 6-10a) and merging it with its symmetrical duplicate, a 2D forceps model is created, as is shown in Fig. 6-10c).
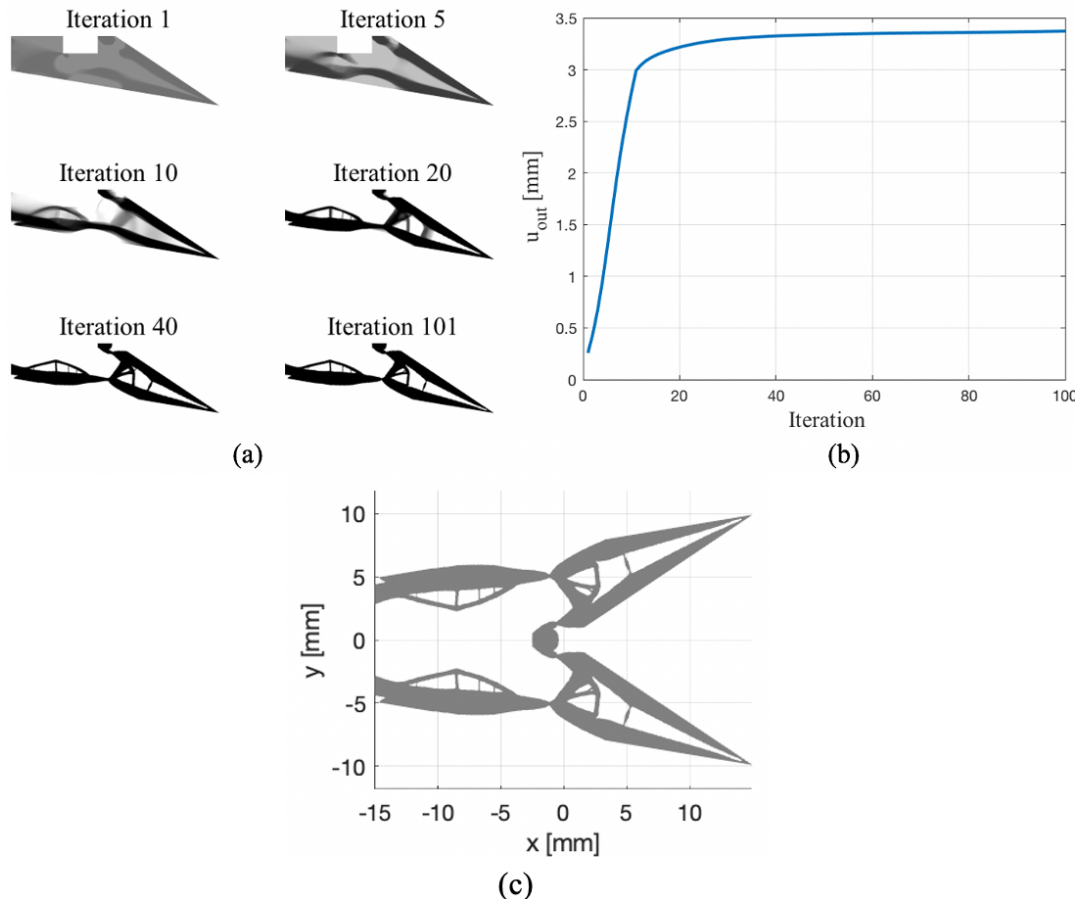

Figure 6-10: Automatic synthesis process for the design problem in Fig. 6-9: a) The density distribution during some iterations, b) Value of u_{out} during the synthesis process, c) The created 2D forceps model.

Experimental Tests

Experimental tests were also conducted to evaluate the clamping performance of the realized MIS-forceps. The realized 2D forceps in Fig. 6-10c) were extruded and integrated into the 3D model in Fig. 6-11a) for performing the experiment, where an additional base is constructed to fix the forceps. The entire 3D model was SLS-printed. In the experiment (see Fig. 6-11b), weight f_G of different masses was attached to a single cable to pull the forceps to test its clamping performance. To mimic the resistance of the clamped soft tissues, a linear spring was placed between the forceps tips, which is identical to the loading case of the optimization problem in Fig. 6-9. The spring stiffness was chosen as half of k_{out} since two serially connected springs of k_{out}, which are symmetrically applied on the two forceps tips, can be treated as a spring of $k_s = \frac{k_{out}}{2} = 0.25 N/mm$. A digital microscope (Conrad DP-M17) was used to measure the
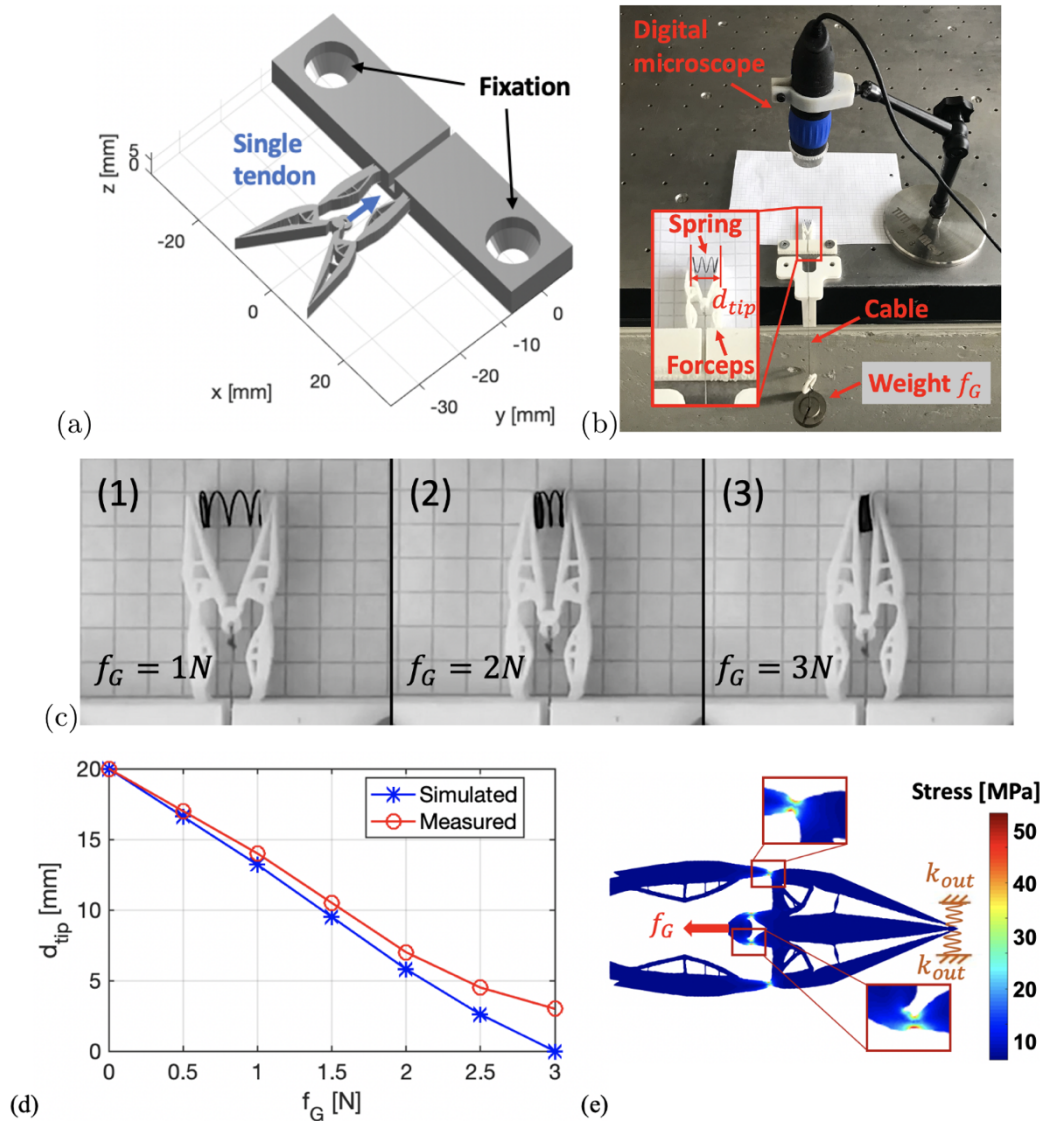
Figure 6-11: Evaluation of the clamping performance of the realized forceps in Fig. 6-10: a) The SG of the realized forceps and the fixed base, b) Experimental setup, c) Measurements with different tendon forces (1: $f_G = 1N$, 2: $f_G = 2N$, 3: $f_G = 3N$), d) A comparison of the measured and FEM-simulated d_{tip}, e) The FEM-simulated stress distribution of the realized compliant forceps in the closed state.

distance d_{tip} between the forceps tips. Fig. 6-11c) shows several measurements, where each measurement was repeated for three times. Mean value of the measured results is reported in Fig. 6-11d). To further analyze the clamping performance of the forceps, large-displacement FEM analysis were also performed to calculate d_{tip}. The simulated results are presented in Fig. 6-11d) as well. Both the measured and simulated results show that the printed forceps can successfully grasp and compress the elastic spring with the single tendon force. Besides, the gripping force applied on the spring is almost proportional to the tendon force f_G. It can also be noticed that there is a large error between the simulated and measured d_{tip}, when f_G is greater than 2 N. The reason for the large error is that the fully compressed spring still has an incompressible thickness of 2.5 mm. Fig. 6-11e) presents the simulated quasi-static stress distribution of the fully closed forceps, which was calculated by the large-displacement FEM tool in Chapter 3. The polyamide was treated as a linear elastic material in our cases with a high yield strength of 58 MPa. We can see that the most deformations and stresses were located in the thin-wall
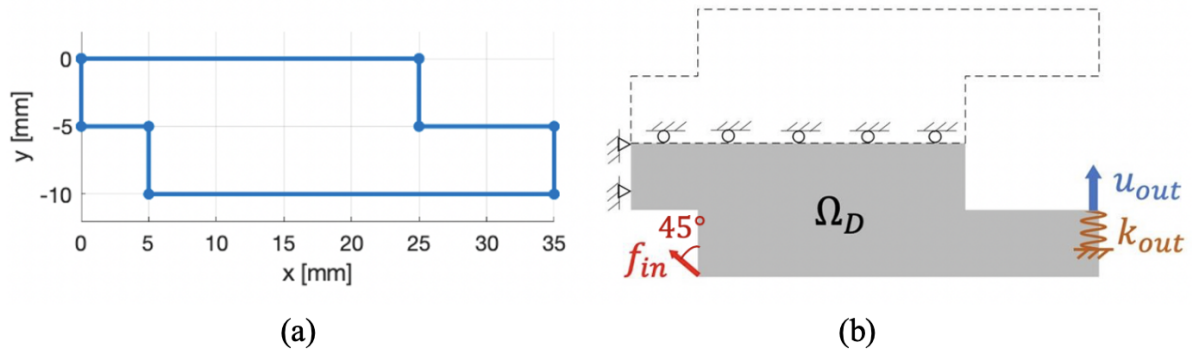
(a)　　　　　　　　　　　　　　　　(b)

Figure 6-12: The design problem of the second MIS-forceps: a) The initial design domain (CPL), b) The schematic diagram of the formulated optimization problem.

flexure hinges of the compliant forceps during the clamping movement. From the kinematic point of view, the flexure hinges are created in the synthesis process in order to realize motion transmissions of the compliant forceps.

6.2.2　MIS-Forceps Actuated by Symmetric Tendon Forces

Design Problem and Synthesis Result

For the second MIS-forceps, a pair of symmetric tendon forces were used to actuate it. The initial design domain and the formulated design problem is presented in Fig. 6-12. Different from the first synthesis example, the actuation force in Fig. 6-12b) was applied on both
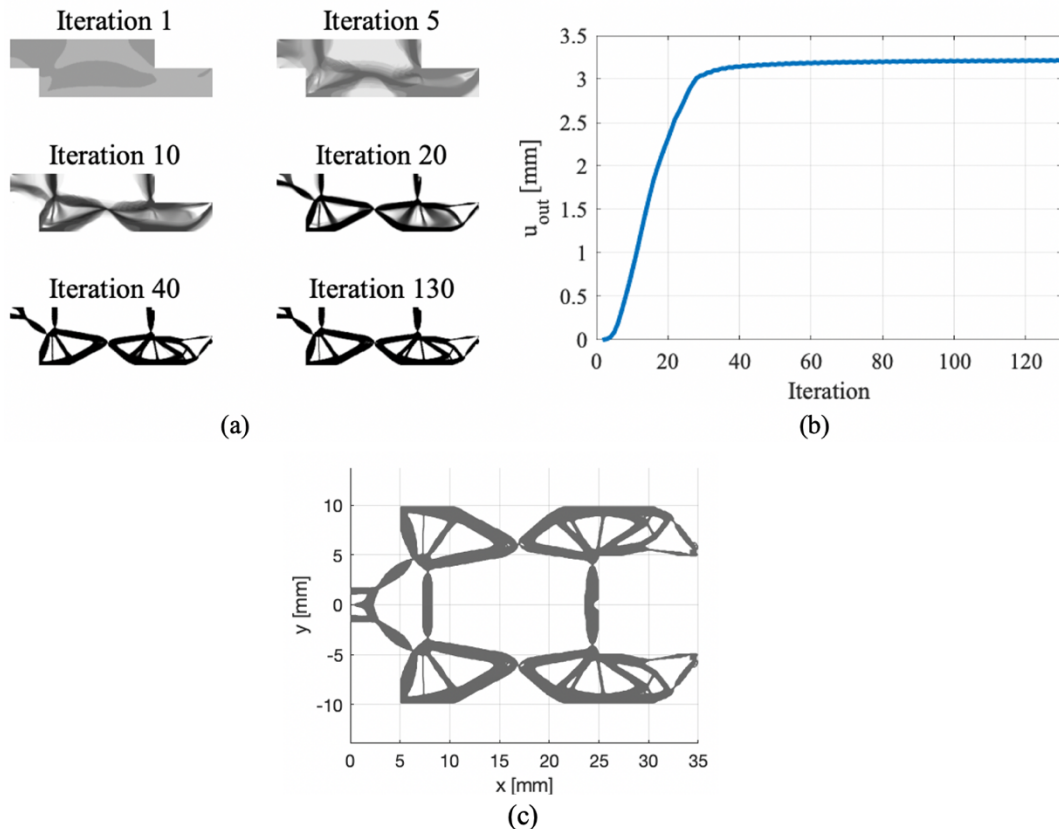

(a)　　　　　　　　　　　　　　　　(b)

(c)

Figure 6-13: Automatic synthesis process for the design problem in Fig. 6-12: a) The density distribution during some iterations, b) Value of u_{out} during the synthesis process, c) the created 2D forceps model.

symmetric design domains with an offset from the symmetric axis. In the synthesis, the left side of the design domain was fixed, and the volume constraint was set to 0.35. The other parameters were the same as those in the first example. Fig. 6-13a) shows the evolution process of ρ in the design domain. The final topology is reached at the 130th iteration. The convergence time was 56.21 s. Fig. 6-13b) shows the trend of the objective function u_{out} over the iterations. It can be noticed that u_{out} reaches its maximum, 3.2 mm, at the end of the synthesis. Fig. 6-13c) shows the 2D forceps model created by extracting the boundary CPL of the optimized density distribution in Fig. 6-13a) and merging it with its symmetrical duplicate.

Experimental Tests

Similar to the first case, experiments were also conducted to evaluate the clamping performance of the second MIS-forceps. The realized 3D forceps model is presented in Fig. 6-14a) and also SLS-printed for performing the experiments. In the experiment, the two actuation cables are attached to both sides of the forceps respectively and go through the canals of the base. The two cables are then connected together in a united canal and driven by weight f_G of different masses. In this case, the symmetric pulling forces applied on the forceps were both $\frac{f_G}{2}$. The linear spring of 0.25 N/mm was also placed between the forceps tips as gripping resistance. Fig. 6-14b) shows several measurements, where each measurement was also repeated for three times. It can
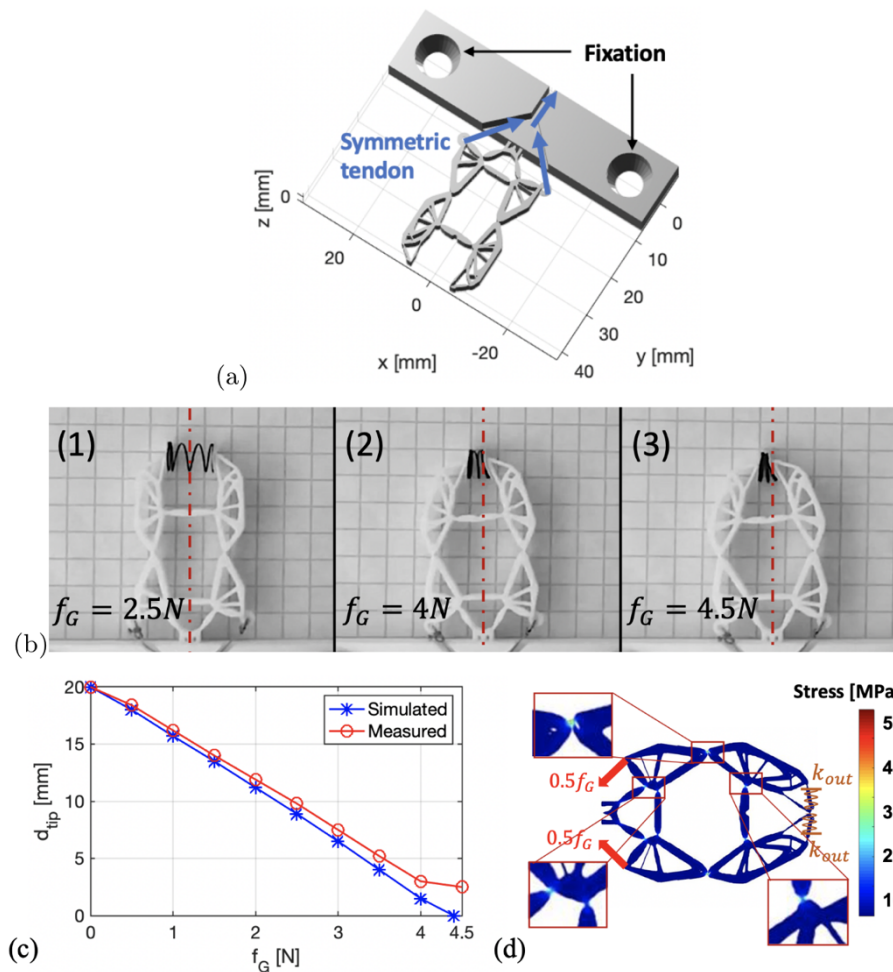

Figure 6-14: Evaluation of the clamping performance of the realized forceps in Fig. 6-13: a) The SG of the realized forceps and the fixed base, b) Measurements with different tendon forces (1: $f_G = 2.5N$, 2: $f_G = 4N$, 3: $f_G = 4.5N$), c) A comparison of the measured and FEM-simulated d_{tip}, d) The FEM-simulated stress distribution of the realized compliant forceps in the closed state.

be seen that, the forceps can be successfully closed in a teeth-biting manner, which is different from the complete closure of the forceps jaws in the first example. This interesting feature could be developed for point-clamping or suturing in the robot-assisted MIS. The measured d_{tip} and the FEM-simulated results are reported in Fig. 6-14c). We can see that d_{tip} and f_G are strongly linear correlated. Fig. 6-14d) presents the FEM-simulated stress distribution of the closed forceps. It can be noticed that the most deformations occur in the thin-wall flexure hinges, which is the same as the first MIS-forceps.

From the measured motions of the SLS-printed prototype in Fig. 6-14b) we can see that the forceps tips were successfully closed with the symmetric tendon forces. However, the asymmetric motions of the left and right parts of the forceps can also be noticed in the figure, as the second deformed shape in Fig. 6-14b) shows. A possible reason for this problem could be the manufacturing error that occurred when printing the complex forceps structure, which led to an asymmetric structure of the printed forceps. From this point of view, the single-tendon actuation concept from the first MIS-forceps is more stable for the 3D-printed compliant forceps.

6.2.3 3D-Printed Manipulator Prototype for Robot-Assisted MIS

A SLS-printed continuum manipulator is presented in this section to demonstrate the application of the synthesized forceps in robot-assisted MIS. The realized MIS-forceps in Fig. 6-10 was used as the end effector and the modeling of the manipulator was realized using the SGCL language. As can be seen in Fig. 6-15a), in order to realize dexterous manipulation, the manipulator contained two bending sections whose bending orientations were perpendicular to each other. The compliant rolling-contact joint (CRJ) from Shaw et al., 2018 was employed as the elementary bending unit because of its large bending angle and high flexibility, as is shown in
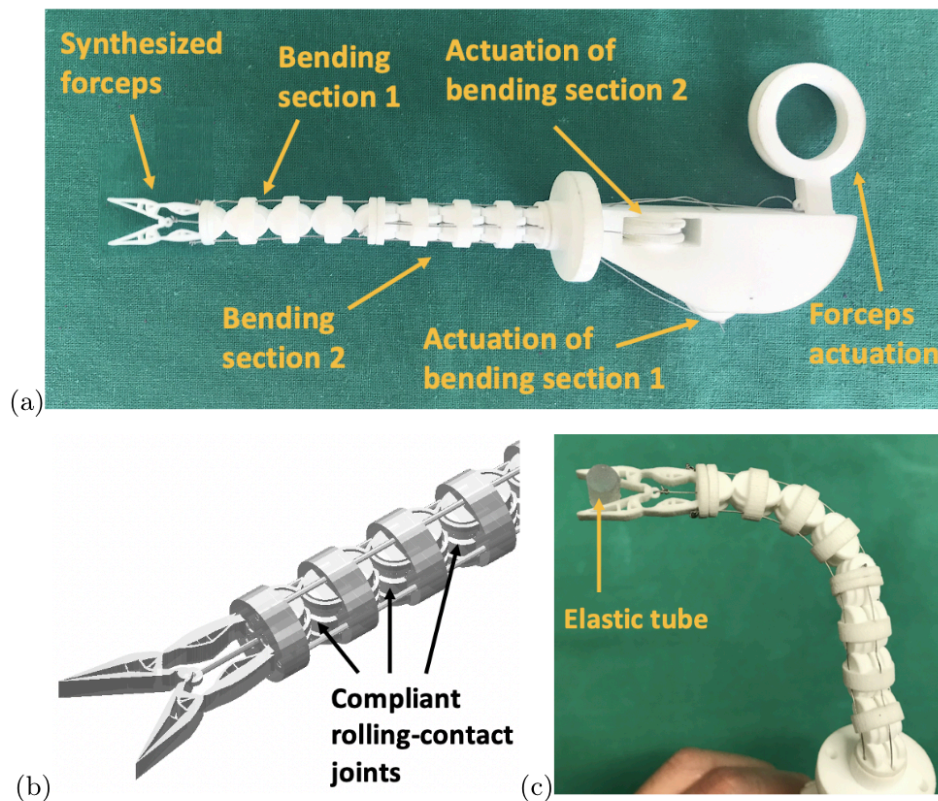

Figure 6-15: A 3D-printed continuum manipulator for robot-assisted MIS: a) The prototype of the manipulator, b) The 3D geometry of the bending section, c) Test of the performance of the manipulator by clamping an elastic tube.

Fig. 6-15b). To achieve the cable-actuation of the two bending sections and the forceps, three independent rollers were introduced and attached to the corresponding cables. In Fig. 6-15c), an elastic tube, fabricated with the Stereolithography (SLA) printing using an elastomer material (Elastic Resin), was used to evaluate the clamping performance of the prototype. The result shows that the manipulator can successfully grasp the elastic tube using the synthesized forceps and at the same time achieve flexible manipulation. In this way, the application of the automatic synthesis method in the robot-assisted MIS was demonstrated.

6.3 Compliant Surgical Forceps With Adaptive Grasping Functions

The synthesized surgical forceps in Section 6.1 and 6.2 can successfully achieve robust clamping, as the experimental results presented. However, the realized compliant forceps still perform stiff grasping, and thus can damage sensitive organs and tissues during the operation. From this point of view, adaptive grasping function is required for safe manipulation of vulnerable structures. In this section, two adaptive compliant forceps were synthesized using our 2D topology optimization method. In particular, the first forceps has been designed for traditional open surgery while the second one has been realized for robot-assisted MIS.

6.3.1 Adaptive Compliant Forceps for Open Surgery

The design problem described in Fig. 6-16 was used to perform the automatic design process of open surgery forceps. The design objective is also to achieve the maximum displacement u_{out} of the forceps tip. The maximum length and width of the initial design domain were respectively 100 mm and 60 mm (see Fig. 6-17). The design domain was meshed into triangular elements with a maximum size of 0.4 mm. The forceps was printed with selective laser sintering (SLS) by using polyamide (PA2200) as material. The material parameters for the FEA process were the same as in Section 6.1 and 6.2 while the displacement-based input load was 5 mm. To achieve adaptive grasping function, two linear springs with stiffness k_{out} and k_{tissue} are applied on the tip and the middle region of the gripper jaw respectively. This makes it possible to simulate the external resistances during the grasping movements. We assumed that $k_{tissue} = 0.75 N/mm$ is greater than $k_{out} = 0.5 N/mm$ so that the sensitive tissue becomes the main external resistance for the forceps to overcome. The proposed topology optimization process of compliant mechanisms always tries to achieve the maximum output displacement while overcoming the internal and external resistance, hence the deformation of the tissue during grasping will also be minimized in the optimization process. The prescribed volume fraction g was set to 0.3. The topology optimization process converged at the 99th iteration within a tolerance ε of
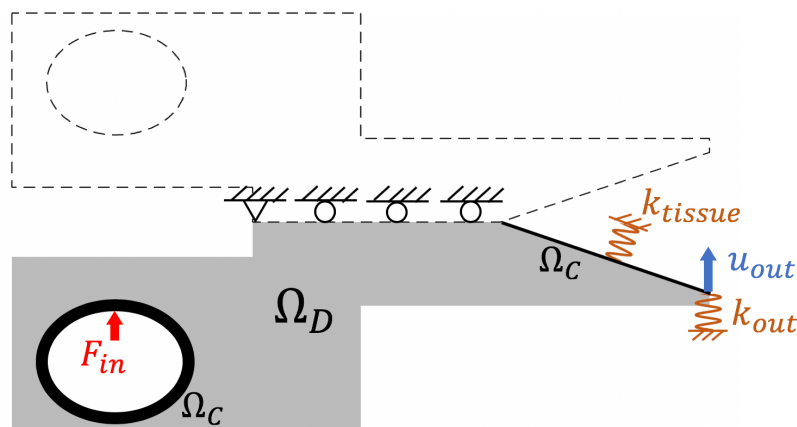

Figure 6-16: Design problem for the adaptive compliant forceps for open surgery.
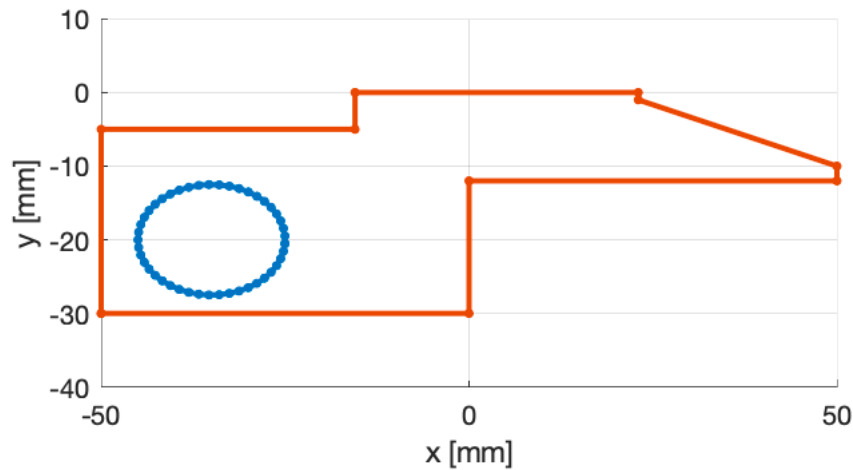
Figure 6-17: The 2D geometry of the design domain in Fig. 6-16.

0.005. The convergence time was 51.94 s. Fig. 6-18a) shows the trend of the objective function f over the iterations. It can be noticed that f reaches its maximum, 9.15 mm, at the end of the process. The evolution process of the forceps topology in some iterations is illustrated in Fig. 6-18b).
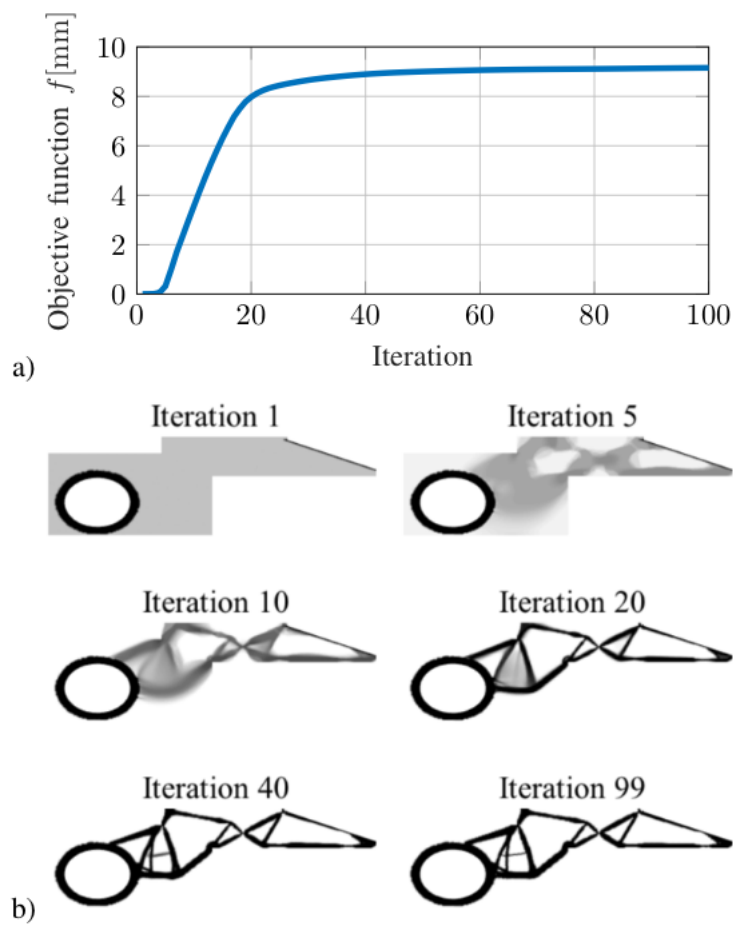

Figure 6-18: Topology optimization process of the adaptive compliant forceps for open surgery: a) The trend of the objective function f during the optimization process, b) The evolution process during some iterations. The final topology of the forceps is reached at the 99th iteration.

95

As mentioned in Section 4.2.3, the boundary CPL of the optimization result was derived by using the triangular-mesh-based extraction method. The extracted CPL and its symmetrical duplicate were merged and then extruded into a solid with thickness 5 mm. Fig. 6-19a) shows the constructed SG and the SLS-printed prototype of the adaptive compliant forceps. The adaptive grasping function is shown in the FE-simulations presented in Fig. 6-19b). The simulation is based on the geometrically non-linear FE-methods described in Chapter 3. In both simulations, a symmetrical load is applied on each handle pair to shorten their distance by 10 mm. In the first case, no external resistance is applied on the gripper jaw. In the second case, a rigid cylinder with the diameter of 12 mm is placed between the gripper jaws of the forceps. From the simulation results it can be noticed that, the optimized compliant forceps deforms its gripper jaws to fit the shape of the grasped object, as illustrated in Fig. 6-19b). The figure also shows how large stresses emerged at the most deformed part of the forceps.
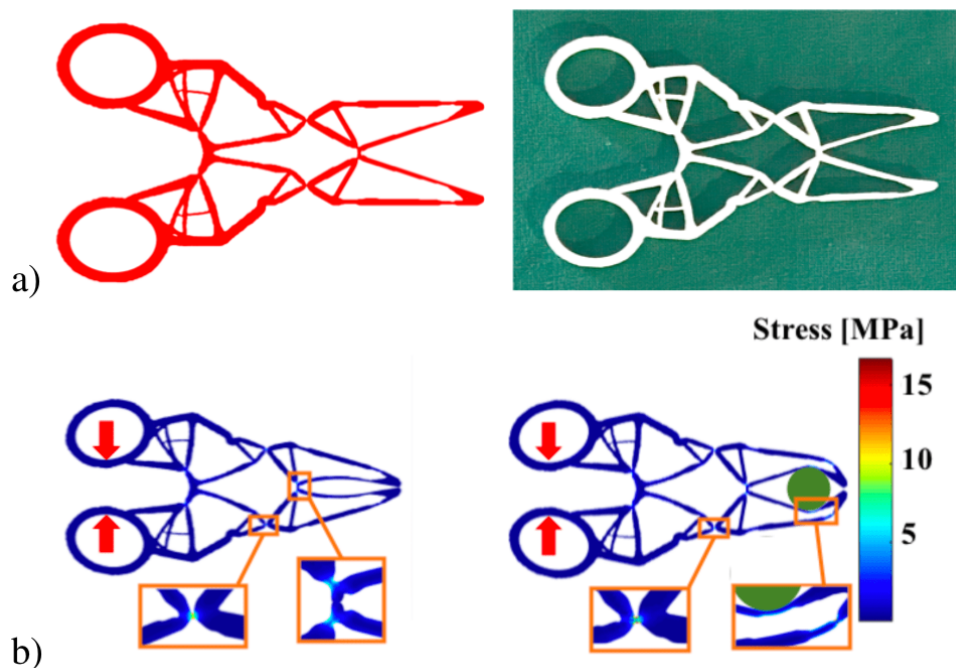

Figure 6-19: Design of the adaptive forceps for open surgery: a) The surface model of the forceps and the 3D-printed prototype, b) FE-simulations of the deformed compliant forceps showing the adaptive grasping function and stress distribution. The red arrows indicate the applied load.

6.3.2 Adaptive Compliant Forceps for Robot-assisted MIS

In this section, different from open surgery forceps, we used our 2D automatic design method to synthesize a cable-driven MIS-forceps head. The design settings are illustrated in Fig. 6-20, while the corresponding design domain is shown in Fig. 6-21. The maximum length and width for initial design domain of the entire forceps head were chosen as 20 mm and 12 mm, respectively. The maximum element size of the FE-meshing was 0.1 mm. The left boundary of the design domain was defined as fixed in the FEA process. The dragging force of the cable was applied on the symmetric axis of the design domain with a distance of 7.5 mm to the left boundary and the displacement-based load F_{in} was set to 2 mm. The values of k_{tissue} and k_{out} were chosen as the one of the forceps realized for open surgery, i.e. 0.75 N/mm and 0.5 N/mm. The contact surface of the gripper jaw was defined as a geometrical constraint Ω_C. The volume fraction g was 0.4. The material property of the forceps and realization process were also the same of the one used for the open surgery forceps presented in the previous section.
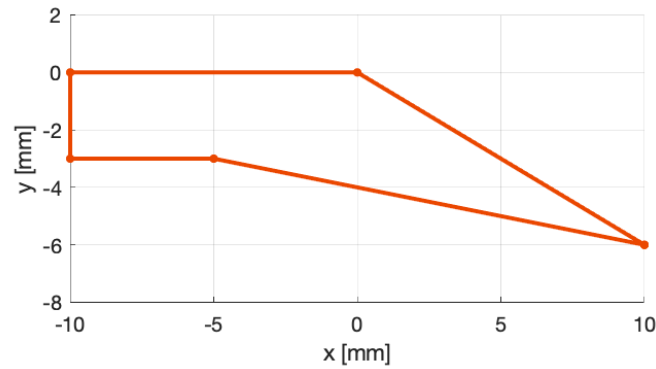
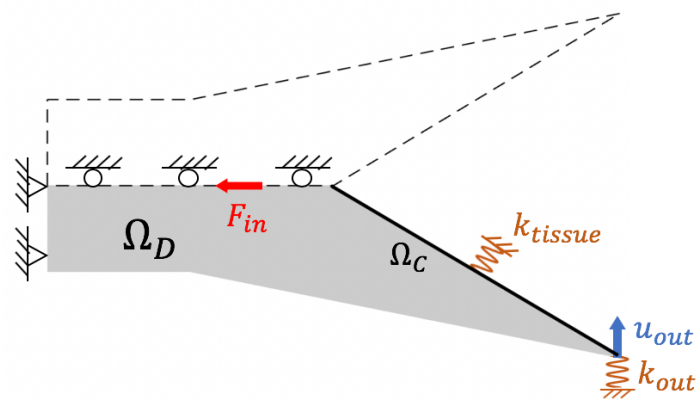**Figure 6-20: The 2D design domain for the design problem in Section 6.3.2.**

Figure 6-21: A schematic diagram illustrating the design problem for the adaptive compliant MIS-forceps.
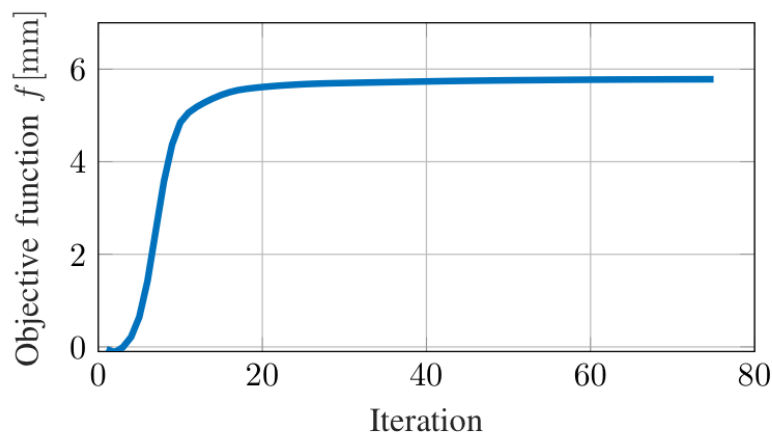

Figure 6-22: The trend of objective function during the optimization process.

The topology optimization process converged at the 75th iteration within a tolerance ε of 0.005. The convergence time was 28.67 s. Fig. 6-22 shows the trend of the objective function f, which has a maximum value of 5.78 mm. The evolution process of the forceps topology is illustrated in Fig. 6-23. FE-simulations were also performed to show the adaptive function of the forceps head. The results are shown in Fig. 6-24. In this case, large stresses also emerged in the thin structure of the flexure hinges. Additionally, the SLS-printed prototype of the adaptive forceps head was embedded into a laparoscopic manipulator and used for grasping phantom organs and tissues. The entire system was realized with our SG Library. A soft robot structure was integrated in the mechanism to increase the dexterity of the manipulator and also to show the

potential of the forceps for robot-assisted MIS. The bending movements of the soft robotic structure were actuated by two rotating wheels in the handle as shown in Fig. 6-25.

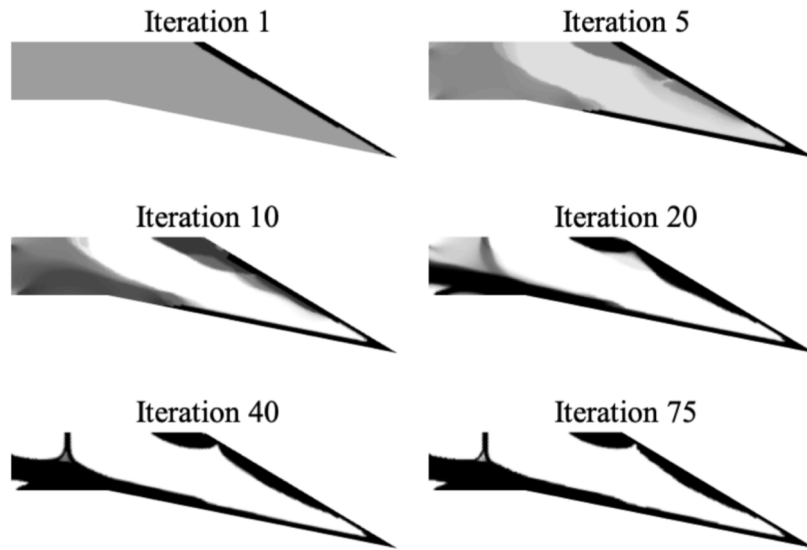**Figure 6-23: The evolution process of the density distribution during some iterations.**
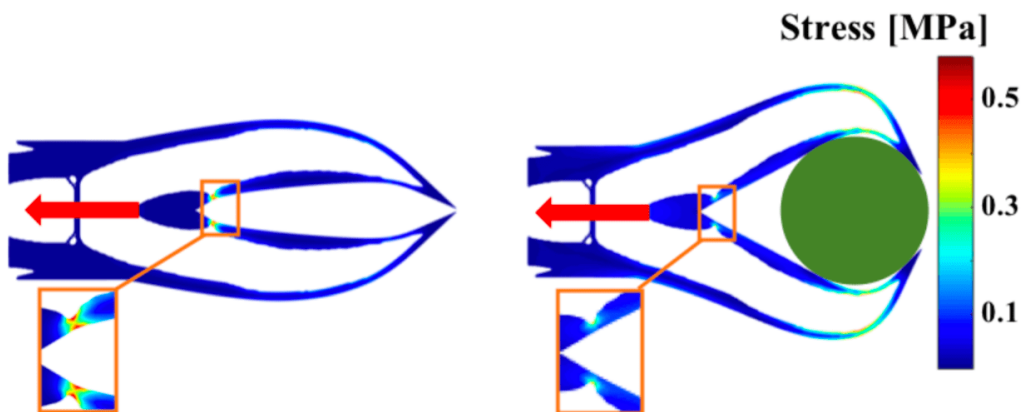

Figure 6-24: FE-simulations of the deformed compliant forceps showing the adaptive grasping function and stress distribution. The red arrows indicate the applied load.
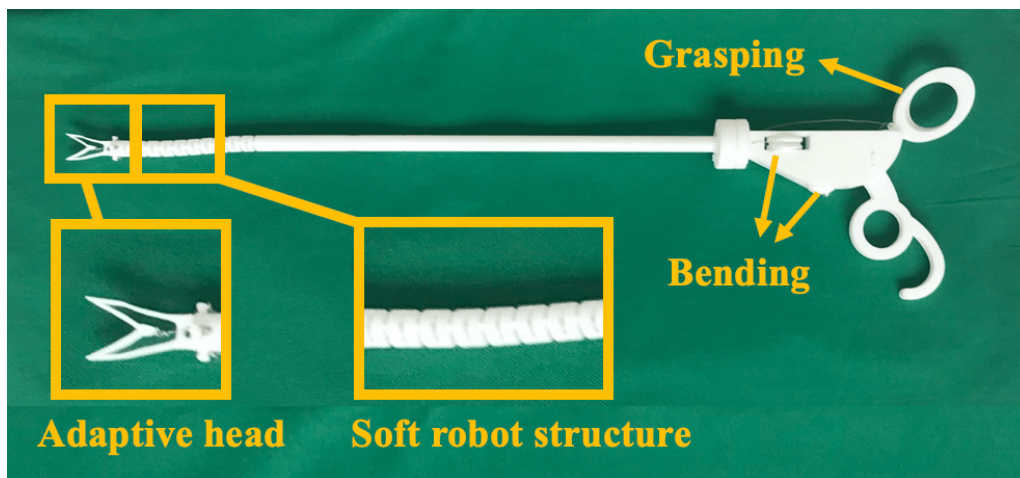

Figure 6-25: SLS-printed laparoscopic manipulator for robotic MIS. Adaptive grasping can be achieved by using the automatically designed forceps.

6.3.3   Testing of the Grasping Performance

This section compares the grasping effect of the realized adaptive compliant forceps with conventional surgical forceps and presents its adaptation in robot-assisted MIS.

Experimental Setup

3D-printed elastic tubes, which emulate the elastic property of organs and tissues, were used to test the grasping effect of our prototypes. The cylindrical tubes were modeled using the SG Library and fabricated with the stereolithography (SLA) printer (Formlabs Form2). A material with silicone-like properties (Elastic Resin), which is suitable for mimicking humans' tissues, was used for printing the parts. A digital microscope (Conrad DP-M17) was used to measure the size of the deformed tubes during the tests. An overview of the experimental setup is shown in Fig. 6-26.
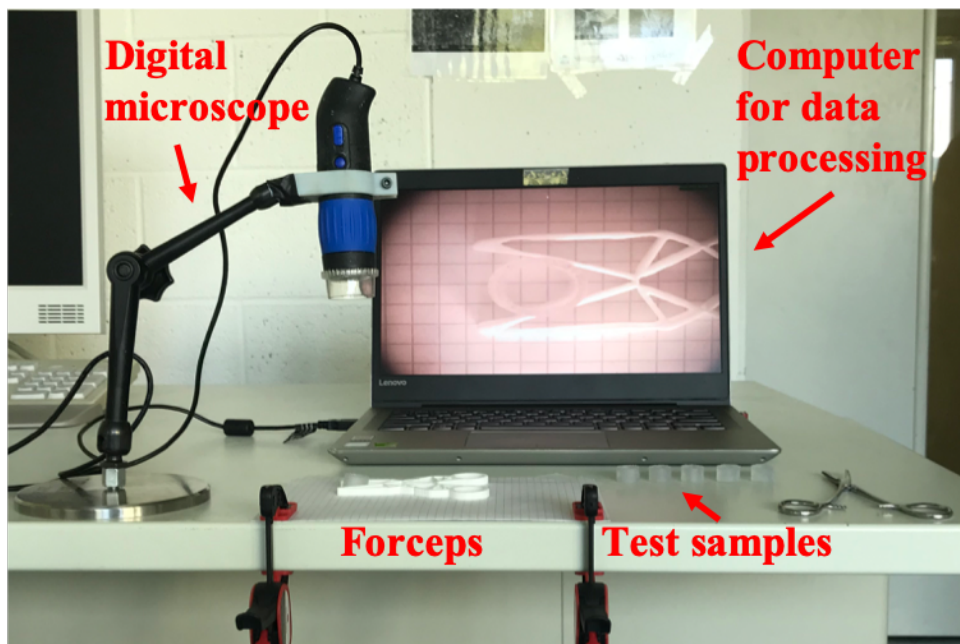

Figure 6-26: Experimental setup: Elastic tubes with different ratios of internal and external diameter are used as test samples to evaluate the grasping effect of conventional and adaptive forceps. A digital microscope measures deformation of the tubes.

The commercially available Cambridge Endo laparoscopic manipulator (Lee, Chamorro and Lee, 2010), which is also equipped with a soft robotic bending mechanism as in the SLS-printed manipulator, and the Kelly Hemostatic forceps, have been used to compare and evaluate the grasping effect of our adaptive compliant forceps (see Fig. 6-27). The tubes were constructed with different ratios of internal and external diameter, k_{ie}, to imitate blood vessels of different thickness. The ratio k_d of the diameter of the deformed tube in x and y axis was calculated to evaluate the grasping effect. In particular:

$$k_{ie} = \frac{d_{int}}{d_{ext}} \qquad (6\text{-}3)$$

$$k_d = \frac{d_{y,deformed\ tube}}{d_{x,deformed\ tube}} \qquad (6\text{-}4)$$
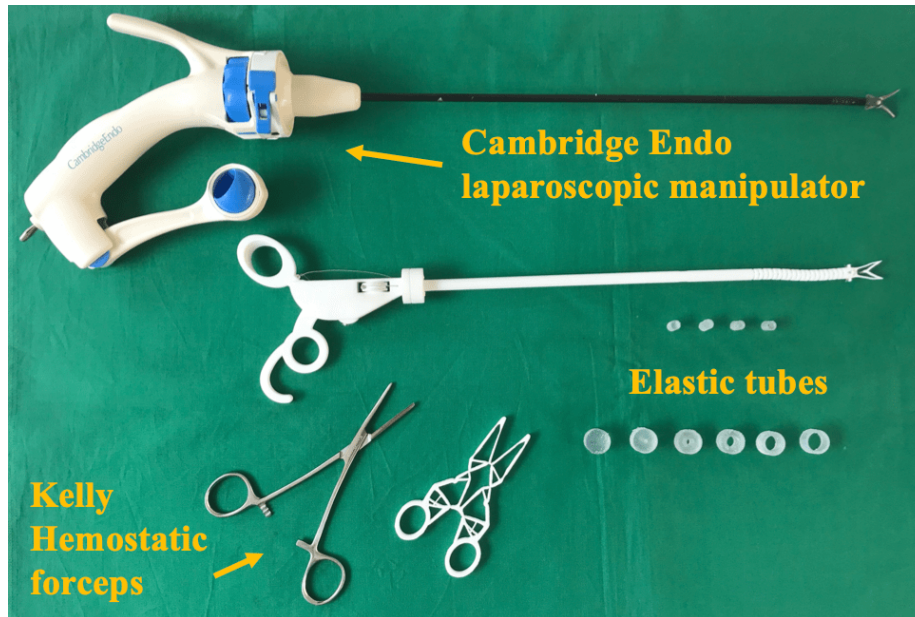
Figure 6-27: Different forceps and elastic tubes used in the experiments.

where the y-axis was defined as the direction in which the gripping force is applied on the tube and the x-axis was its orthogonal axis. The loads were applied on the forceps handle to achieve complete closure of both forceps tips. With conventional forceps, which perform stiff grasping, the elastic tubes were sometimes fully squeezed before complete closure of forceps tips was achieved. For these cases, where the piece prevents full closure of the forceps, k_d was calculated based on the shape of the fully deformed part. The external diameter of the tubes for open surgery forceps and MIS forceps were 15 mm and 6 mm, respectively.
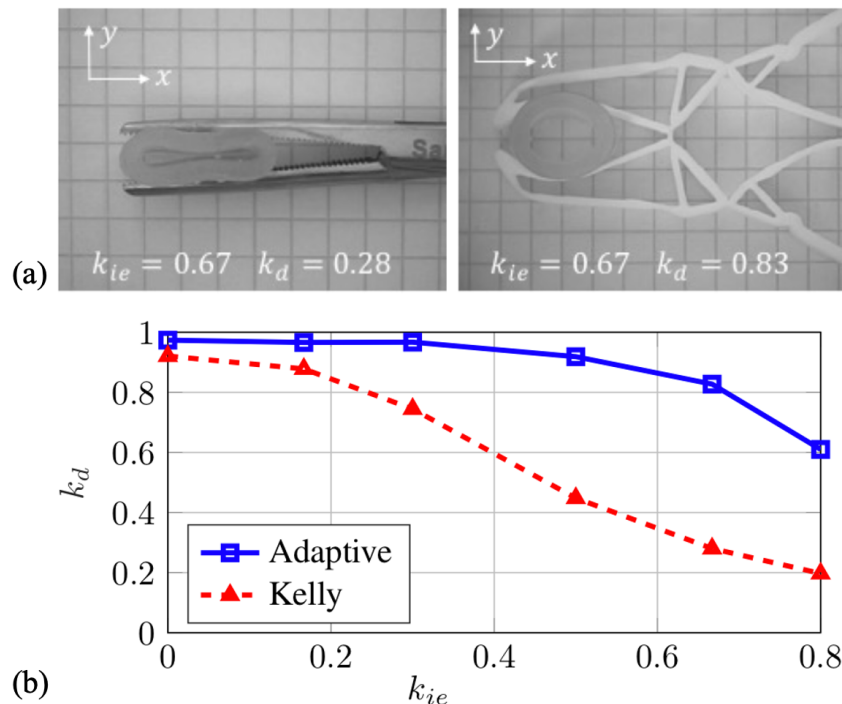

Figure 6-28: Experimental results: a) Experiment of open surgery forceps, b) Trend of the deformations of the tubes over the ratio of the internal and external diameters. The tube is less deformed when using the adaptive forceps (blue) than the Kelly Hemostatic forceps (red).

Experimental Results

Experimental tests were carried out to evaluate the grasping effect of the developed adaptive compliant forceps and compare them with conventional forceps. Each test was repeated three times. Mean value of the results is reported in Fig. 6-28 and 6-29. k_d was higher when the tube was slightly deformed. Hence, the objective of the adaptive grasping function was to make k_d always close to 1, independent from the object to be grasped. As is shown in Fig. 6-28b) and Fig. 6-29b), the k_d values of our adaptive compliant forceps are much closer to 1 than the conventional rigid-joint-based forceps. Even for very thin tubes ($k_{ie} > 0.7$), both compliant forceps were still able to perform adaptive grasping ($k_d > 0.6$). These results haven shown the robustness of the realized adaptive grasping function. While, for very thick tubes ($k_{ie} < 0.2$), which are very stiff and hard to deform, the k_d value of all forceps was close to 1. This situation is similar to the case of diseased and blocked blood vessels.
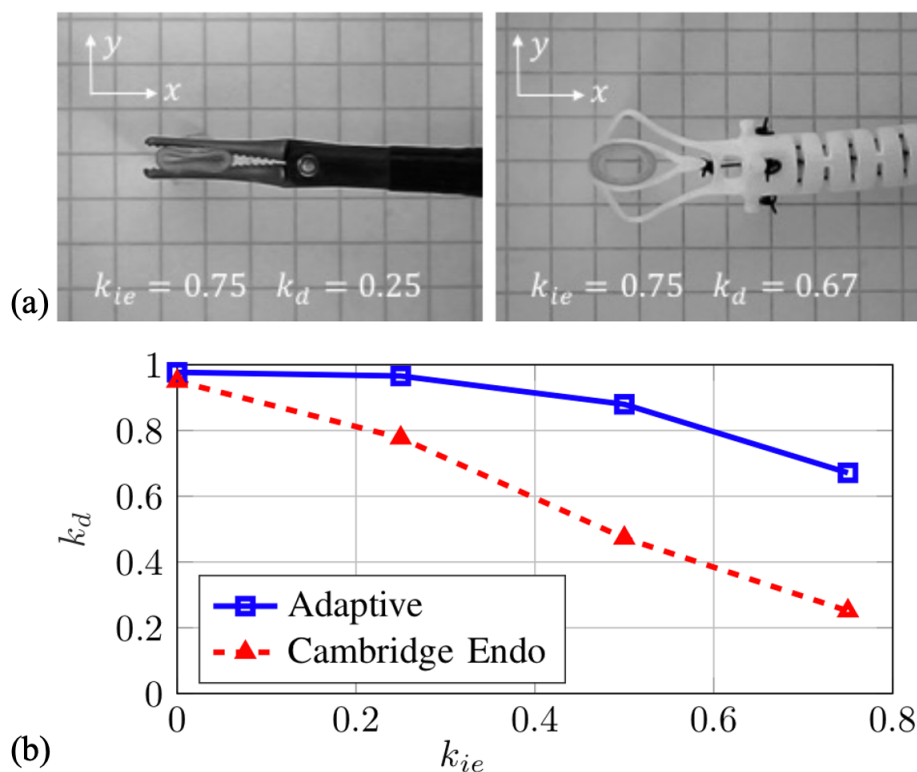

Figure 6-29: Experimental results: a) Experiment of MIS forceps, b) Trend of the deformations of the tubes over the ratio of the internal and external diameters. The tube is less deformed by the adaptive MIS forceps (blue) than the conventional MIS forceps (red).

6.3.4 Endoscopic Experiment in Training Box

To evaluate the grasping effect of the adaptive forceps head of the robot-assisted laparoscopic manipulator, we have also conducted an experiment in a training box equipped with an endoscopic camera (Depstech Endoscope, China). The experimental setup is shown in Fig. 6-30a). An artificial artery printed with the SLA-printer Form2 and imitating the material properties of the soft tissues, was used during the test. Grasping of the elastic artery in the training box was performed with the Cambridge Endo laparoscopic manipulator and our SLS-printed manipulator. With the equipped soft robot structure, both manipulators can also perform bending movements during grasping. The proposed MIS forceps could successfully perform adaptive grasping in the training box while the conventional forceps squeezed the artificial artery completely,

as shown in Fig. 6-30b). The adaptive grasping function of our robot-assisted MIS forceps was thus, also successfully verified in a MIS setup.

Figure 6-30: Endoscopic experiment in a training box: a) Setup of the test composed of a training box, an endoscopic camera, a laparoscopic manipulator and a monitor, b) Images taken by the endoscopic camera showing the grasping effect of the conventional and adaptive MIS forceps.

6.4 Multifunctional Compliant Forceps Combining Stiff and Adaptive Grasping Functions

In this section, we aimed to synthesize a MIS-forceps that combines both the stiff and adaptive grasping function. The motivation is that, in Fig. 6-19b) and Fig. 6-24, it can be seen that the realized adaptive forceps still have a big gap between the jaws when fully closed, even without grasping an object. This phenomenon could lead to grasping failure when the object is small. We aim to solve this problem by using the multi-objective formulation of our 3D topology optimization method.

Fig. 6-31 shows the two loading cases that are involved in the 3D design problem. In the first loading case (see Fig. 6-31a)), the design domain was comprised of $60 \times 20 \times 20$ cubic elements with $l_e = 0.375 \ mm$. The distance between the objective point $\boldsymbol{P_{out}}$ and the bottom of the design domain was 1.875 mm while the volume constraint g was set to 0.08. The surface on the x-y plane and the x-z plane were defined as symmetric surfaces and all the boundary conditions were also mirrored. The displacement-based load F_a was 5 mm. Herein, the first loading case was set in order to realize a stiff-grasping forceps. In the second loading case (see Fig. 6-31b)), a linear spring was applied on the middle of the gripper jaw to realize the adaptive grasping function, while the material properties were identical to the case in Section 6.3 since

it was also SLS-printed using polyamide. The weighting factors ω_l of the two cases were both 0.5.
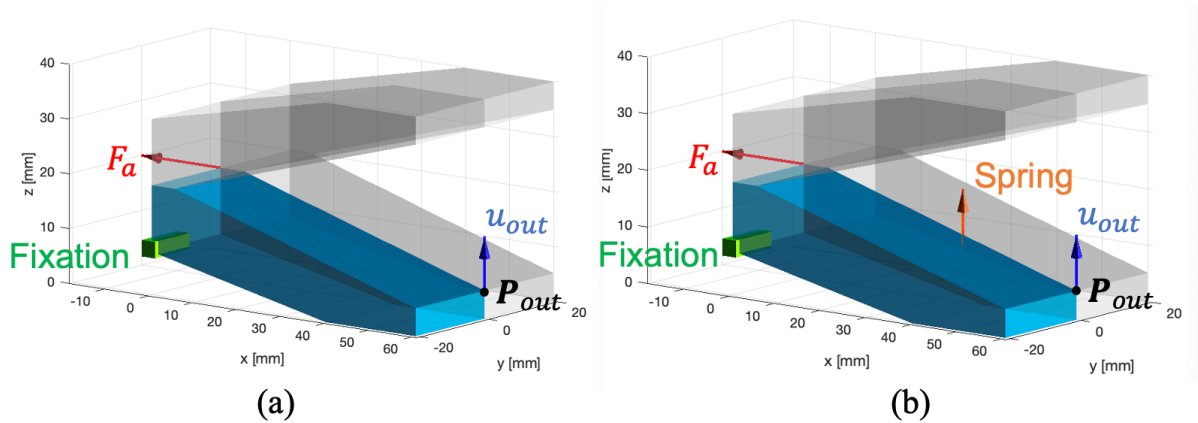

(a)   (b)

Figure 6-31: Multiple loading cases in the design problem: a) The first loading case for a realizing stiff grasping forceps, b) The second loading case for realizing an adaptive grasping forceps.
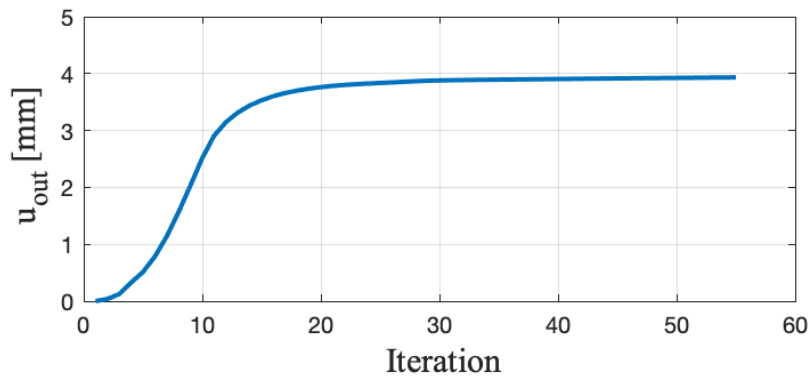

Figure 6-32: The trend of the output displacement during the optimization process.
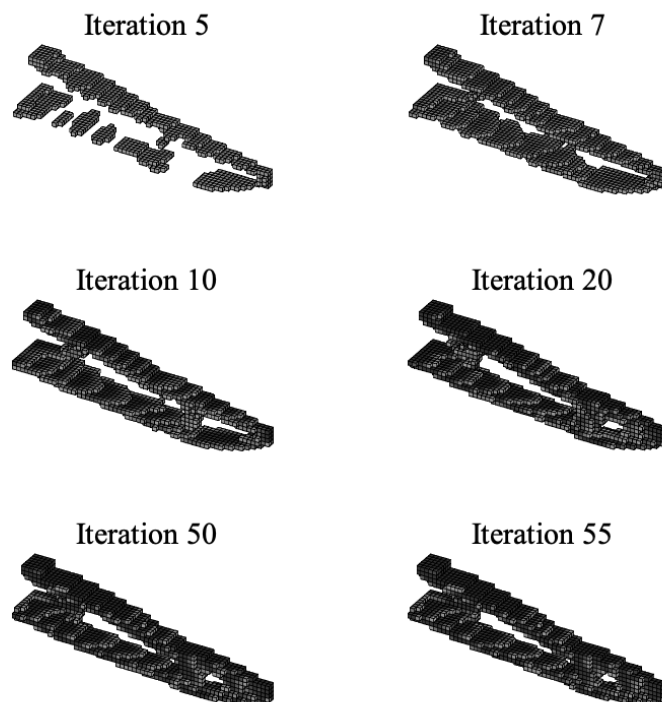

Figure 6-33: The evolution process of the design variable in some iterations.

Fig. 6-32 shows the trend of u_{out} during the optimization process. It can be seen that synthesis process converged at the 55th iteration, reaching a maximum u_{out} of 3.93 mm. The entire synthesis process took 186.6 seconds. From the trend of u_{out} and the evolution process of the design variable $\boldsymbol{\rho}$ (see Fig. 6-33), we can see that the main structure of the gripper has already emerged in the 20th iteration. Fig. 6-34a) shows the post-processed 3D model of the MIS-forceps where a connector is integrated to mount the forceps onto a commercially available laparoscopic manipulator. In this case, the manipulator of Karl Storz 33500 M (Karl Storz GmbH, Germany) was used, as is shown in Fig. 6-34b).
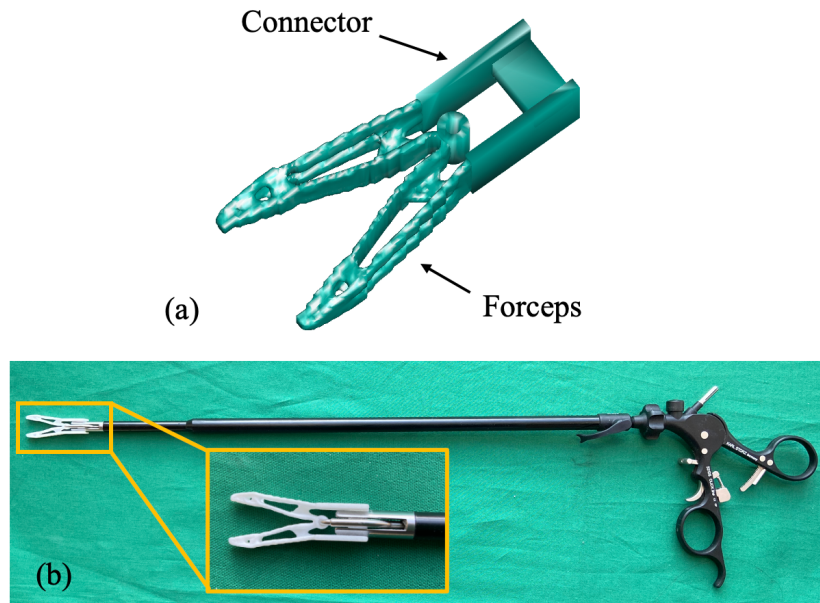
Figure 6-34: The realized MIS-forceps with both stiff and adaptive grasping functions: a) The 3D surface model of the realized forceps, b) The SLS-printed forceps which is mounted onto a laparoscopic manipulator.
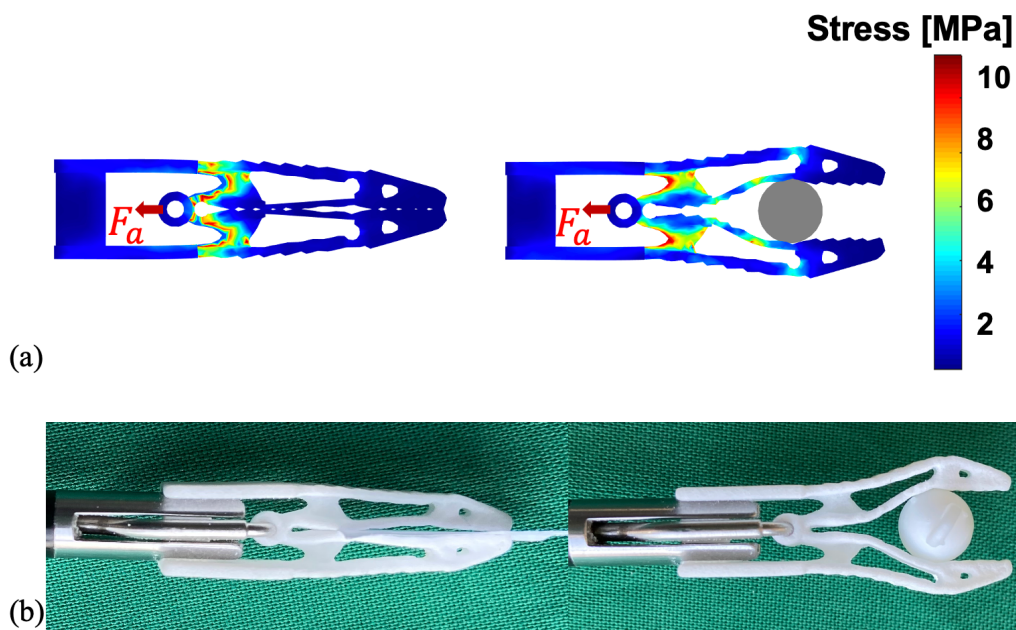
Figure 6-35: Evaluation of the grasping performance of the realized MIS-forceps: a) FE-simulation results of the forceps grasping a piece of paper (left) and a rigid sphere (right), b) Experimental test of the printed forceps grasping a piece of paper (left) and a rigid sphere (right).

To evaluate the grasping performance of the realized forceps, FE-simulations and grasping tests were conducted. In the tests, the forceps was actuated by the manipulator to grasp a piece of paper and a rigid sphere for comparison. As is shown in Fig. 6-35, both the simulation and test results indicate that, the paper can be tightly held without gap while the sphere can be grasped adaptively. Therefore, the stiff and adaptive grasping function of the realized forceps are verified.

6.5 Bionic Prosthetic Finger

In this section, we use the proposed 3D topology optimization method to synthesize a compliant prosthetic finger. Since the continuum-structure-based topology optimization method is inspired from biological evolution, the realized compliant finger also has a bionic design.

6.5.1   Synthesis and Fabrication of the Prosthetic Finger

Fig. 6-36 is a schematic diagram showing the design problem of the compliant finger. It can be seen that, the design domain is a blue cuboid comprised of $80 \times 20 \times 10$ cubic elements with $l_e = 0.5\ mm$. The design objective was to maximize the tip displacement u_{out}. Since the prosthetic finger is supposed to have a symmetric structure, we defined the right surface of the design domain (the x-y plane in Fig. 6-36) as the symmetric surface and all the boundary conditions were also mirrored. In this way, the predefined design domain was used to synthesize only half of the finger, which reduced the computational cost greatly. A displacement-based push force F_a of 5 mm was applied on the design domain to actuate the compliant finger, while the blue arrow indicated the output displacement u_{out} . $k_{obj} = 0.75N/mm$ and $k_{out} = 0.5N/mm$ were linear springs applied on the middle and end of the finger in x-axis, in order to imitate the grasping resistance on the finger joint and tip, respectively. The green domain depicted the fixation boundary condition. Since the compliant finger was SLS-printed with the polyamide (PA2200), which is a linearly elastic material, the Young's modulus E_0 and Poisson's ratio v were set to 1700 MPa and 0.3 according to its data sheet.
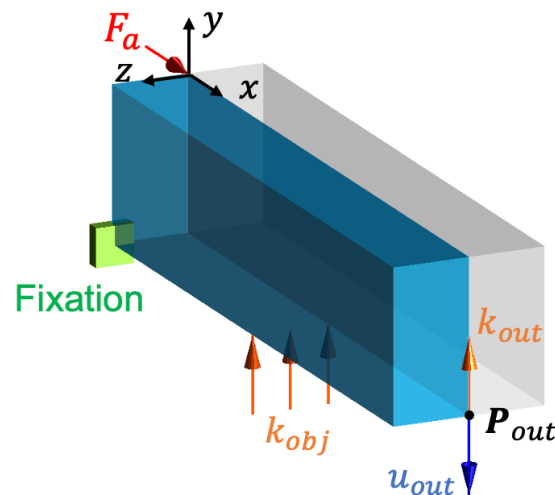
Figure 6-36: Schematic diagram illustrating the design problem of the prosthetic finger.

Fig. 6-37 shows the evolution process of the density distribution to achieve the final topology, where the objective function converged at the 100th iteration. The maximum u_{out} has reached 11.64 mm, as is shown in the diagram in Fig. 6-38. The entire design process took 123.6 seconds. After post-processing, a symmetrical and monolithic bionic finger was realized (see Fig. 6-39a), which was then SLS-printed for testing the bending performance. In the prototype, a

flange was also constructed (see Fig. 6-39b) so that the finger could be fixed and then actuated by servo motor.
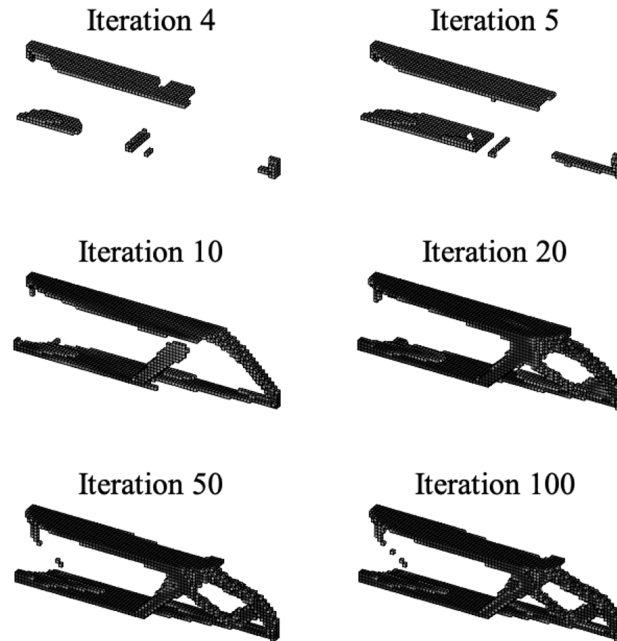
Figure 6-37: The evolution of the density distribution x during the 3D topology optimization process, where x converged at the 100ᵗʰ iteration. The elements with $x_e > 0.5$ are plotted and the black color represents solid element in the design domain.
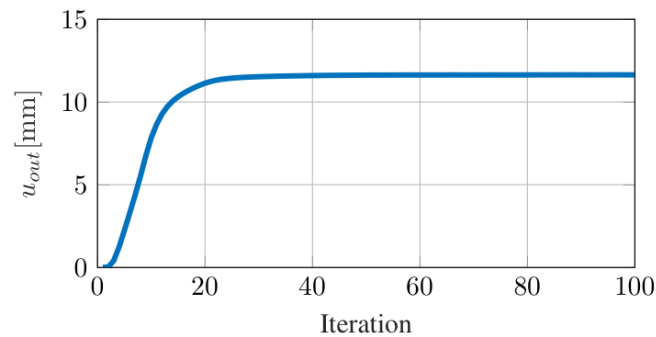
Figure 6-38: The value of u_{out} in the optimization process.
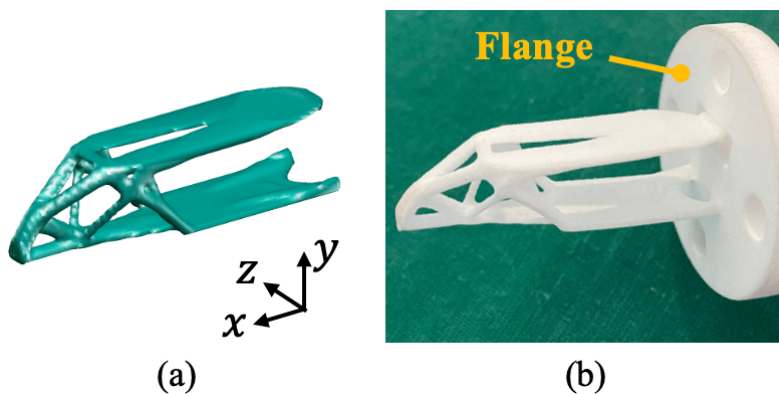
Figure 6-39: The realized prosthetic finger: a) The surface model, b) The SLS-printed prototype.

6.5.2   FEM-Based Simulation and Payload Test

Large-displacement FEA was conducted to analyze the stress distribution of the realized bionic finger during the bending movement. From the simulation result in Fig. 6-40a), it can be noticed that the most deformations and stresses are evenly located in the second half of the finger, which could effectively prevent the mechanical fatigue. The bent prototype in Fig. 6-40b) has also verified the deformed shape.

(a)   (b)
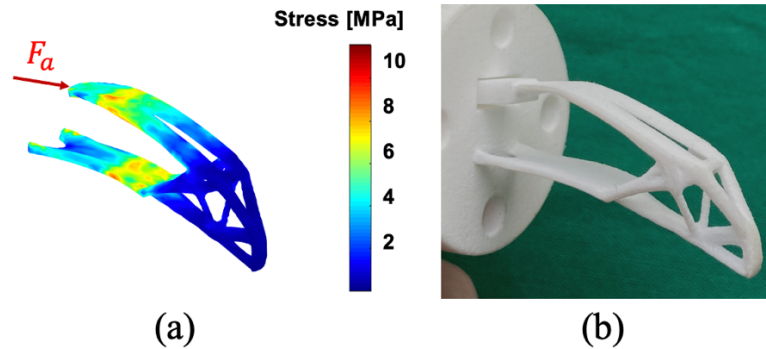
Figure 6-40: The bent finger with a displacement-based actuation force F_a of 5 mm: a) The FE-simulated stress distribution, b) The bent prototype.
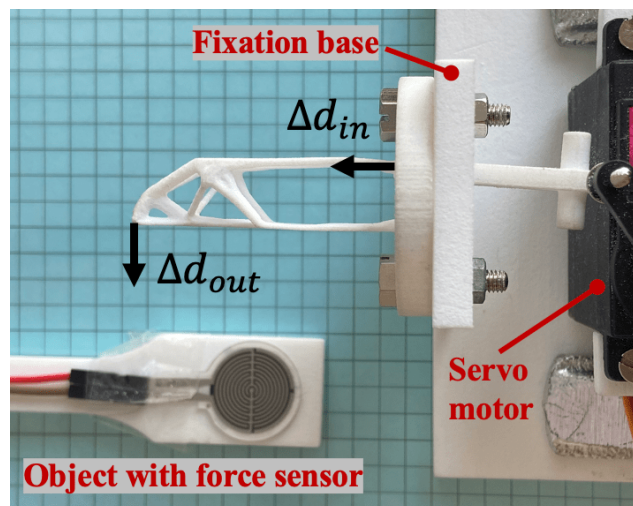
Figure 6-41: The experimental setup of the payload test. The force sensor (Film Sensor Technology, China) attached on the object was used to keep the touching force F_t constant.
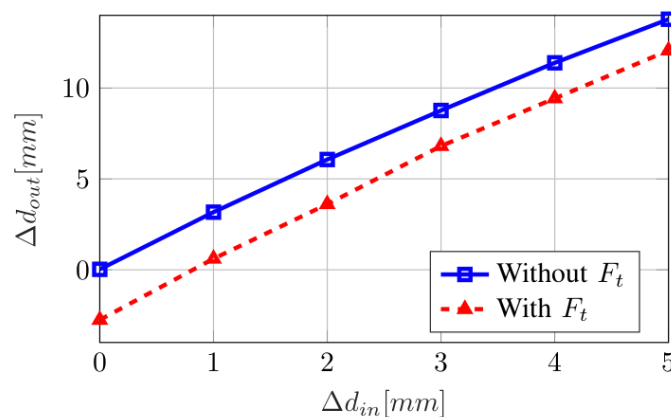
Figure 6-42: Trend of the measured fingertip displacements Δd_{out} over the input displacements Δd_{in} (with and without F_t).

A payload test was also carried out to evaluate the robustness of the realized finger when touching an object. The experimental setup is shown in Fig. 6-41, where the printed compliant finger was mounted on a platform and actuated by a servo motor to realize the bending movement. In the payload test, a sensor-controlled constant touching force F_t was applied at the tip of the finger while the finger was bent by the actuation force. F_t was chosen as 2 N to imitate the normal touching force of the human finger. The vertical displacement Δd_{out} of the fingertip and the displacement Δd_{in} of the actuation point were measured to analyze the touching force's influence on the bending performance. For comparison, Δd_{in} and Δd_{out} were also measured when no touching force was applied.

The experiment results are reported in Fig. 6-42. It can be seen that, Δd_{out} increases almost linearly with Δd_{in}. For the untouched (blue curve) and touched (red curve) cases, the largest difference in Δd_{out} is about 2.8 mm, which occurs when $\Delta d_{in} = 0mm$. This small difference in Δd_{out} shows that the realized compliant finger is stable and robust against the normal touching force. The feasibility of our automatic method for designing the bionic prosthetic finger was thus successfully verified.

6.6 Conclusion of the Chapter

In this chapter, 5 design cases were presented to verify the feasibility of our automatic design framework for designing compliant medical instruments. The realized instruments were all SLS-printed using the polyamide as material. FEM-based simulations and experimental tests were also carried out to evaluate the mechanical performance of the realized instruments. Results demonstrated that the design process of compliant medical instruments can be greatly simplified by using our design framework.

7 Conclusions and Outlook

In this chapter, the work in this dissertation is summarized. Firstly, the developed modeling and automatic synthesis methods for compliant medical instruments are discussed, in terms of their efficiency and performance. Then, future work is also outlined for further improvement.

7.1 FEM-Based Modeling Methods

As a prerequisite for implementing the bionic structural optimization methods, a FEM-based modeling framework was developed in MATLAB to analyze the mechanics of bio-inspired compliant instruments, as is described in Chapter 3. The modeling framework was comprised of a geometry modeling tool and a non-linear FEM formulation that achieves high-fidelity modeling of large displacements, tendon-driven mechanisms and contact problems. The modeling of a snake-like continuum manipulator, a flytrap-like compliant forceps and a fish-fin-inspired adaptive finger have been presented to show the performance of the proposed methods in modeling 3D-printed compliant medical robots and devices. Simulation and experimental results have demonstrated the accuracy and plausibility of the presented framework.

The presented modeling methods outperform the previous work (Camarillo *et al.*, 2008; Huang *et al.*, 2018), in which the PCCM and PRBM method are used for simplifying the geometries and the realized modeling frameworks are only applicable to the specific models. Besides, the proposed non-linear FEM formulation has also integrated the contact modeling by using the node-to-facet contact search method. Compared to the node-to-node search method in Coevoet, Escande and Duriez, 2017, the node-to-facet formulation is more efficient for solving sophisticated surface contact problems which frequently occur in the bio-inspired compliant mechanisms. On the other hand, although many commercial FEM software can also perform robust modeling of compliant mechanisms, they have the disadvantage of being a closed black-box where the source code cannot be directly modified by the user. Hence, the implementation of new modeling methods or new optimization algorithms often requires the integration of different developing environments via additional interfaces, as in Liu and Chiu, 2017; Runge *et al.*, 2017; Chen *et al.*, 2019. In this work, all modeling methods are implemented in MATLAB and can be easily extended, which greatly improves the modeling efficiency.

7.2 Topology Optimization Based Synthesis Methods

To realize the automatic synthesis of compliant medical instruments, continuum-structure-based topology optimization methods were incorporated into our design framework. Since the proposed topology optimization methods are inspired from biological evolution, the realized instruments also have a bionic design. To deeply explore the synthesis potential, the 2D and 3D version of the topology optimization method were both implemented in our design framework (see Chapter 4). While the 3D optimization method is generally slower than its 2D version, it can achieve more optimized design details in the third dimension. Besides, some design parameters, such as the mesh size and volume fraction, also play an important role in achieving the ideal design results (see Chapter 5). In order to evaluate the performance of the developed synthesis methods for designing compliant medical instruments, 5 design examples were presented in Chapter 6 with different design scenarios. Their mechanical performance was successfully verified by FE-simulations and experimental tests.

The proposed automatic design framework greatly simplifies the synthesis process of complaint medical instruments, since the user only needs to provide a design domain, a design objective and the corresponding boundary conditions to start the design process. The final design proposal

can be automatically achieved by using the implemented topology optimization methods. Besides, the proposed synthesis method is easy to extend and modify due to the unified design framework. The synthesis cases in Chapter 6 have shown that, the proposed design framework is also very agile. As is presented in Section 6.1 and Section 6.3, it is possible to convert a stiff-grasping forceps into an adaptive-grasping forceps by adding a spring into the design problem. Different actuation mechanisms could also be realized by defining different loading cases, as can be seen in Section 6.2. Therefore, it is possible to use the presented method to efficiently realize task-specific and patient-specific compliant instruments.

7.3 Limitations and Future Work

Nevertheless, the automatic design framework can still be improved in several aspects.

For the FEM-based modeling methods, as is shown in Fig. 3-6, the PDE Toolbox always generates a tetrahedral mesh with a constant element size, which provides less accurate modeling results compared to the hexahedral mesh, and also makes the simulation computationally expensive when the thin flexure hinges and the thick parts of the compliant structure have the same element size. This problem could be solved by introducing hexahedral elements and variable element size into the meshing algorithm of the PDE Toolbox, and will be further analyzed in the future. On the other hand, beside the displacement measurement, fatigue tests should also be carried out to evaluate the plausibility of the simulation results. In future work, the author plans to integrate the modeling of other mechanisms, such as the pneumatic actuation using hyperelastic materials, into the non-linear FEM formulation.

For the topology optimization based synthesis methods, additional stress constraints could be introduced into the design problem to prevent the thin flexure hinges from being created in the final topologies, as they could lead to large stresses or even mechanical fatigue of the realized instruments. Since the latter part of the optimization process often made little contribution to the final topology (see Fig. 6-18b) and Fig. 6-23 for example), the author will refine the convergence criterion and the move limit m in the future so that the optimization process could converge faster to reduce the computational cost. On the other hand, the author utilized the linear FEA in each step of the optimization algorithm to reduce the computational cost. Although the realized instruments have been proven to be functional, it is still worthwhile to incorporate the non-linear FEA into the optimization algorithm to explore the non-linear feature. In future work, the author plans to use other materials, such as titan and nitinol, to fabricate the realized medical instruments and deeply explore the benefit of using the proposed automatic design method in different medical applications.

Bibliography

Aguib, H. (2011) 'Adaptive Mechanismen zum sicheren Greifen und Fügen von Gewebe in der laparoskopischen Operationstechnik', *Dissertation*, Fakultät für Maschinenwesen, Technische Universität München.

Andreassen, E. *et al.* (2011) 'Efficient topology optimization in MATLAB using 88 lines of code', *Structural and Multidisciplinary Optimization*, 43(1), pp. 1–16. doi: 10.1007/s00158-010-0594-7.

Awtar, S. and Sen, S. (2010) 'A Generalized Constraint Model for Two-Dimensional Beam Flexures: Nonlinear Load-Displacement Formulation', *Journal of Mechanical Design*, 132(8). doi: 10.1115/1.4002005.

Awtar, S., Slocum, A. H. and Sevincer, E. (2006) 'Characteristics of Beam-Based Flexure Modules', *Journal of Mechanical Design*, 129(6), pp. 625–639. doi: 10.1115/1.2717231.

Baek, C., Yoon, K. and Kim, D.-N. (2016) 'Finite element modeling of concentric-tube continuum robots', *Structural Engineering and Mechanics*, 57(5), pp. 809–821.

Bendsøe, M. P. (1989) 'Optimal shape design as a material distribution problem', *Structural Optimization*, 1, pp. 193–202. doi: 10.1007/BF01650949.

Bendsoe, M. P. and Sigmund, O. (2003) *Topology Optimization: Theory, Design and Applications*. Springer-Verlag Berlin Heidelberg.

Bernardoni, P. *et al.* (2004) 'A new compliant mechanism design methodology based on flexible building blocks', in Smith, R. C. (ed.) *Smart Structures and Materials 2004: Modeling, Signal Processing, and Control*. SPIE, pp. 244–254. doi: 10.1117/12.539498.

Bieze, T. M. *et al.* (2018) 'Finite Element Method-Based Kinematics and Closed-Loop Control of Soft, Continuum Manipulators', *Soft Robotics*, 5(3), pp. 348–364. doi: 10.1089/soro.2017.0079.

De Borst, R. *et al.* (2012) *Nonlinear finite element analysis of solids and structures*. John Wiley & Sons.

Botsch, M. *et al.* (2010) *Polygon mesh processing*. CRC press.

Bourdin, B. (2001) 'Filters in topology optimization', *International Journal for Numerical Methods in Engineering*, 50(9), pp. 2143–2158. doi: https://doi.org/10.1002/nme.116.

Camarillo, D. B. *et al.* (2008) 'Mechanics Modeling of Tendon-Driven Continuum Manipulators', *IEEE Transactions on Robotics*, 24(6), pp. 1262–1273. doi: 10.1109/TRO.2008.2002311.

Chen, F. *et al.* (2019) 'Automatic Design of Soft Dielectric Elastomer Actuators With Optimal Spatial Electric Fields', *IEEE Transactions on Robotics*. IEEE, PP(June), pp. 1–16. doi: 10.1109/tro.2019.2920108.

Chen, Y.-H. and Lan, C.-C. (2012) 'An Adjustable Constant-Force Mechanism for Adaptive End-Effector Operations', *Journal of Mechanical Design*, 134(3). doi: 10.1115/1.4005865.

Coevoet, E., Escande, A. and Duriez, C. (2017) 'Optimization-Based Inverse Model of Soft Robots With Contact Handling', *IEEE Robotics and Automation Letters*, 2(3), pp. 1413–1419. doi: 10.1109/LRA.2017.2669367.

Edmondson, B. J. *et al.* (2013) 'Oriceps: Origami-Inspired Forceps'. doi: 10.1115/SMASIS2013-3299.

Frecker, M. I. *et al.* (1997) 'Topological Synthesis of Compliant Mechanisms Using Multi-Criteria Optimization', *Journal of Mechanical Design*, 119(2), pp. 238–245. doi: 10.1115/1.2826242.

Frecker, M. I., Dziedzic, R. P. and Haluck, R. S. (2002) 'Design of multifunctional compliant mechanisms for minimally invasive surgery', *Minimally Invasive Therapy & Allied Technologies*, 11(5–6), pp. 311–319. doi: 10.1080/13645706.2003.11873732.

Gallego, J. A. and Herder, J. (2009) 'Synthesis methods in compliant mechanisms: An overview', *Proceedings of the ASME Design Engineering Technical Conference*, 7(PARTS A AND B), pp. 193–214. doi: 10.1115/DETC2009-86845.

Gonenc, B. *et al.* (2013) 'Design of 3-DOF force sensing micro-forceps for robot assisted vitreoretinal surgery', in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, pp. 5686–5689. doi: 10.1109/EMBC.2013.6610841.

Gravagne, I. A., Rahn, C. D. and Walker, I. D. (2003) 'Large deflection dynamics and control for planar continuum robots', *IEEE/ASME Transactions on Mechatronics*, 8(2), pp. 299–307. doi: 10.1109/TMECH.2003.812829.

Gravagne, I. A. and Walker, I. D. (2000) 'On the kinematics of remotely-actuated continuum robots', in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, pp. 2544–2550 vol.3. doi: 10.1109/ROBOT.2000.846411.

Greigarn, T. and Çavuşoğlu, M. C. (2015) 'Pseudo-rigid-body model and kinematic analysis of MRI-actuated catheters', in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2236–2243. doi: 10.1109/ICRA.2015.7139495.

Groenwold, A. A. and Etman, L. F. P. (2008) 'On the equivalence of optimality criterion and sequential approximate optimization methods in the classical topology layout problem', *International Journal for Numerical Methods in Engineering*, 73(3), pp. 297–316. doi: https://doi.org/10.1002/nme.2071.

Guthart, G. S. and Salisbury, K. J. (2000) 'Intuitive telesurgery system: overview and application', in *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 618–621.

Hao, G., Kong, X. and Reuben, R. L. (2011) 'A nonlinear analysis of spatial compliant parallel modules: Multi-beam modules', *Mechanism and Machine Theory*, 46(5), pp. 680–706. doi: https://doi.org/10.1016/j.mechmachtheory.2010.12.007.

Hoffmann, C. M. (1989) *Geometric and Solid Modeling: An Introduction, Geometric and Solid Modeling: An Introduction.* Available at: http://dl.acm.org/citation.cfm?id=74803.

Howell, L. L. and Midha, A. (1994) 'A method for the design of compliant mechanisms with small-length flexural pivots', *Journal of Mechanical Design, Transactions of the ASME*, 116(1), pp. 280–290. doi: 10.1115/1.2919359.

Howell, L. L. and Midha, A. (1995) 'Parametric Deflection Approximations for End-Loaded, Large-Deflection Beams in Compliant Mechanisms', *Journal of Mechanical Design*, 117(1), pp. 156–165. doi: 10.1115/1.2826101.

Hu, Y. *et al.* (2019) 'Design and Fabrication of a 3D Printed Metallic Flexible Joint for Snake-Like Surgical Robot', *IEEE Robotics and Automation Letters*. IEEE, PP(January), p. 1. doi: 10.1109/LRA.2019.2896475.

Huang, S. *et al.* (2018) 'Statics of Continuum Space Manipulators With Nonconstant Curvature via Pseudorigid-Body 3R Model', *IEEE Access*, 6, pp. 70854–70865. doi: 10.1109/ACCESS.2018.2881261.

Intuitive Surgical Inc. (2020) *Da Vinci Vessel Sealer Extend*. Available at: https://www.intuitive.com/en-us/products-and-services/da-vinci/energy/vessel-sealer-extend (Accessed: 30 November 2019).

Jones, B. A. and Walker, I. D. (2006) 'Kinematics for multisection continuum robots', *IEEE Transactions on Robotics*, 22(1), pp. 43–55. doi: 10.1109/TRO.2005.861458.

Jutte, C. V and Kota, S. (2007) 'Design of Planar Nonlinear Springs for Prescribed Load-Displacement Functions', pp. 161–170. doi: 10.1115/DETC2007-35535.

Kim, C. J., Kota, S. and Moon, Y.-M. (2005) 'An Instant Center Approach Toward the Conceptual Design of Compliant Mechanisms', *Journal of Mechanical Design*, 128(3), pp. 542–550. doi: 10.1115/1.2181992.

Kim, N.-H. (2014) *Introduction to Nonlinear Finite Element Analysis*. Springer Science & Business Media.

Kota, S. *et al.* (2005) 'Design and application of compliant mechanisms for surgical tools', *Journal of Biomechanical Engineering*, 127(6), pp. 981–989. doi: 10.1115/1.2056561.

Krieger, Y. S. *et al.* (2017) 'Fatigue strength of laser sintered flexure hinge structures for soft robotic applications', in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1230–1235. doi: 10.1109/IROS.2017.8202297.

Lan, C. and Wang, J. (2011) 'Design of adjustable constant-force forceps for robot-assisted surgical manipulation', in *2011 IEEE International Conference on Robotics and Automation*, pp. 386–391. doi: 10.1109/ICRA.2011.5979556.

Lan, C., Wang, J. and Chen, Y. (2010) 'A compliant constant-force mechanism for adaptive robot end-effector operations', in *2010 IEEE International Conference on Robotics and Automation*, pp. 2131–2136. doi: 10.1109/ROBOT.2010.5509928.

de Lange, D. J. B. A., Langelaar, M. and Herder, J. L. (2008) 'Towards the design of a statically balanced compliant laparoscopic grasper using topology optimization', in *ASME 2008 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pp. 293–305.

Lee, Woojin, Chamorro, A. and Lee, Woojoong (2010) 'Surgical instrument'. US.

Liu, C. and Chiu, C. (2017) 'Optimal Design of a Soft Robotic Gripper with High Mechanical Advantage for Grasping Irregular Objects', in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2846–2851.

Liu, K. and Tovar, A. (2014) 'An efficient 3D topology optimization code written in Matlab', *Structural and Multidisciplinary Optimization*, 50(6), pp. 1175–1196. doi: 10.1007/s00158-014-1107-x.

Lorensen, W. E. and Cline, H. E. (1987) 'Marching Cubes: A High Resolution 3D Surface Construction Algorithm', in *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*. New York, NY, USA: Association for Computing Machinery (SIGGRAPH '87), pp. 163–169. doi: 10.1145/37401.37422.

Lueth, T. C. (2015) 'SG-Library: Entwicklung einer konstruktiven MATLAB-Toolbox zur räumlichen Modellierung von Körpern, Gelenken und Getrieben', in *11. Kolloquium Getriebetechnik*, pp. 183–203. doi: 10.14459/2015md1276136.

Mahl, T., Hildebrandt, A. and Sawodny, O. (2014) 'A Variable Curvature Continuum Kinematics for Kinematic Control of the Bionic Handling Assistant', *IEEE Transactions on Robotics*, 30(4), pp. 935–949. doi: 10.1109/TRO.2014.2314777.

Morimoto, T. K., Hawkes, E. W. and Okamura, A. M. (2017) 'Design of a Compact Actuation and Control System for Flexible Medical Robots', *IEEE Robotics and Automation Letters*, 2(3), pp. 1579–1585. doi: 10.1109/LRA.2017.2676240.

Murphy, M. D., Midha, A. and Howell, L. L. (1996) 'The topological synthesis of compliant mechanisms', *Mechanism and Machine Theory*, 31(2), pp. 185–199. doi: https://doi.org/10.1016/0094-114X(95)00055-4.

Pereira, A. *et al.* (2011) 'An efficient and compact Matlab implementation of topology optimization: Application to compliant mechanism', *Proceedings of the XXXII Iberian Latin American Congress On Computational Methods In Engineering*.

Pucheta, M. A. and Cardona, A. (2010) 'Design of bistable compliant mechanisms using precision–position and rigid-body replacement methods', *Mechanism and Machine Theory*, 45(2), pp. 304–326. doi: https://doi.org/10.1016/j.mechmachtheory.2009.09.009.

Rao, S. S. (1983) *Optimization theory and applications*. New York: JOHN WILEY & SONS, INC.

Roppenecker, D. B. *et al.* (2014) 'Modular body of the multi arm snake-like robot', in *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*, pp. 374–379. doi: 10.1109/ROBIO.2014.7090359.

Runge, G. *et al.* (2017) 'A framework for the kinematic modeling of soft material robots combining finite element analysis and piecewise constant curvature kinematics', in *2017 3rd International Conference on Control, Automation and Robotics (ICCAR)*, pp. 7–14. doi: 10.1109/ICCAR.2017.7942652.

Saggere, L. and Kota, S. (1999) 'Synthesis of Planar, Compliant Four-Bar Mechanisms for Compliant-Segment Motion Generation ', *Journal of Mechanical Design*, 123(4), pp. 535–

541. doi: 10.1115/1.1416149.

Sen, S. and Awtar, S. (2013) 'A Closed-Form Nonlinear Model for the Constraint Characteristics of Symmetric Spatial Beams', *Journal of Mechanical Design*, 135(3). doi: 10.1115/1.4023157.

Shuib, S. *et al.* (2007) 'A disposable compliant-forceps for HIV patients', *Journal of Medical Sciences*, 7(4), pp. 591–596.

Sigmund, O. (1997) 'On the Design of Compliant Mechanisms Using Topology Optimization', *Mechanics of Structures and Machines*. Taylor & Francis, 25(4), pp. 493–524. doi: 10.1080/08905459708945415.

Sigmund, O. (2001) 'A 99 line topology optimization code written in matlab', *Structural and Multidisciplinary Optimization*, 21(2), pp. 120–127. doi: 10.1007/s001580050176.

Sopwith, W., Hart, T. and Garner, P. (2002) 'Preventing infection from reusable medical equipment: a systematic review', *BMC Infectious Diseases*, 2(1), p. 4. doi: 10.1186/1471-2334-2-4.

Stoll, J. and Dupont, P. (2006) 'Force control for grasping soft tissue', in *Proc. of IEEE Int. Conf. of Robotics and Automation*, pp. 4309--4311.

Su, H.-J. (2009) 'A Pseudorigid-Body 3R Model for Determining Large Deflection of Cantilever Beams Subject to Tip Loads', *Journal of Mechanisms and Robotics*, 1(2). doi: 10.1115/1.3046148.

Sun, Y., Liu, Y. and Lueth, T. C. (2019) 'FE-Analysis of Bio-inspired Compliant Mechanisms in Matlab for Medical Applications', in *2019 IEEE International Conference on Cyborg and Bionic Systems*.

Talischi, C. *et al.* (2012) 'PolyTop : a Matlab implementation of a general topology optimization framework using unstructured polygonal finite element meshes', *Structural and Multidisciplinary Optimization*, 45(3), pp. 329–357. doi: 10.1007/s00158-011-0696-x.

Trobec, R. and Gersak, B. (1997) 'Direct measurement of clamping forces in cardiovascular surgery', *Medical and Biological Engineering and Computing*, 35(1), pp. 17–20. doi: 10.1007/BF02510386.

Turner, M. J. *et al.* (1956) 'Stiffness and Deflection Analysis of Complex Structures', *Journal of the Aeronautical Sciences*, 23(9), pp. 805–823. doi: 10.2514/8.3664.

Vehar, C. M. and Kota, S. (2006) 'Generalized Synthesis of Nonlinear Springs for Prescribed Load-Displacement Functions', pp. 221–230. doi: 10.1115/DETC2006-99657.

Venkiteswaran, V. K. and Su, H.-J. (2015) 'A parameter optimization framework for determining the pseudo-rigid-body model of cantilever-beams', *Precision Engineering*, 40, pp. 46–54. doi: https://doi.org/10.1016/j.precisioneng.2014.10.002.

Wenger, R. (2013) *Isosurfaces: Geometry, Topology, and Algorithms*. CRC press.

Xu, D. and Ananthasuresh, G. K. (2003) 'Freeform Skeletal Shape Optimization of Compliant Mechanisms ', *Journal of Mechanical Design*, 125(2), pp. 253–261. doi: 10.1115/1.1563634.

Zhang, H. *et al.* (2019) 'Topology Optimized Multimaterial Soft Fingers for Applications on Grippers, Rehabilitation, and Artificial Hands', *IEEE/ASME Transactions on Mechatronics*, 24(1), pp. 120–131. doi: 10.1109/TMECH.2018.2874067.

Zheng, Y. *et al.* (2016) 'Topology optimization of a fully compliant prosthetic finger: Design and testing', in *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pp. 1029–1034. doi: 10.1109/BIOROB.2016.7523766.

Zhou, H. and Ting, K.-L. (2005) 'Shape and Size Synthesis of Compliant Mechanisms Using Wide Curve Theory', *Journal of Mechanical Design*, 128(3), pp. 551–558. doi: 10.1115/1.2180809.

Zienkiewics, O. C. and Taylor, R. L. (2000) *The finite element method. Volume 2: Solid Mechanics*. 5th edn. Butterworth-Heinemann.

Zuo, Z. H. and Xie, Y. M. (2015) 'A simple and compact Python code for complex 3D topology optimization', *Advances in Engineering Software*, 85, pp. 1–11. doi: https://doi.org/10.1016/j.advengsoft.2015.02.006.