Flow-Driven Soft Robots: Theoretical, Computational and Experimental Investigations

Hossein HashemiDehaghi

A Thesis

in

The Department

of

Mechanical, Industrial and Aerospace Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of Master of Applied Science (Mechanical Engineering)

Concordia University

Montréal, Québec, Canada

December 2025

© Hossein HashemiDehaghi, 2025

Concordia University

School of Graduate Studies

This is to certify that the thesis prepared

By: **Hossein HashemiDehaghi**

Entitled: **Flow-Driven Soft Robots: Theoretical, Computational and Experimental Investigations**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Mechanical Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____ Dr. Behrooz Yousefzadeh, Chair and Examiner

_____ Dr. Lyes Kadem, Internal Examiner

_____ Dr. Mojtaba Kheiri, Supervisor

_____ Dr. Amir Hooshiar, Co-supervisor

Approved by _____

Dr. Muthukumaran Packirisamy, Chair

Department of Mechanical, Industrial and Aerospace Engineering

_____ 2025 _____

Dr. Mourad Debbabi, Dean

Gina Cody School of Engineering and Computer Science

Abstract

Flow-Driven Soft Robots: Theoretical, Computational and Experimental Investigations

Hossein HashemiDehaghi

This thesis presents a new approach to actuating soft robots using internal fluid flow, aiming to understand both its feasibility and the principles that govern its behavior. While previous research in this area is limited, this work introduces a theoretical model of a soft, slender cylindrical robot driven by incompressible fluid flow through embedded channels. The robot's static deflection is described using a geometrically-exact equation derived from Euler-Bernoulli beam theory, with flow dynamics incorporated through control volume analysis.

The study also includes fluid-structure interaction simulations based on a hyperelastic material model. The Yeoh third-order model, fitted to tensile test data from the silicone material, was employed to replicate the robot's deformation during the loading phase with high accuracy.

To validate the model, prototypes were fabricated using silicone additive manufacturing. A custom test setup was designed to measure planar deflections in response to varying liquid water flow rates. Image processing techniques were used to track the robot's mid-axis shape and quantify tip deflection, which showed strong alignment with theoretical predictions. On average, the experiments resulted in a tip deflection of approximately 58% of the robot's initial length, with average tip position error of 9% comparing to the analytical model.

Experimental results revealed a different behavior of the soft robot during flow velocity decrease, marked by reduced deflection of the tip, that was not captured by the theoretical or simulation models.

Overall, the findings support the potential of flow-driven actuation in soft robotics and emphasize the need to account for material nonlinearities in future modeling efforts.

To my beloved parents

Mohammad Hashemi, Zoleikha Eram

and

my lovely siblings,

Ehsan & Sarina

Acknowledgments

I am deeply grateful to my supervisors, **Dr. Mojtaba Kheiri** and **Dr. Amir Hooshiar**, whose expertise, patience, and unwavering support guided this research from its earliest stages to completion. Their mentorship has been both inspiring and transformative.

My sincere thanks to **Dr. Hang Xu** for his valuable assistance during key moments of this journey.

To my dear friends—**Sina, Mohadeseh, Amar, Antoine, Abishek, Shahrzad, Reza, Mahdi**—and many others whose names cannot all be listed here, thank you for your encouragement, laughter, and companionship.

To my beloved **family**, especially my aunt **Zohreh Eram**, your unconditional love and belief in me have been my anchor. This achievement is as much yours as it is mine.

Above all, I offer my heartfelt gratitude to **God**, whose remembrance brought peace to my heart and strength to my spirit. In every moment of doubt, His unwavering love reminded me that all will be well.

Contents

List of Figures **x**

List of Tables **xv**

1 Introduction **1**

1.1 Background 1

1.2 Research Motivations 7

1.3 Objectives 8

2 Analytical Modeling **10**

2.1 Actuation Principles in Flow-Driven Soft Robots 11

2.2 Control Volume Analysis of the Tip Region 12

2.3 Derivation of the Static Equilibrium Equation 16

2.3.1 Dimensionless Form of the Governing Equation 20

2.4 Numerical Solution Approach 22

2.4.1 Verification of the Numerical Approach 22

2.4.2 Parametric Shape Analysis 24

3 Computational Modeling **27**

3.1 Coupling Investigation for FSI Analysis 28

3.2 Turbulence Model Study 34
3.3 Fluid-Structure Interaction Study for The Soft Robot 38
3.3.1 Robot Design 38
3.3.2 CFD Analysis 38
3.3.3 Structural FEA Analysis 46

4 Experimentation **50**
4.1 Prototype Development 51
4.1.1 Additive Manufacturing 51
4.1.2 Tensile Test on Silicone Specimen Using DIC Method 54
4.1.3 Injection Molding 57
4.2 Test Setup and Methodology 63
4.2.1 Equipment and Components 63
4.2.2 Flow Sensor Validation 64
4.2.3 Image-Based Deformation Tracking 65
4.2.4 Data Acquisition Protocol 66
4.3 Experimental Results 68

5 Results Comparison and Model Evaluation **74**
5.1 Analytical Model Validation 75
5.2 Computational Model Validation 81

6 Conclusion and Future Work **85**
6.1 Conclusion 85
6.2 Future Work 88

References **90**

A Source Codes **98**

A.1 The Flow Sensor Arduino Script 98

A.2 The Image Processing Matlab Script 102

B Simulation Results for Turbulence Model Comparison **117**

List of Figures

1.1 Minimally invasive surgery (MIS) using a soft robotic catheter in an ex-vivo experimental setup. Reproduced from [1]. 3

1.2 Reproduced illustration of a tendon-driven soft robot with its tendon driver modules [2]. 3

1.3 Schematic of a magnetically controlled catheter designed for vascular surgery. Reproduced from [3]. 4

1.4 McKibben fluid-driven artificial muscles used in soft robotics. Images reproduced from [4]. 6

1.5 Pure bending of Pneu-Net fluidic actuator. Reproduced from [5]. 7

2.1 Schematic of a cylindrical soft robot with total length L and diameter D, actuated by fluid flow at bulk velocity U through a single U-shaped internal channel. The channel has a circular cross-section with diameter d, follows the robot's length L, and is positioned at a distance ϵ from the robot's centerline. The parameter R denotes the distance from the center of the upstream and downstream segments of the channel to the channel's mid-axis. 11

2.2 Two-dimensional view of the fluid control volume at the U-bent segment near the soft robot's tip. 13

2.3 Schematic of the flow-driven soft robot subjected to tip follower force components f_x and f_y, as well as a tip moment. The distributed follower load f_N acts perpendicular to the robot's surface. The angle $\theta(s)$ represents the orientation between the tangent line and the x-axis. Gravity acts in the positive x-direction, causing the soft robot to hang from the top. 16
2.4 Differential control volumes of the channel used to calculate the perpendicular load exerted by fluid flow on the structure. The flow rotates around the n-axis, which is normal to the tangential direction τ. 17
2.5 Verification of the numerical solution (solid lines) against Argyris and Symeonidis [6] (markers) for a cantilevered beam under follower loads of 0.05, 0.1, and 0.2 kN/cm. 23
2.6 Verification of the numerical solution (markers) against the analytical solution (solid lines) for the governing equation of the soft robot in the absence of gravity (Equation 2.35), for $B = 0.5, 1.5$, and 3. 24
2.7 Equilibrium shapes under parametric variation of B and Γ. 26
3.1 Schematic of the elastic pipe conveying fluid, adapted from [7]. 28
3.2 Isometric view of the CFD and structural FEA meshes used in the validation study of a fluid-conveying pipe. 30
3.3 Lateral deflection (mm) of the pipe for various flow velocities. 31
3.4 Side view of the elastic pipe centerlines for different flow velocities. Symbols represent experimental data from [7], and lines are the standard one-way FSI results. 32
3.5 Side view of the elastic pipe centerlines for different flow velocities. Symbols represent experimental data from [7], and lines are the modified one-way FSI results. 33

3.6 Velocity magnitude at the mid-plane of the pipe, simulating the experiment by Anwer and So [8]. 36
3.7 Wall pressure contour for the pipe simulated according to the experiment by Anwer and So [8]. 37
3.8 Schematic of the flow-driven soft robot used in this study, illustrating the two parallel U-shaped internal channels. 39
3.9 Visualization of the soft robot with two symmetrically placed channels. The region which is highlighted in red, shows the flow path where water enters and exits the robot through the channel placed the bottom, as defined in the CFD simulation setup. 40
3.10 CFD mesh of the flow channel at both ends 41
3.11 Static pressure and velocity magnitude contours at $U = 1.61\,\mathrm{m/s}$. 43
3.12 Static pressure and velocity magnitude contours at $U = 4.65\,\mathrm{m/s}$. 44
3.13 Symmetric secondary flow jets after the U-bent segment at a flow rate of $U = 4.65\,\mathrm{m/s}$. The flow transitions back to fully developed after traveling a certain distance in the downstream segment. 45
3.14 Finite-element mesh (106,000 unstructured elements) generated with the `Nonlinear Mechanics` physics preference. 46
3.15 Lateral deflection contour of the flow-driven soft robot at $U = 1.61\,\mathrm{m/s}$ using the modified one-way FSI approach. 48
3.16 Lateral deflection contour of the flow-driven soft robot at $U = 4.65\,\mathrm{m/s}$ using the modified one-way FSI approach. 49
4.1 Formlabs 3B+ SLA printer used for rapid prototyping. 52
4.2 Common issues encountered during SLA printing with soft materials. 52
4.3 The 3D-printed robot in its uncured “green” state. 53

4.4 3D-printed flow-driven soft robot fabricated with silicone resin: (a) demonstration of prototype flexibility under gravitational force, (b) cross-sectional view showing two embedded flow channels. 54
4.5 Dimensions of the dog-bone specimen used for tensile testing. 55
4.6 Tensile test setup with DIC system for strain measurement. 57
4.7 Vertical strain distribution on the specimen at F = 35 N. 58
4.8 Engineering and true stress-strain curves obtained from the tensile test. . . . 58
4.9 CAD models of the mold components for silicone casting. 60
4.10 3D-printed mold components used for silicone injection molding. 61
4.11 Final soft robotic component fabricated from silicone elastomer. 61
4.12 Final configuration of the soft robot after component fabrication and bonding 62
4.13 Test setup used to evaluate the deformation behavior of the soft robot. . . . 63
4.14 Checkerboard image for camera calibration. Axes are for image processing only and are in pixels, not system coordinates. 66
4.15 Image processing workflow used to extract the centerline of the robot from video frames. 67
4.16 Superimposed image of robot deformation. 69
4.17 Extracted centerline of the soft robot in various flow velocities at the loading (forward) phase. 71
4.18 Extracted centerline of the soft robot in various flow velocities at the unloading (backward) phase. 72
4.19 Dimensionless tip deflection of the robot as a function of the dimensionless flow velocity. Filled symbols correspond to deflections while increasing the flow velocity, while hollow symbols indicate deflections while decreasing the flow velocity. 73

5.1 Variation of effective bending stiffness (EI_{eff}) with flow velocity. Stiffness values are derived by fitting the theoretical model to the experimentally observed centerline shape of the soft robot. 76

5.2 Comparison of sample centerlines of the soft robot from tests 1 to 9 (markers) with the analytical model (solid lines) using an effective bending stiffness of $EI_{\text{eff}} = 6.912 \times 10^{-4}\ \text{N} \cdot \text{m}^2$. 79

5.3 Selected experimental centerline shapes of the soft robot from tests 1 to 9 (markers), showing strong agreement with the analytical model (solid lines) using an effective stiffness of $EI_{\text{eff}} = 6.912 \times 10^{-4}\ \text{N} \cdot \text{m}^2$. 80

5.4 Comparison of robot centerline profiles obtained from the standard one-way FSI simulations (markers) and the modified theoretical model (solid lines) incorporating only a follower tensile force at the tip. Results are shown for flow rates of $Q = 1.37$, 1.61, 1.91, and 2.00 L/min. The close agreement between the model and simulation highlights the dominant role of the tip follower force in shaping the deformation under standard one-way FSI approach in ANSYS, which is insufficient for modeling the main case of this study. 82

5.5 Comparison between experimentally measured robot deformation (markers) and modified FSI simulation results (solid lines) under identical flow velocities in Figure 5.3. The simulation effectively captures the robot's bending behavior, demonstrating strong agreement with experimental observations. . 83

5.6 Mid-axis distortion in the FEA simulation caused by channel expansion under constant pressure loading. The deformation reflects the influence of internal pressure on the channel geometry. 84

List of Tables

3.1 Comparison of simulated and experimental (see [7]) tip deflection values for the elastic pipe at various flow velocities. 31

3.2 Comparison of the modified FSI and experimental (see [7]) tip deflection values for the elastic pipe at various flow velocities. 33

3.3 Mesh element study for the CFD analysis. 40

4.1 Mechanical properties of Sil-Poxy and casted elastomers compared to 3D-printed Silicone 40A resin. 62

4.2 Validation results of the Omega flow sensor. 65

5.1 Analytical vs. Experimental tip positions. 77

5.2 CFD-derived inlet and outlet pressures for the soft robot at various flow velocities, corresponding to the experimental shapes shown in Figure 5.3. . . . 78

B.1 CFD and experimental results for the **k-ε turbulence model**, separated by Outer and Inner Bend regions. Experimental data reproduced from [8]. . . . 118

B.2 CFD and experimental results for the **k-ω SST turbulence model**, separated by Outer and Inner Bend regions. Experimental data reproduced from [8]. 119

B.3 CFD and experimental results for the **RSM Stress-ω turbulence model**, separated by Outer and Inner Bend regions. Experimental data reproduced from [8]. . . . 120

B.4 CFD and experimental results for the **RSM Stress-BSL** turbulence model, separated by Outer and Inner Bend regions. Experimental data reproduced from [8]. 121
B.5 CFD and experimental results for the **k-ω SST turbulence model** with refined mesh, separated by Outer and Inner Bend regions. Experimental data reproduced from [8]. 122
B.6 CFD and experimental results for the **URANS k-ω SST turbulence model**, separated by Outer and Inner Bend regions. Experimental data reproduced from [8]. 123

Chapter 1

Introduction

1.1 Background

While traditional rigid-link robots continue to evolve and dominate many industrial and precision tasks, a complementary branch of robotics has emerged, soft robotics, focused on systems built from highly compliant, elastically deformable materials such as silicones, elastomers, and soft polymers. Unlike rigid robots, which rely on discrete joints and stiff components, soft robots leverage material compliance to achieve continuous deformation and adaptive shape changes. This fundamental difference results in behavior often described as having effectively higher degrees of freedom: soft bodies can conform to surrounding geometries, absorb impacts without catastrophic failure, and interact safely with fragile environments [9]. A key engineering trade-off exists between adaptability and controllable stiffness or force. Rigid robots typically offer superior positional accuracy, repeatable kinematics, and greater load capacity. In contrast, soft robots sacrifice some of these capabilities in exchange for unmatched conformity, intrinsic safety in contact-rich tasks, and mechanical resilience to unpredictable disturbances. This synergy explains the growing prioritization of

soft designs in applications where safe interaction, access to constrained geometries, or operation in unstructured environments is essential [9]. Soft robotics has rapidly expanded into domains such as inspection, wearable assistance, underwater exploration, search-and-rescue, and industrial process control, fields where passive compliance and robustness to accidental contact are highly valued [10]. Among these, healthcare—particularly minimally invasive surgery (MIS)—has emerged as a major driver of research and clinical translation. In MIS and catheter-based interventions, devices must navigate complex anatomical pathways, minimize localized stress, and operate reliably near delicate, dynamic tissues such as those found in the beating heart. Soft continuum devices and elastomeric tools have been proposed for endoscopy, intravascular interventions, and other intrabody tasks due to their compliance, which reduces the risk of perforation, embolization, and vessel injury while enabling access to remote targets [1, 11–14]. Their miniaturizable cross-sections, novel dexterous motions, and biocompatible materials continue to motivate research toward clinical adaptation. Figure 1.1 illustrates the minimally invasive surgery using a soft robotic catheter.

Soft robots are actuated through various physical principles, each offering distinct trade-offs in power density, integration complexity, and suitability for sensitive environments. Mechanical (tendon/cable-driven) actuation uses remote motors to pull routed cables or tendons, producing bending and shape change. This method benefits from mature motor technologies, high power-conversion efficiency, and the ability to co-locate heavy actuators at the robot's base, allowing for compact distal sections. However, it suffers from nonlinear friction, tendon hysteresis, complex modeling, and transmission vulnerabilities that are critical in precision and surgical contexts [15–18]. Figure 1.2 shows a tendon-driven soft robot along with its tendon drivers.

Electrically driven devices—such as dielectric elastomer actuators, piezoelectric stacks, and electrically active polymers—offer fast responses and compact form factors. Yet, they

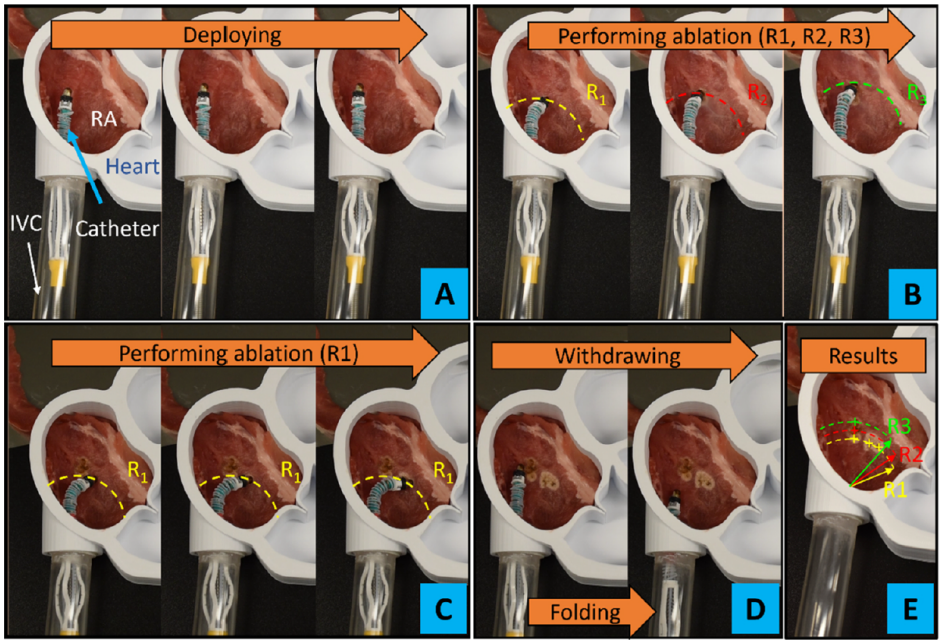


Figure 1.1: Minimally invasive surgery (MIS) using a soft robotic catheter in an ex-vivo experimental setup. Reproduced from [1].

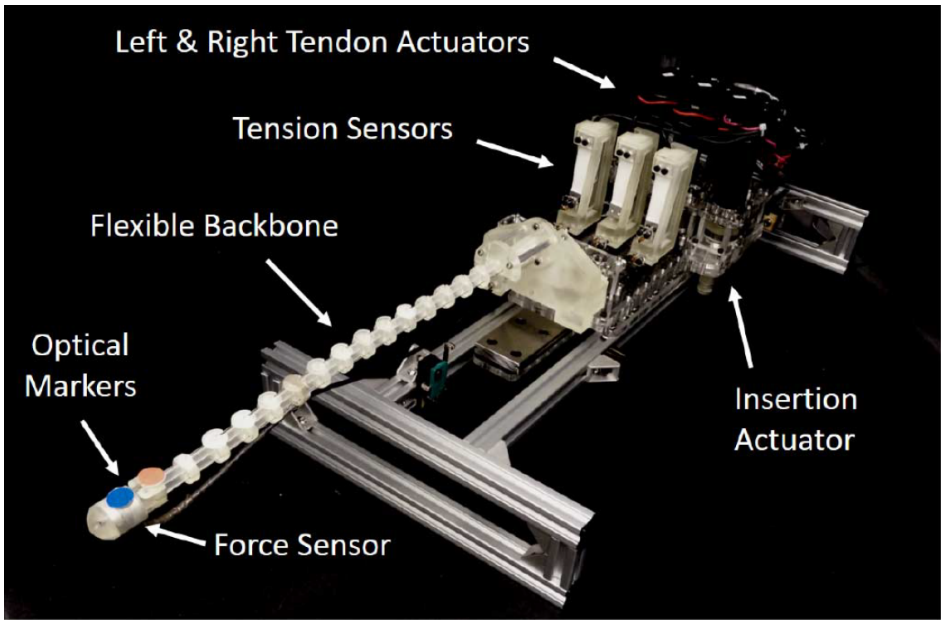


Figure 1.2: Reproduced illustration of a tendon-driven soft robot with its tendon driver modules [2].

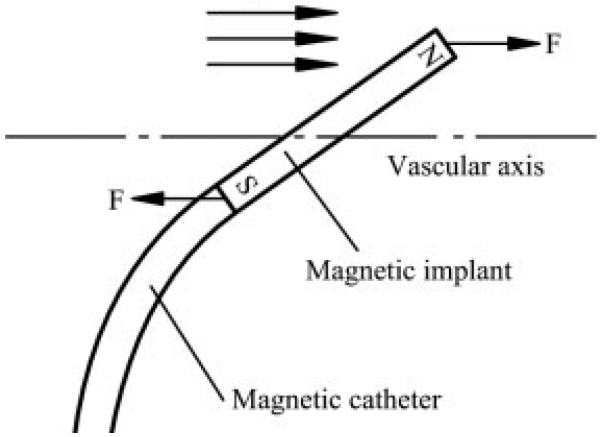


Figure 1.3: Schematic of a magnetically controlled catheter designed for vascular surgery. Reproduced from [3].

often require high voltages, complex material fabrication, and pose integration and safety challenges in biological environments [17, 19, 20]. Magnetic and thermal actuation provide complementary capabilities. Magnetic control (Figure 1.3) enables untethered actuation for microscale robots and supports field-responsive materials (e.g., magnetorheological elastomers) for adjustable stiffness. However, it typically requires external field generators and is limited by field penetration and local power density in larger systems [17, 19, 21]. Thermal approaches, including shape memory alloys and phase-change materials, can produce large strains or discrete stiffness changes but are generally energy-intensive and slow due to heating and cooling cycles, making them less suitable for rapid or continuous modulation [19, 22].

Fluidic actuation is one of the most widely adopted mechanisms in soft robotics due to its natural compatibility with deformable chambers and straightforward fabrication techniques [17, 19]. By pressurizing internal cavities, these actuators can produce bending, extension, torsion, and complex multimodal motion with relatively simple geometries. Pneumatics benefit from the ubiquity of air and enable lightweight actuators, while hydraulics offer higher power density and more efficient force transmission for load-intensive tasks. Importantly, fluidic systems keep electrical components out of the workspace, reduce thermal

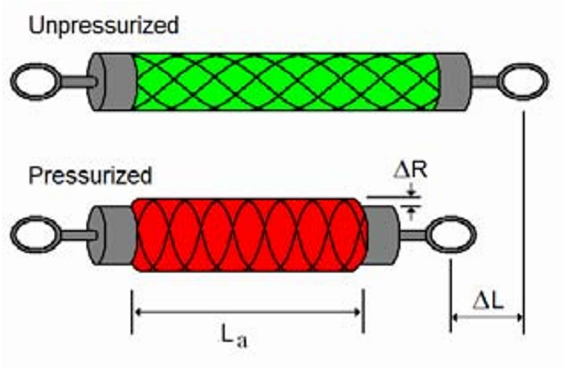
risks, and allow the use of biocompatible working fluids—critical for medical and hazardous-environment applications [1, 23, 24].

A wide range of soft robots utilize fluidic actuation. Pneumatic artificial muscles (PAMs), such as McKibben actuators (Figure 1.4), were among the earliest examples [25]. More recent designs include fluidic elastomer actuators (FEAs), which deform by pressurizing embedded channels in elastomeric materials (Figure 1.5). These are often implemented using Pneu-Net architectures, enabling complex motions with minimal energy once deformed [26, 27]. Examples include quadrupedal soft robots capable of adaptive locomotion [28], camouflage-enabled robots with tunable surface properties [29], and snake-inspired robots using bidirectional FEAs for serpentine movement [30]. The design space has expanded with fiber-reinforced, fluid-filled elastomeric enclosures that tune anisotropy and force density [31], programmable fluidic networks enabling origami-style sequential self-folding [32], and hydraulic filament artificial muscles offering scalable, robust output for flexible systems [33]. Recirculating air flow architectures also demonstrate embodied control [34]. Fundamental models of compressible air flow in soft structures with slowly varying geometry clarify how pressure–structure coupling governs the deformation [35].

Fluidic actuation has also enabled soft manipulators and grippers. The OctArm continuum manipulator demonstrated pneumatic actuation in both air and underwater environments [36], while PneuFlex-based hands achieved dexterous grasping [37]. Other designs explored torsional motion using helically arranged tubes [38], and hybrid robots combining locomotion and grasping [39]. In aquatic applications, manta ray-inspired swimming robots used bending pneumatic actuators [40], and hydraulic soft robotic fish achieved 3D swimming capabilities [41]. A "pneudraulic" scheme that places pneumatic and hydraulic circuits in series [42]. Jung et al. review emerging pump-less fluidic actuators driven by liquid–gas phase changes [43]. Fluidic actuation is particularly attractive for MIS due to



(a) Image of Mckibben muscle.



(b) Operating principle of Mckibben muscle.

Figure 1.4: McKibben fluid-driven artificial muscles used in soft robotics. Images reproduced from [4].

its support for compliant interaction and passive shape adaptation. Elastomeric chambers distribute contact forces, reducing tissue stress, and chamber-based designs can be miniaturized to navigate unstructured anatomical paths [11, 12, 44]. Hydraulic systems enable higher force output when needed, while pneumatic designs simplify exhaust and reduce distal mass—features that can be tailored to specific tasks and environmental constraints. Clinically oriented systems such as STIFF-FLOP and modular intra-organ robots demonstrate dexterous navigation, shape control, and task execution in constrained anatomy for MIS [45–47]. Earlier analyses have also contributed to understanding how internal fluid flow interacts with deformable structures; for example, Marzban et al. conducted a parametric study on the fluid–structure behavior of 3D suspended polymeric microfluidic channels for bioelement detection [48]. Related theoretical frameworks capture stiffness and continuum dynamics in compliant, fluid-driven structures: Lie-theoretic stiffness modelling extends geometric reasoning for soft robots [49], while Cosserat-rod formulations enable real-time simulation and control of continuum manipulators [50]. In parallel, beam models with embedded parallel-channel networks quantify fluid–structure coupling [51].

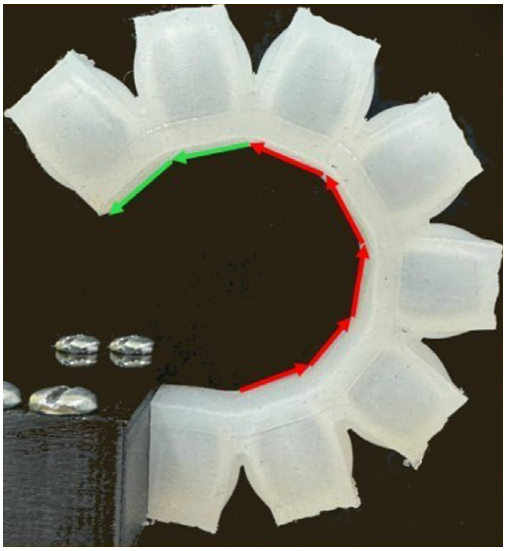


Figure 1.5: Pure bending of Pneu-Net fluidic actuator. Reproduced from [5].

Despite these advantages, traditional fluidic implementations face practical limitations that hinder intrabody deployment and precise control. Most systems rely on external reservoirs, valves, and tubing, which increase bulk and complicate clinical integration [1, 19]. Pneumatic systems suffer from compressibility, introducing nonlinear dynamics, reduced bandwidth, hysteresis, and acoustic noise. Hydraulic systems, while incompressible, require careful sealing and fluid management to prevent contamination [19]. Control remains challenging: distributed compliance leads to high-dimensional, nonlinear dynamics that complicate accurate, repeatable positioning and modeling.

1.2 Research Motivations

Several specific limitations in current fluidic actuation systems motivate the exploration of continuous, incompressible flow-driven soft actuators for clinical and other safety-critical applications. Conventional chamber-based fluidic systems typically rely on discrete pressurization events (e.g., inflation and deflation), which generate transient pressure spikes and

limit fine-grained control. Pneumatic systems are further constrained by the compressibility of air, which undermines repeatable positioning—an undesirable trait in surgical contexts.

A flow-driven approach, where steady or precisely modulated fluid flow through channels or tailored geometries governs deformation, offers a promising alternative. Continuous flow actuation can produce smoother motion, enable superior control over deformation without abrupt pressure transients, and facilitate the integration of stabilization strategies. Furthermore, co-designing channel geometry, working fluid selection, and flow regulation allows for balancing compliance and force output, potentially combining the high force generation of hydraulics with the fine control typically absent in pneumatic systems. Given the limited direct attention flow-driven actuation has received in the literature, investigating its principles and practical implementation is both timely and necessary. This research aims to uncover new strategies to enhance controllability, reduce peripheral complexity, and expand the range of safe, biocompatible soft robotic devices suitable for minimally invasive interventions and other demanding environments.

1.3 Objectives

The objective of this research is to conceptualize and establish the modeling and analysis tools for flow-driven soft robots. To achieve this, the study pursued four specific aims:

1. **Conceptual and Analytical Development**

   Formulate the concept of a flow-driven soft robot and derive the mathematical relationships governing its static deformation. These derivations support parametric studies and identify key inputs relevant for future control strategies.

2. **Prototyping and Fabrication**

Design and manufacture soft robotic prototypes based on the flow-driven concept, investigating suitable fabrication methods and materials to ensure functional compliance and robustness.

3. **Experimental Validation**

   Conduct controlled experiments to verify the deformation behavior of the prototypes, thereby validating the analytical models and providing empirical benchmarks for refinement.

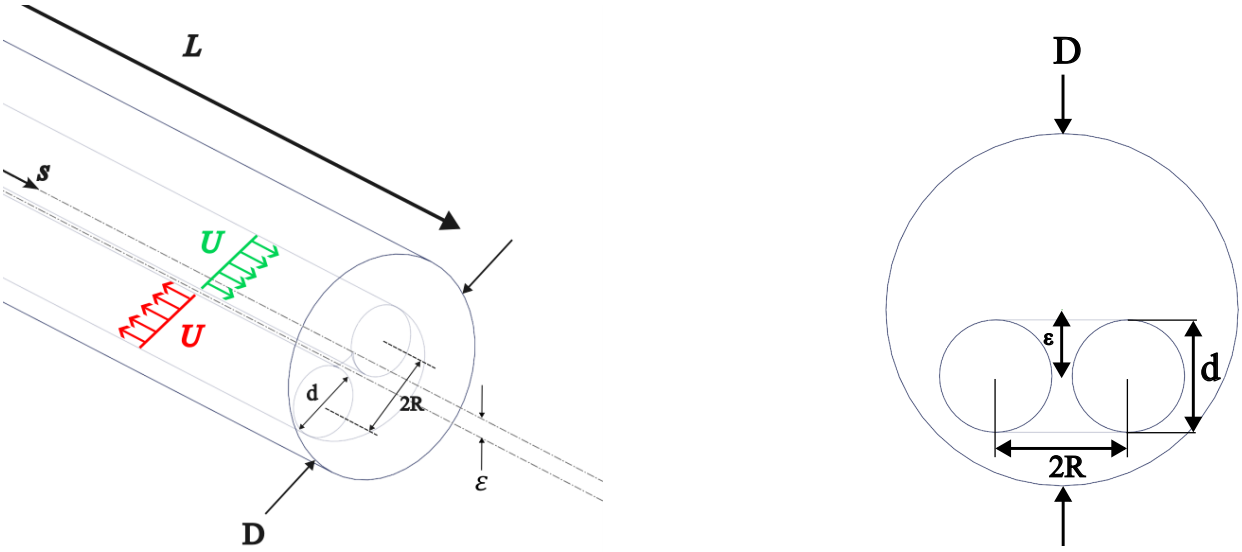
4. **Computational Simulation**

   Perform full three-dimensional fluid–structure interaction simulations to capture the coupled behavior of the robot under flow actuation. These simulations assess performance and establish a foundation for future optimization studies.

Chapter 2

Analytical Modeling

This chapter establishes an analytical framework necessary to understand the fundamental mechanisms driving the flow-driven soft robot concept introduced in this work. Beginning with a simplified model, a cylindrical soft robot featuring an internal U-shaped channel, the analysis focuses on the fluid-structure interactions at the robot's tip. A control volume approach is employed to extract the tip forces and moment generated by the internal fluid flow. These effects are then incorporated into a structural model using Euler–Bernoulli beam theory, resulting in a nonlinear equation governing static deflection in two dimensions. The model accounts for tip follower forces, tip moment, and distributed pressure along the robot's body, effectively replacing the physical U-shaped tip with its mechanical influence. To facilitate analysis, the governing equation is nondimensionalized, and a numerical solution is implemented in MATLAB. The code is first verified against a benchmark case involving a cantilevered beam subjected to distributed loading at three different magnitudes. This analytical foundation enables parametric studies of the robot's deformation behavior under various flow conditions.



(a) Isometric view.

(b) Side view.

Figure 2.1: Schematic of a cylindrical soft robot with total length L and diameter D, actuated by fluid flow at bulk velocity U through a single U-shaped internal channel. The channel has a circular cross-section with diameter d, follows the robot's length L, and is positioned at a distance ϵ from the robot's centerline. The parameter R denotes the distance from the center of the upstream and downstream segments of the channel to the channel's mid-axis.

2.1 Actuation Principles in Flow-Driven Soft Robots

The actuation in a flow-driven soft robot is fundamentally governed due to the fluid pressure and friction exerted on the internal channel walls. These channels are embedded within the body of the robot, allowing fluid to circulate and generate the desired mechanical response.

For simplicity, the robot can be modeled as a cantilevered cylindrical structure, characterized by length L and diameter D. To ensure controlled fluid dynamics and prevent unintended discharge, especially critical in medical applications, the channels must be designed such that both the inlet and outlet are located at the fixed end of the robot. This closed-loop configuration ensures that no fluid escapes into the operating environment, such as the human body. One basic shape of the channel is a U-shaped design comprised of two straight segments and a U-bent segment, as shown in Figure 2.1. The fluid enters through

the inlet with bulk velocity U, and flows through the upstream straight segment of length L, then it turns through the U-bent of radius R and continues through the downstream straight segment which has the same length L and is parallel to the upstream channel. All sections have diameter d and are placed in distance ϵ from the centerline of the soft robot.

2.2 Control Volume Analysis of the Tip Region

To analyze the static deformation of the robot, we model it as a slender structural body, excluding the tip region where the U-bent segment is located. The mechanical influence of the tip is incorporated into the rest of the body as equivalent reaction forces and moments.

Figure 2.2 illustrates the control volume of an incompressible fluid (e.g., water) flowing through U-bent segment at the tip. The robot is assumed to be in a deformed configuration that is already in static equilibrium, with the tip oriented at an angle relative to the horizontal axis.

A curvilinear coordinate system $(\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{n}}, \hat{\boldsymbol{b}})$ is adopted for the channel with the origin located symmetrically with respect to the two straight segments on the plane of the flow channel. In this system:

- $\hat{\boldsymbol{\tau}}$ is tangent to the centerline of the channel.
- $\hat{\boldsymbol{n}}$ is oriented perpendicular to the channel plane, pointing toward the centerline of the robot.
- $\hat{\boldsymbol{b}}$ completes the right-handed coordinate system.

The flow is assumed to be steady state, incompressible, fully turbulent. The flow velocity is assumed to be uniform (i.e., plug flow assumption), which is consistent with the assumption of turbulent flow. Using a control volume analysis to apply the conservation of the linear

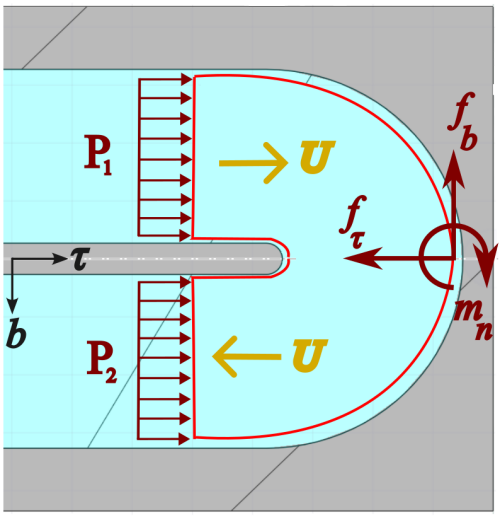


Figure 2.2: Two-dimensional view of the fluid control volume at the U-bent segment near the soft robot's tip.

and angular momentums principles allows us to determine the net force and moment exerted by the fluid on the tip and in turn on the robot.

The resultant force $\mathbf{f} = f_\tau \hat{\boldsymbol{\tau}} + f_n \hat{\mathbf{n}} + f_b \hat{\mathbf{b}}$ includes contributions from shear and normal components arising from interactions between the flow and the channel walls.

To simplify the analysis, we assume a constant cross-sectional area throughout the channel. Under these assumptions, the equations for the conservation of the linear momentum can be expressed as:

For the $\hat{\boldsymbol{\tau}}$-direction:

$$\begin{aligned} \Sigma F_\tau &= P_1 A + P_2 A - f_\tau \\ &= \int_{\text{C.S.}} V_\tau \, \rho \, (\mathbf{V} \cdot \mathbf{n}_{cs}) \, dA = -\rho A U^2 - \rho A U^2 \qquad (2.1) \\ \Rightarrow \quad f_\tau &= (P_1 + P_2) A + 2\rho A U^2, \end{aligned}$$

where P_1 and P_2 are the static pressures at the upstream and downstream of the U-bent segment at the tip, respectively, $A = \frac{\pi}{4}d^2$ is the channel cross-section area and ρ is the fluid density.

Equation 2.1 represents a follower-type tensile force acting at the tip of the robot. The first term accounts for the pressure contributions at the inlet and outlet of the control volume, while the second term captures the fluid momentum change.

For the $\hat{\boldsymbol{b}}$-direction:

$$\begin{aligned} \Sigma F_b &= -f_b \\ &= \int_{\text{C.S.}} V_b \, \rho \, (\mathbf{V} \cdot \mathbf{n}_{cs}) \, dA \end{aligned} \tag{2.2}$$

Assuming $V_b \approx 0$, the integral term vanishes, leading to:

$$\Rightarrow f_b = 0 \tag{2.3}$$

Similarly, for the $\hat{\boldsymbol{n}}$-direction, under the assumption $V_n \approx 0$, we obtain:

$$\Rightarrow f_n = 0 \tag{2.4}$$

At the same time, the upstream and downstream pressures surrounding the U-bent segment can be expressed as:

$$P_1 = P_{\text{in}} - P_{\text{loss, upstream}} \tag{2.5}$$

$$P_2 = P_{\text{out}} + P_{\text{loss, downstream}}, \tag{2.6}$$

Where P_{in} and P_{out} are the static pressure at the inlet and outlet of the flow channel, respectively. The terms $P_{\text{loss, upstream}}$ and $P_{\text{loss, downstream}}$ represent the pressure losses in the

upstream and downstream segments, respectively.

Assuming $P_{\text{loss, upstream}} = P_{\text{loss, downstream}}$, and substituting Equations 2.5 and 2.6 into Equation 2.1, we obtain the following expression:

$$\Rightarrow f_\tau = (P_{in} + P_{out})\, A + 2\,\rho\, A\, U^2. \tag{2.7}$$

We now consider the angular momentum balance to evaluate the moment exerted by the fluid, using control volume analysis with rotation about the $\hat{\boldsymbol{n}}$-direction.

$$\begin{aligned} \Sigma M_n &= -P_1 A\, R + P_2 A\, R - m_n \\ &= \int_{\text{C.S.}} (\mathbf{r} \times \mathbf{V})\, \rho\, (\mathbf{V} \cdot \mathbf{n}_{cs})\, dA = \rho\, A\, R\, U^2 - \rho\, A\, R\, U^2 = 0 \\ \Rightarrow m_n &= (P_2 - P_1) A\, R. \end{aligned} \tag{2.8}$$

where R denotes to the radial distance from the upstream and downstream sections to the centerline of the U-shaped channel (see Figure 2.1). Knowing that:

$$P_1 = P_2 + P_{\text{loss, U-turn}} \tag{2.9}$$

which we can use to convert Equation 2.8 into:

$$\Rightarrow m_n = -P_{\text{loss, U-turn}} A\, R \tag{2.10}$$

where $P_{\text{loss, U-turn}}$ is the pressure loss inside the U-bent segment.

As illustrated in Figure 2.1, introducing a tensile force can be advantageous for generating a moment about the $\hat{\boldsymbol{b}}$-direction, particularly when the channel is carved with an offset from

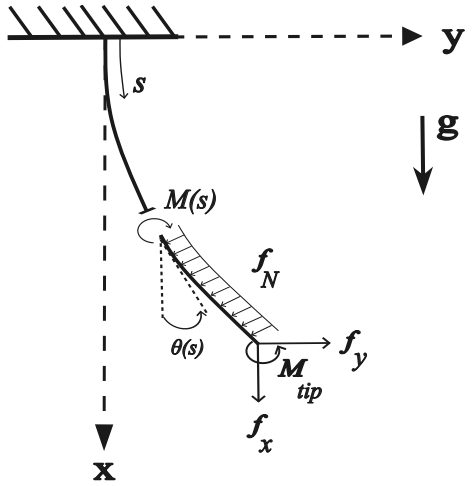


Figure 2.3: Schematic of the flow-driven soft robot subjected to tip follower force components f_x and f_y, as well as a tip moment. The distributed follower load f_N acts perpendicular to the robot's surface. The angle $\theta(s)$ represents the orientation between the tangent line and the x-axis. Gravity acts in the positive x-direction, causing the soft robot to hang from the top.

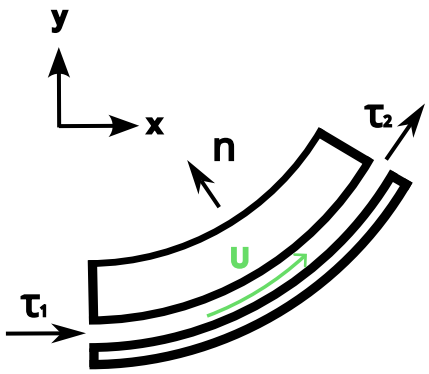
the robot's centerline, represented by the eccentricity $\boldsymbol{\varepsilon}$:

$$\begin{aligned} \boldsymbol{M_{tip}} &= \boldsymbol{\varepsilon} \times f_\tau \, \hat{\boldsymbol{\tau}} \\ &= -\varepsilon \, \hat{\boldsymbol{n}} \times f_\tau \, \hat{\boldsymbol{\tau}} \end{aligned} \tag{2.11}$$

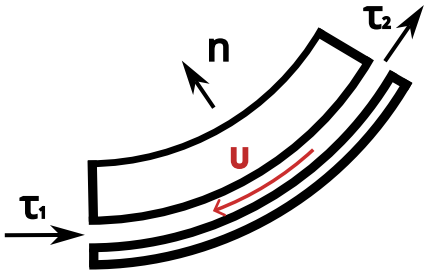
2.3 Derivation of the Static Equilibrium Equation

The soft robot is modeled as a slender cylindrical beam of constant length L, cross-sectional area A_r, and flexural rigidity EI. A schematic of the robot in the xy-plane is shown in Figure 2.3. The robot is modeled as a cantilevered beam, fixed at one end and free at the other.

At the free end, the robot experiences a resultant tensile follower force f_τ and an external moment M_{tip}, as described in Section 2.2. The force is decomposed into components along the x- and y-directions:



(a) control volume representing the upstream segment of the channel



(b) control volume representing the downstream segment

Figure 2.4: Differential control volumes of the channel used to calculate the perpendicular load exerted by fluid flow on the structure. The flow rotates around the n-axis, which is normal to the tangential direction τ.

$$f_x = f_\tau \cos\theta(L), \quad f_y = f_\tau \sin\theta(L), \tag{2.12}$$

no indent where $\theta(s)$ denote the angle between the centerline of the robot and the x-axis at arc length position s.

A distributed follower force, represented by $\mathbf{f}_N$, acts normal to the centerline of the robot, representing the net pressure force, along its length. This force is derived from the linear momentum balance of a differential element (Figure 2.4) in the upstream and downstream segments.

For the upstream segment,

$$-R_u + P_u A\hat{\boldsymbol{\tau}}_1 - \left(P_u + \frac{dP}{ds}\bigg|_u \delta s\right) A\hat{\boldsymbol{\tau}}_2 = \rho A U^2(\hat{\boldsymbol{\tau}}_2 - \hat{\boldsymbol{\tau}}_1), \tag{2.13}$$

and for the downstream segment,

$$-R_d + P_d A\hat{\boldsymbol{\tau}}_1 - \left(P_d + \left.\frac{dP}{ds}\right|_d \delta s\right) A\hat{\boldsymbol{\tau}}_2 = \rho A U^2(\hat{\boldsymbol{\tau}}_2 - \hat{\boldsymbol{\tau}}_1), \tag{2.14}$$

Adding Equations 2.13 and 2.14:

$$-R_u - R_d + (P_u + P_d)A(\hat{\boldsymbol{\tau}}_1 - \hat{\boldsymbol{\tau}}_2) = 2\rho A U^2(\hat{\boldsymbol{\tau}}_2 - \hat{\boldsymbol{\tau}}_1), \tag{2.15}$$

where the assumption of $(dP/ds)|_d = -(dP/ds)|_u$ has been implemented.

Assuming a small angular increment $d\theta$ such that $\hat{\boldsymbol{\tau}}_2 = \hat{\boldsymbol{\tau}}_1 + d\theta\,\hat{\boldsymbol{n}}$, the distributed follower load is obtained:

$$\mathbf{f}_N = \frac{R_u}{ds} + \frac{R_d}{ds} = -\left((P_u + P_d)A + 2\rho A U^2\right)\frac{d\theta}{ds}\hat{\boldsymbol{n}}, \tag{2.16}$$

Gravity acts in the $+x$ direction and is included in the moment balance. Considering a differential segment at distance s from the fixed end, the moment balance is (see. Figure 2.3):

$$\begin{aligned} &-M(s) - f_x[y(L) - y(s)] + f_y[x(L) - x(s)] - \int_s^L |\mathbf{f}_N| \sin\theta(\sigma)[y(\sigma) - y(s)]d\sigma \\ &- \int_s^L |\mathbf{f}_N| \cos\theta(\sigma)[x(\sigma) - x(s)]d\sigma - \int_s^L (2\rho A + \rho_r A_r) g[y(\sigma) - y(s)]d\sigma + M_{\text{tip}} = 0, \end{aligned} \tag{2.17}$$

where A_r and ρ_r denote the cross-sectional area and material density of the robot structure, respectively.

Differentiating Equation 2.17 with respect to s, and applying the Leibniz rule with $dx/ds = \cos\theta(s)$, $dy/ds = \sin\theta(s)$ yields:

$$\begin{aligned} &-\frac{dM(s)}{ds} + f_x \sin\theta(s) - f_y \cos\theta(s) + \sin\theta(s) \int_s^L |\mathbf{f}_N| \sin\theta(\sigma)d\sigma \\ &+ \cos\theta(s) \int_s^L |\mathbf{f}_N| \cos\theta(\sigma)d\sigma + (2\rho A + \rho_r A_r) g(L - s) \sin\theta(s) = 0. \end{aligned} \tag{2.18}$$

Substituting Equations 2.12 and 2.16 into Equation 2.18 gives,

$$\begin{aligned}
&-\frac{dM(s)}{ds} + f_\tau \sin\theta(s)\cos\theta(L) - f_\tau \cos\theta(s)\sin\theta(L) \\
&+\sin\theta(s)\int_s^L \left((P_u + P_d)A + 2\rho A U^2\right)\frac{d\theta}{ds}\sin\theta(\sigma)d\sigma \\
&+\cos\theta(s)\int_s^L \left((P_u + P_d)A + 2\rho A U^2\right)\frac{d\theta}{ds}\cos\theta(\sigma)d\sigma \\
&\qquad +(2\rho A + \rho_r A_r)g(L-s)\sin\theta(s) = 0.
\end{aligned} \tag{2.19}$$

Using Euler-Bernoulli's beam theory:

$$\frac{dM(s)}{ds} = \frac{d}{ds}\left(EI\frac{d\theta(s)}{ds}\right) = EI\frac{d^2\theta(s)}{ds^2}. \tag{2.20}$$

Assuming that the pressure loss between the inlet and a point located at the distance s from the fixed end on the usptream segment is the same as the pressure loss between a point located at the same distance s on the downstream segment and the outlet, we can write $P_u + P_d = P_{\text{in}} + P_{\text{out}}$, which then simplifies Equation 2.19 to:

$$\begin{aligned}
&-EI\frac{d^2\theta(s)}{ds^2} + \left((P_{\text{in}} + P_{\text{out}})A + 2\rho A U^2 - f_\tau\right)\sin(\theta(s) - \theta(L)) \\
&\qquad +(2\rho A + \rho_r A_r)g(L-s)\sin\theta(s) = 0.
\end{aligned} \tag{2.21}$$

In the final form, the effect of the tensile follower force is nullified by the net pressure load normal to the centerline of the robot:

$$EI\frac{d^2\theta(s)}{ds^2} - (2\rho A + \rho_r A_r)g(L-s)\sin\theta(s) = 0. \tag{2.22}$$

The boundary conditions are:

$$\theta(0) = 0, \quad EI\frac{d\theta(s)}{ds}\bigg|_{s=L} = \varepsilon\, f_\tau \tag{2.23}$$

The following equations are used to reconstruct the deformed shape of the robot in the xy-plane:

$$x(s) = \int_0^s \cos\theta(\xi)\, d\xi, \quad y(s) = \int_0^s \sin\theta(\xi)\, d\xi. \tag{2.24}$$

2.3.1 Dimensionless Form of the Governing Equation

To generalize the static equilibrium model and reduce the number of physical parameters, we cast the governing equation in the dimensionless form. This approach not only simplifies the mathematical form but also facilitates comparison across different robot geometries and material properties.

We begin by introducing the dimensionless arc-length parameter:

$$\eta = \frac{s}{L}, \qquad \eta \in [0, 1] \tag{2.25}$$

which maps the physical domain $s \in [0, L]$ to a normalized interval. Using the chain rule, the spatial derivatives transform as:

$$\frac{d}{ds} = \frac{1}{L}\frac{d}{d\eta}, \qquad \frac{d^2}{ds^2} = \frac{1}{L^2}\frac{d^2}{d\eta^2} \tag{2.26}$$

Substituting these expressions into the dimensional Equation 2.22 gives:

$$EI \cdot \frac{1}{L^2}\frac{d^2\theta}{d\eta^2} - (2\rho A + \rho_r A_r)\, gL(1-\eta)\sin\theta = 0 \tag{2.27}$$

Dividing through by EI/L^2, the dimensionless form becomes:

$$\frac{d^2\theta}{d\eta^2} - \Gamma(1-\eta)\sin\theta = 0, \qquad \eta \in [0, 1] \tag{2.28}$$

where the dimensionless load parameter Γ is defined as:

$$\Gamma = \frac{(2\rho A + \rho_r A_r) g L^3}{EI} \tag{2.29}$$

This parameter encapsulates the relative importance of gravitational loading and the bending stiffness.

The boundary conditions are similarly transformed. At the clamped end, the slope remains zero:

$$\theta(0) = 0 \tag{2.30}$$

At the free end, the bending moment boundary condition is expressed as:

$$EI \cdot \frac{1}{L} \frac{d\theta}{d\eta}\Big|_{\eta=1} = f_\tau \varepsilon \tag{2.31}$$

which leads to the dimensionless form:

$$\frac{d\theta}{d\eta}(1) = B \tag{2.32}$$

with the dimensionless tip-moment parameter:

$$B = \frac{L f_\tau \varepsilon}{EI} \tag{2.33}$$

leading to:

$$B = \frac{L \varepsilon A \left[(P_{\text{in}} + P_{\text{out}}) + 2\rho U^2 \right]}{EI} \tag{2.34}$$

The parameter B represents the ratio between the actuation load and the restraining load (bending stiffness). Increasing B is expected to increase deformation. Several factors can be used to vary B, including geometric parameters (L, A, ε and I), material properties

of the robot (E), and fluid properties or operating conditions (P, ρ, and U). For example, using a softer material (i.e., reducing E) lowers the structural stiffness, thereby increasing B and producing larger deformations. Similarly, increasing the channel cross-sectional area A reduces the second moment of area I; both these changes contribute to an increase in B. It is also seen that both static and dynamic pressures influence the actuation load, and a denser fluid like water could create larger actuation loads compared to, for example, air.

2.4 Numerical Solution Approach

2.4.1 Verification of the Numerical Approach

Before proceeding with the full solution of the governing equations, the numerical implementation was verified against a classical benchmark problem. Specifically, a cantilevered beam subjected to a uniformly distributed normal follower load along its neutral axis was considered. This problem was previously solved by Argyris and Symeonidis [6] using a nonlinear finite element model, and their results serve as a reference for verification.

The verification was performed using a MATLAB implementation of the boundary value problem solver `bvp5c`, which is well-suited for solving nonlinear differential equations with boundary conditions. The beam properties used in the verification study were: length $L = 100\,\text{cm}$, Young's modulus $E = 2.1 \times 10^7\,\text{N/cm}^2$, and moment of inertia $I = 1.666667\,\text{cm}^4$.

To assess the accuracy of the numerical solution, the deformation profiles obtained from the MATLAB code were compared against the reference solution for three different load magnitudes: $P_L = 0.05\,\text{kN/cm}$, $0.1\,\text{kN/cm}$, and $0.2\,\text{kN/cm}$. Figure 2.5 illustrates the deformation curves for each load case alongside the reference solution. The comparison shows excellent agreement across all cases, confirming the correctness and reliability of the numerical implementation.

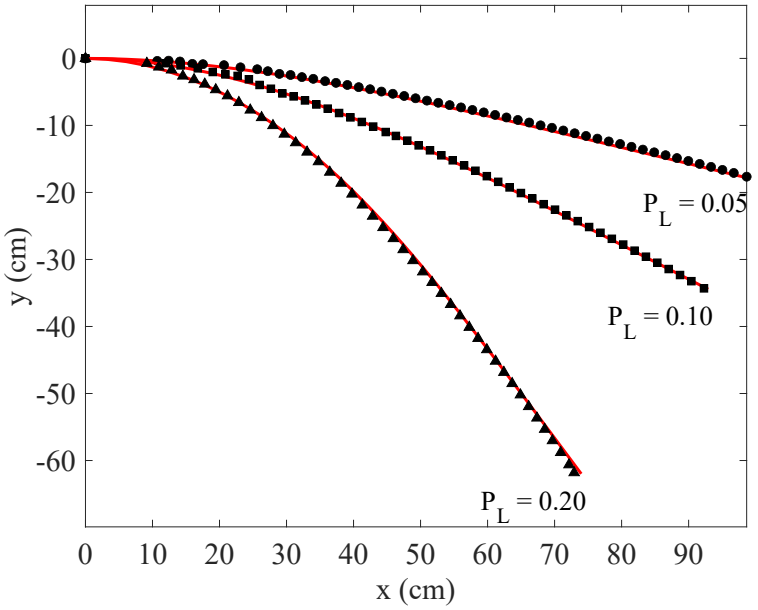


Figure 2.5: Verification of the numerical solution (solid lines) against Argyris and Symeonidis [6] (markers) for a cantilevered beam under follower loads of 0.05, 0.1, and 0.2 kN/cm.

Another form of verification can be performed on the simplified version of Equation 2.28 by neglecting the gravity term, which results in:

$$\frac{d^2\theta}{d\eta^2} = 0. \tag{2.35}$$

Applying the boundary conditions, the solution for the dimensionless slope becomes:

$$\theta(\eta) = B\eta, \tag{2.36}$$

To reconstruct the shape from this closed-form solution using Equations 2.24, we obtain:

$$\frac{x}{L} = \int_0^\eta \cos\big(\theta(\xi^*)\big)\, d\xi^* = \frac{1}{B}\sin(B\eta), \tag{2.37}$$

$$\frac{y}{L} = \int_0^\eta \sin\big(\theta(\xi^*)\big)\, d\xi^* = \frac{1}{B}\big(1-\cos(B\eta)\big). \tag{2.38}$$

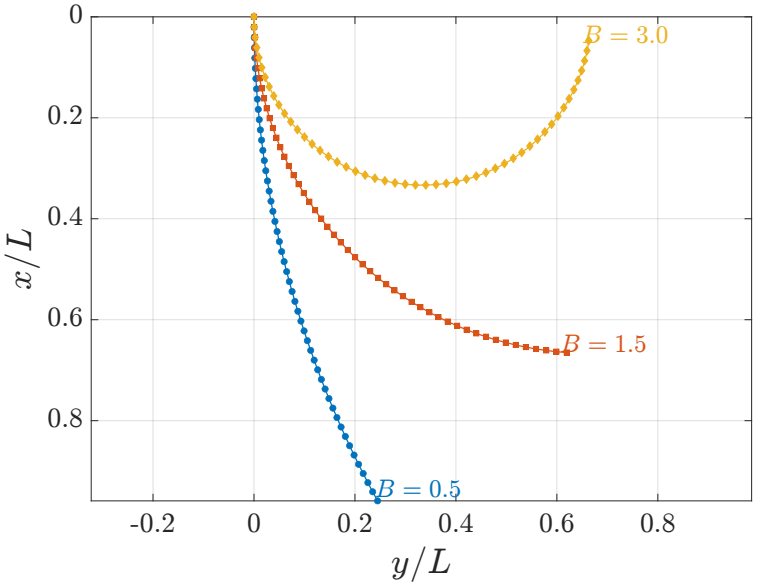


Figure 2.6: Verification of the numerical solution (markers) against the analytical solution (solid lines) for the governing equation of the soft robot in the absence of gravity (Equation 2.35), for $B = 0.5, 1.5$, and 3.

Figure 2.6 compares the shape of the robot in the absence of gravity, obtained from the analytical solution, with the numerical solution computed using the `bvp5c` function for $B = 0.5, 1.5$, and 3. As shown, the numerical solution accurately captures the simplified form of the governing equation for the soft robot.

2.4.2 Parametric Shape Analysis

To investigate the behavior of the robot governed by Equation 2.28, we conduct a parametric analysis by varying the dimensionless parameters B and Γ. Understanding their influence is essential for evaluating the feasibility of using this configuration in flow-driven soft robotic applications, where controllable deformation and compliance are critical.

Figure 2.7 presents two sets of shape profiles. In the first set (left), $\Gamma = 6$ while B is varied. In the second set (right), $B = 2$ while Γ is varied.

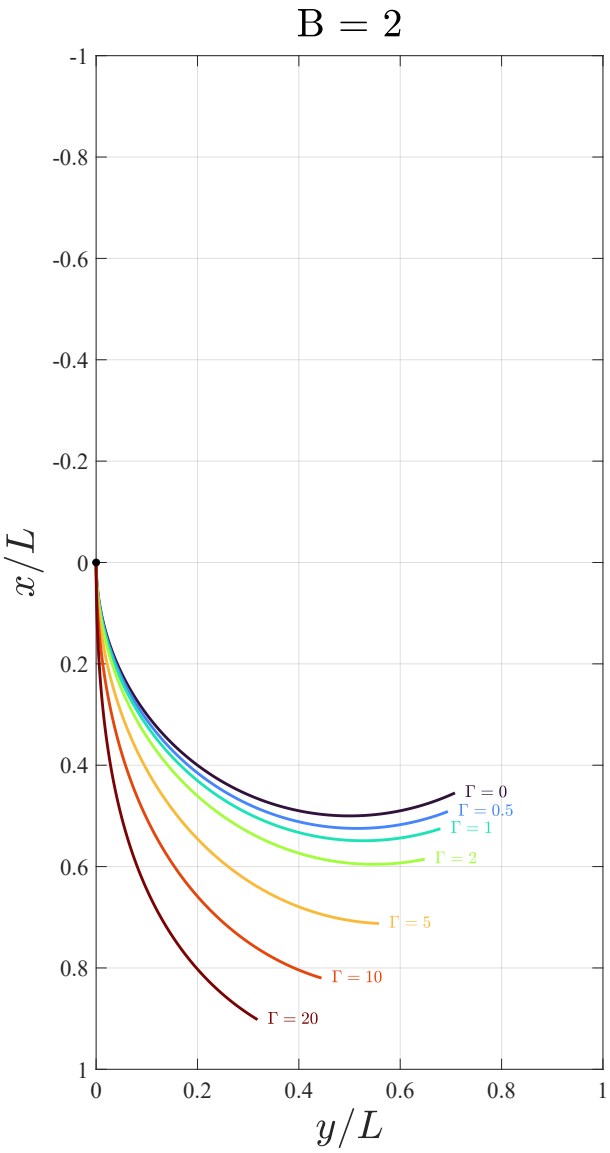
For $\Gamma = 6$, increasing B leads to progressively larger tip deflections and more pronounced

curvature. At low B, the structure remains close to its initially vertical configuration, with minimal lateral displacement. As B increases, the boundary condition applies a larger bending moment at the free end, causing the robot to deflect outward. The resulting shapes demonstrate that B can be used as an effective control input to modulate the posture of the structure.

In contrast, for fixed $B = 2$, increasing Γ results in a gradual straightening of the shape toward the vertical configuration. At $\Gamma = 0$, the structure exhibits significant curvature, as the applied moment dominates in the absence of gravity. As Γ increases, the gravitational effects become more prominent, resisting lateral deflection and suppressing curvature. At high Γ, the robot approaches a nearly straight configuration. This trend highlights a key limitation in soft robotic design: excessive structural weight can significantly reduce the range of achievable deformation, thereby limiting the robot's flexibility. This is, however, applicable when the robot operates in the same plane as the gravity.

The interplay between B and Γ reveals a fundamental trade-off in the design space. While B enables shape control, Γ imposes constraints. For soft robotic applications, it is therefore advantageous to operate in a regime where Γ remains sufficiently low to preserve compliance, while B is tuned to achieve the desired deformation.

From a design perspective, the results suggest that the proposed configuration is feasible for soft robotic implementation, provided that material and geometric parameters are selected to maintain a favorable balance between actuation authority and structural rigidity.



(a) Fixed $\Gamma = 6$, varying B.

(b) Fixed $B = 2$, varying Γ.

Figure 2.7: Equilibrium shapes under parametric variation of B and Γ.

Chapter 3

Computational Modeling

This chapter presents a computational framework developed to further investigate the deformation behavior of the flow-driven soft robot, which is actuated through a single internal channel under varying flow conditions. While initial simulations employed a standard one-way fluid–structure interaction (FSI) approach for its computational efficiency, further analysis revealed that this method was insufficient for accurately capturing the robot's deformation, particularly under elevated outlet pressure conditions. To address this issue, an ad-hoc one-way FSI approach is developed in which a uniformly distributed static pressure is applied on the finite element model of the soft robot. The uniform pressure is obtained from computational fluid dynamics (CFD) simulations, which account for losses in the downstream piping and enable a more realistic representation of the fluid-induced loading. This approach allows for the pressure force to follow the deformation, a requirement which was absent from the standard FSI approach. Overall, this approach effectively captures the interaction governing the robot's behavior, enhancing the reliability of the computational predictions of the robot's performance under controlled actuation.

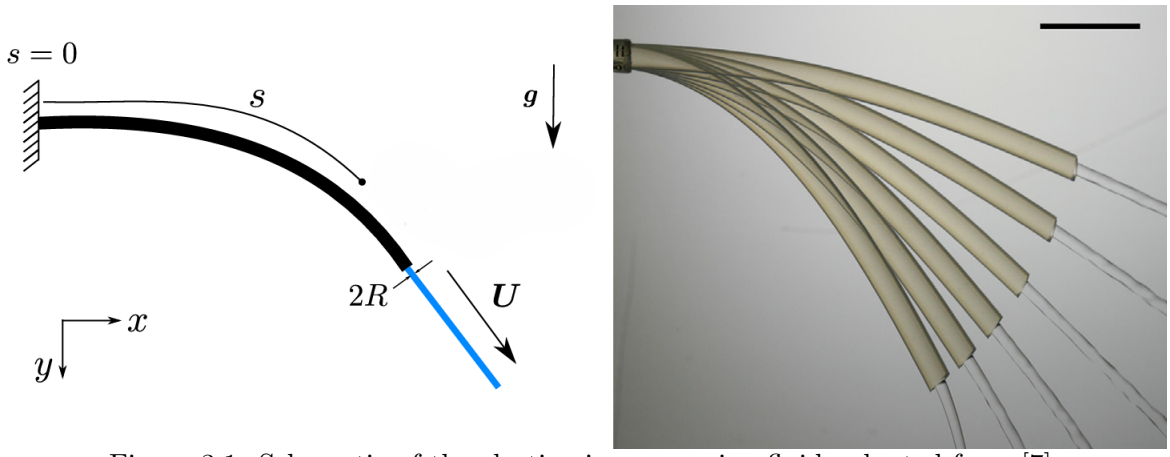


Figure 3.1: Schematic of the elastic pipe conveying fluid, adapted from [7].

3.1 Coupling Investigation for FSI Analysis

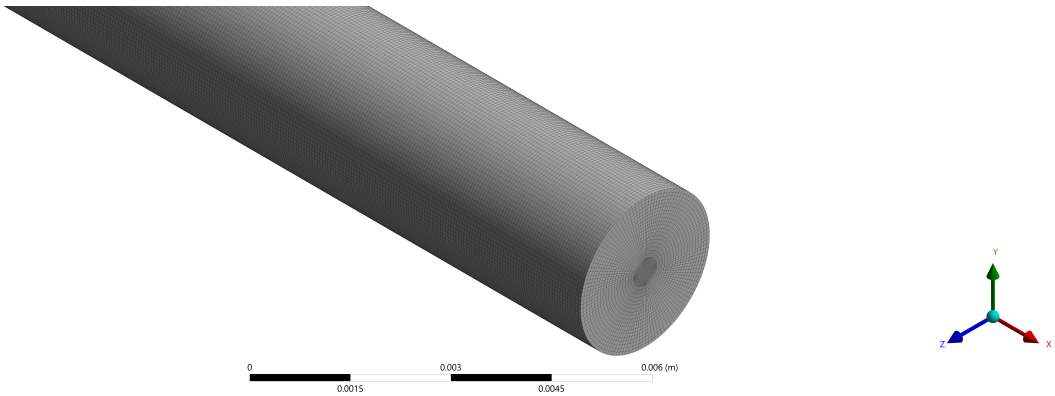
In fluid–structure interaction (FSI) modeling, a key decision involves selecting between one-way and two-way coupling approaches. One-way FSI transfers fluid forces to the structure without accounting for feedback from structural deformation to the fluid domain, whereas two-way FSI incorporates full bidirectional coupling. The choice between these methods depends on the problem's physical characteristics, the required accuracy, and computational feasibility. To inform this decision for the main computational study, a validation case was selected and analyzed using both approaches.

The validation case is based on the experimental work of [7], which investigated the deformation of a horizontal elastic pipe conveying water under gravity along the vertical axis. The pipe is clamped at one end and free at the other, with a total length of 96 mm and an internal radius of 1.36 mm. It is composed of a soft elastic material with a Young's modulus of 5.4 MPa, a second moment of area $I = 3.3$ mm^4, and a linear mass density of 15.9 g/m. The pipe deforms under the combined effects of internal fluid flow and gravity. The objective of this study is to simulate the static planar deflection of the pipe for various flow velocities and compare the results with experimental data.

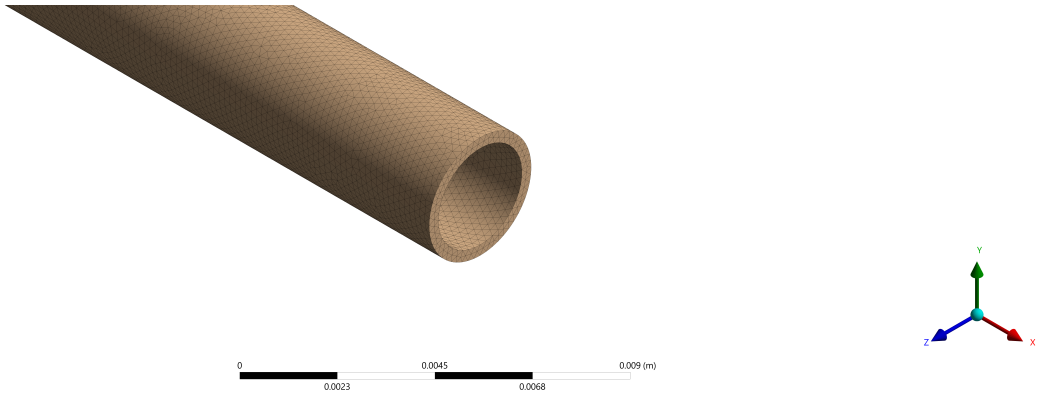
In the standard one-way FSI simulation, the fluid and structural solvers are loosely coupled in a sequential manner. A steady-state CFD simulation is initially conducted by numerically solving the incompressible Navier–Stokes equations under the assumption of turbulent flow, in order to compute the internal flow field. The resulting pressure and wall shear stress distributions on the pipe's inner surface are then mapped onto the structural model in ANSYS Mechanical as external loads. The coupling is unidirectional: the CFD solver (Fluent or CFX) calculates the fluid forces, which are exported and applied to the structural solver. By default, ANSYS Mechanical only maps pressure forces; however, to improve accuracy, the "CFD Results Interpolator" was enabled, allowing both pressure and shear forces to be interpolated and transferred to the structural mesh.

The CFD boundary conditions consisted of a prescribed inlet velocity, zero gauge pressure at the outlet (open to atmosphere), and no-slip wall conditions under the assumption of a smooth surface. For cases involving flow within the turbulent Reynolds regime, the k–ω SST turbulence model was selected due to its robustness in capturing near-wall effects. On the structural side, the inlet was clamped, the outlet was left free, and gravity was applied in the positive y-direction. To account for geometric nonlinearity resulting from large deformations, the "Large Deflection" option was enabled in ANSYS Mechanical, as the pipe experienced strains beyond the small-deformation regime.

Mesh quality played a critical role in ensuring both stability and accuracy of the simulations. A balance was maintained between mesh resolution and computational cost. The CFD domain was discretized using approximately 5.1 million hexagonal quadratic elements. The height of the first mesh layer adjacent to the pipe wall yielded average Y^+ values ranging from 2.32 to 5.93, as the flow velocity increases. Since for the main study (soft robot) it is required to use unstructured meshes due to geometrical complexity, the structural meshing



(a) CFD grid

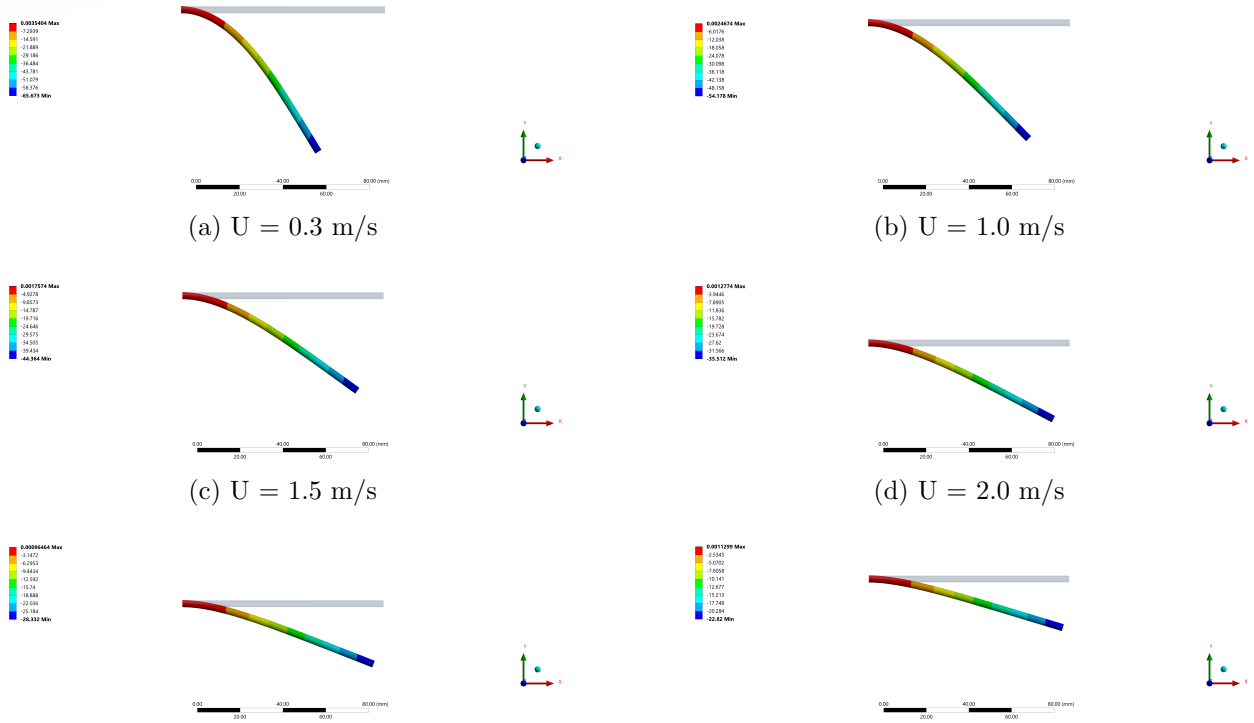


(b) Structural FEA grid

Figure 3.2: Isometric view of the CFD and structural FEA meshes used in the validation study of a fluid-conveying pipe.

was intentionally performed using a 305,000 unstructured mesh elements with a quality rating of 85.1%, optimized for nonlinear mechanical behavior. Figure 3.2 illustrates the CFD and FEA meshes used in this validation study.

Figure 3.3 illustrates the deformed shape of the pipe for various flow velocities. The tip deflection values are compared with the experimental measurements in Table 3.1. The pipe shape was reconstructed using the centerline, as shown in Figure 3.4. At a flow velocity of $U = 0.3$ m/s, where gravitational effects dominate, the simulation shows excellent agreement with the experiment, with a tip deflection error of only 0.1%. As the flow velocity increases,



(a) U = 0.3 m/s

(b) U = 1.0 m/s

(c) U = 1.5 m/s

(d) U = 2.0 m/s

(e) U = 2.5 m/s

(f) U = 3.0 m/s

Figure 3.3: Lateral deflection (mm) of the pipe for various flow velocities.

Table 3.1: Comparison of simulated and experimental (see [7]) tip deflection values for the elastic pipe at various flow velocities.

U (m/s)	**Re**	$y_{\mathbf{exp}}$ **(cm)**	$y_{\mathbf{num}}$ **(cm)**	**Error (%)**
0.3	812	6.50	6.49	0.1%
1.0	2707	6.06	5.41	10.7%
1.5	4060	5.42	4.38	19.2%
2.0	5414	4.58	3.45	24.7%
2.5	6767	3.48	2.74	21.1%
3.0	8121	2.23	2.22	0.3%

fluid-induced forces become more significant, leading to larger deviations from the experimental results. However, at higher velocities, particularly $U = 2.5$ m/s and $U = 3.0$ m/s, the pipe assumes a more horizontal configuration. In this orientation, the flow path more closely match those assumed in the CFD simulation, resulting in reduced errors.

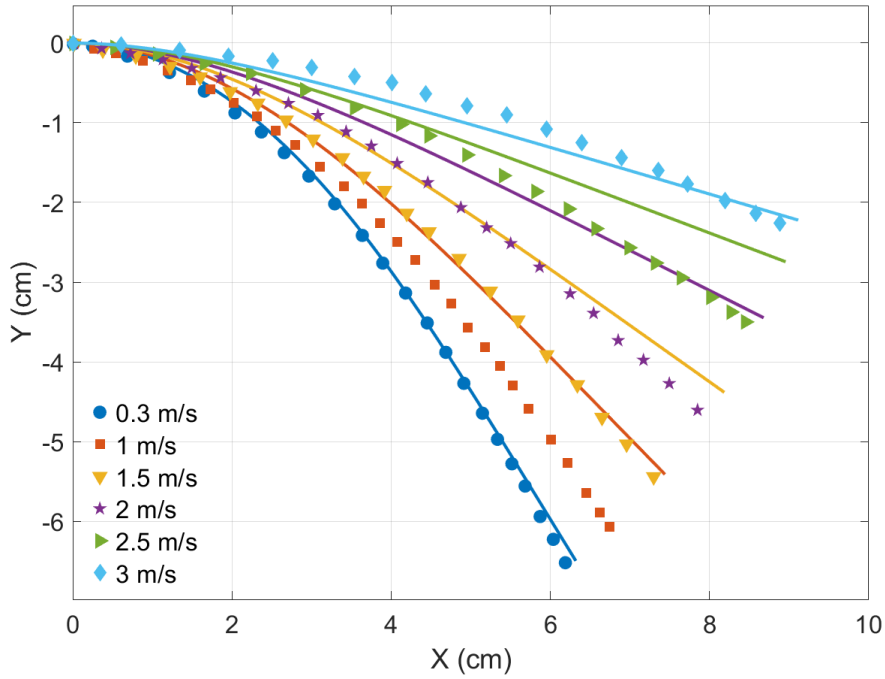


Figure 3.4: Side view of the elastic pipe centerlines for different flow velocities. Symbols represent experimental data from [7], and lines are the standard one-way FSI results.

Subsequently, the one-way FSI approach was modified by replacing the ANSYS force-mapping with constant follower loads on the inner wall: a uniform normal pressure and a tangential shear load that follow the local wall orientation,

$$\begin{aligned} f_N &= P_{\text{in}} - P_{\text{out}} =, \\ f_\tau &= \frac{P_{\text{in}} - P_{\text{out}}}{L_p} A \end{aligned} \tag{3.1}$$

where P_{in} and P_{out} are the inlet and outlet pressures, A is the flow cross-sectional area, and L_p is the pipe length. Since the pipe discharges to the surroundings, $P_{\text{out}} = 0$. These loads were derived through a control volume analysis of a differential element, consistent with the methodology presented in Chapter 2. Table 3.2 compares the experimental tip displacement with the simulation results from this modified approach. As shown in Figure 3.5, the modified one-way FSI accurately captures the pipe's centerline shape, demonstrating better

Table 3.2: Comparison of the modified FSI and experimental (see [7]) tip deflection values for the elastic pipe at various flow velocities.

U (m/s)	**Re**	$y_{\mathbf{exp}}$ **(cm)**	$y_{\mathbf{num}}$ **(cm)**	**Error (%)**
0.3	812	6.50	6.61	1.6%
1.0	2707	6.06	6.28	3.6%
1.5	4060	5.42	5.67	4.6%
2.0	5414	4.58	4.84	5.8%
2.5	6767	3.48	3.69	6.0%
3.0	8121	2.23	2.41	8.1%

effectiveness than the standard method on ANSYS.

In contrast to the one-way approach, the two-way FSI simulation involves a fully coupled, iterative exchange of data between the fluid and structural solvers. This is implemented in ANSYS Workbench using the System Coupling module, which links ANSYS Fluent (or CFX) with ANSYS Mechanical. At each iteration or time step, the CFD solver computes the fluid

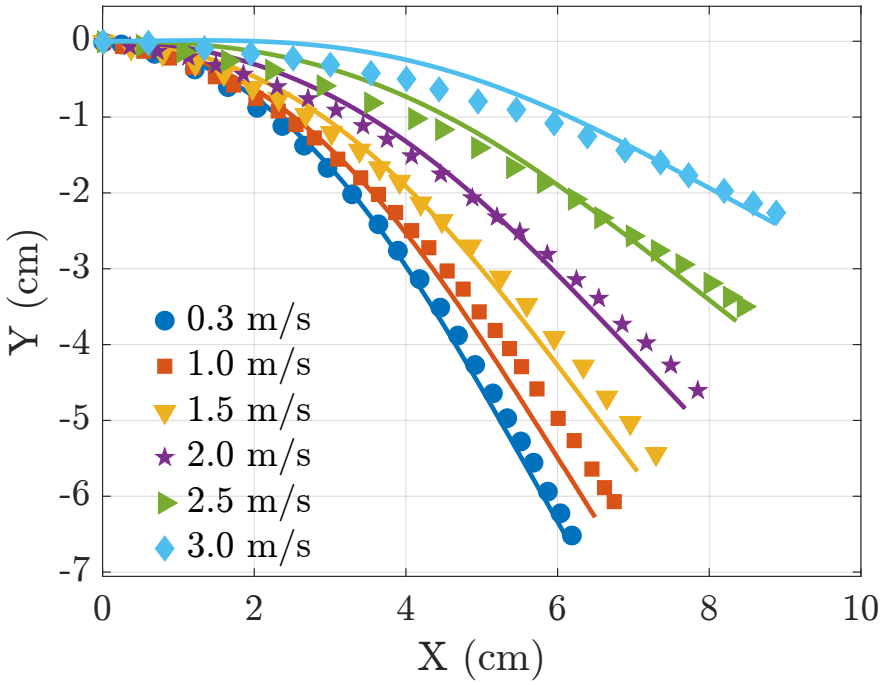


Figure 3.5: Side view of the elastic pipe centerlines for different flow velocities. Symbols represent experimental data from [7], and lines are the modified one-way FSI results.

forces and transfers them to the structural solver. The structural solver then calculates the resulting deformation and updates the mesh, which is sent back to the CFD solver to update the fluid domain. This loop continues until convergence is achieved at each coupling step.

This bidirectional coupling enables accurate modeling of the interactions between fluid flow and structural response, particularly in cases involving large deformations or dynamic effects. However, despite its theoretical advantages, the two-way FSI simulation failed to converge due to numerical instabilities.

According to the ANSYS documentation, such convergence issues are common in scenarios involving thin structural components, materials with low Young's modulus, and fluid–solid systems with similar densities. All these conditions apply to the present case: the pipe wall is only 0.3 mm thick, the material is a soft rubber-like elastic solid, and the density of the pipe is comparable to that of water. These factors contribute to instability in the coupling algorithm, especially when large deformations lead to mesh distortion and nonlinear feedback between solvers.

3.2 Turbulence Model Study

In CFD, turbulence modeling plays a critical role in enabling the simulation of complex flows without the prohibitive cost of resolving all scales of motion. Turbulence models provide a mathematical framework to approximate the influence of unresolved, fluctuating eddies on the mean flow. By averaging the Navier–Stokes equations, these models introduce additional terms that represent the net effect of turbulent fluctuations on momentum and energy transport. This approach allows for the prediction of realistic flow behavior in engineering systems while keeping computational demands within practical limits. Although the specific formulations vary, the essential purpose of turbulence models remains the same: to bridge

the gap between the scales resolved by the computational mesh and the full complexity of turbulent motion.

To determine an appropriate turbulence model for the main simulation study, featuring U-shaped channels with straight upstream and downstream segments connected by a U-bent segment, a validation study was conducted based on the experimental work of Anwer and So [8]. Their study investigated turbulent flow behavior in a 180-degree curved pipe. The experimental setup consisted of a pipe with an internal diameter of 76.2 mm, straight inlet and outlet sections each 7.3 m long, and a U-bend with a mean radius of 495.3 mm. Air was blown through the pipe at a bulk velocity of $W_0 = 10.4$ m/s, with a kinematic viscosity of $\nu = 1.58 \times 10^{-5}$ m^2/s, resulting in a Reynolds number of 50,000.

The experiment was modeled in three dimensions, and various turbulence models were applied in both steady-state and transient simulations. The turbulence intensity at the inlet was calculated to be 3.77%, based on the root-mean-square (RMS) of the turbulent velocity fluctuations reported in the experimental data. The primary quantity of interest was the pressure coefficient, defined as:

$$C_p = \frac{P - P_0}{0.5\rho W_0^2}, \tag{3.2}$$

where P is the local static pressure and P_0 is the reference static pressure measured 18 pipe diameters upstream of the curve. Pressure coefficients were computed at various locations along the pipe and compared to the experimental measurements. The table of results are summarized in Appendix B. Figures 3.6 and 3.7 show the mid-plane velocity magnitude contours and wall pressure distribution, respectively, obtained using the k-ω SST turbulence model.

The simulation results were consistent with the findings reported by Kalpakli et al. [52], which concluded that RANS models generally perform poorly in capturing the complex flow

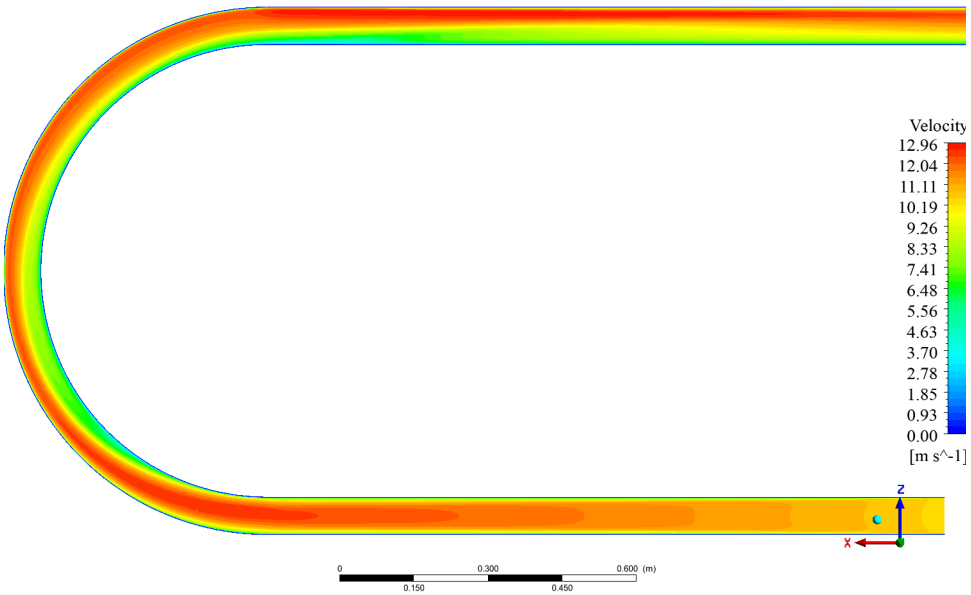


Figure 3.6: Velocity magnitude at the mid-plane of the pipe, simulating the experiment by Anwer and So [8].

behavior in curved geometries. Among the models tested, the k-ω SST model showed relatively better agreement with the experimental pressure coefficients, particularly downstream of the curve. The average error in predicted pressure coefficients downstream of the curve was less than 10% on the outer wall and under 7% on the inner wall. In the bend region itself, the errors were below 30% and 19%, respectively. In contrast, Reynolds Stress Models (RSM) produced the most inaccurate results, significantly overpredicting the pressure ratios.

The article by Kalpakli et al. [52] also reported that Large Eddy Simulation (LES) results were in close agreement with experimental data, both in terms of statistical quantities and instantaneous flow structures. However, LES introduces substantial complexity and computational cost. It requires a much finer mesh, particularly near walls and in regions of high curvature, and imposes strict time-stepping constraints to maintain a Courant number of 1 or less. These requirements make LES impractical for the full-scale simulation of the soft robotic system considered in this study.

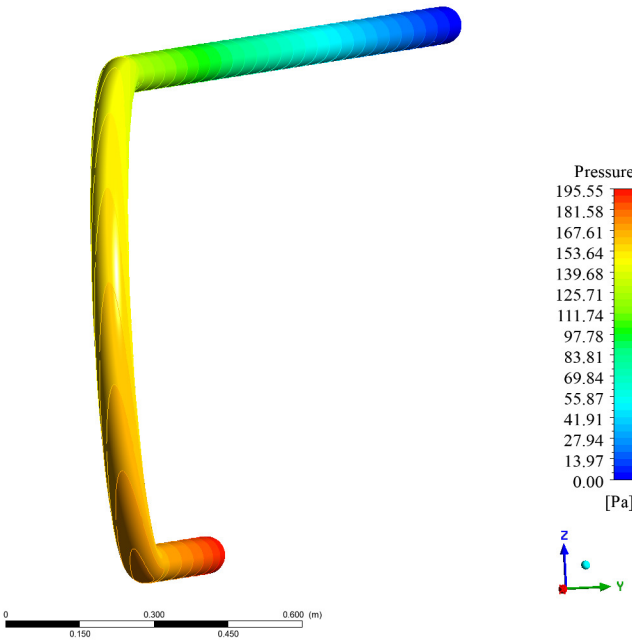


Figure 3.7: Wall pressure contour for the pipe simulated according to the experiment by Anwer and So [8].

Given the trade-offs between accuracy and computational feasibility, and based on the validation results, the k-ω SST RANS model was selected for the soft robot simulations. While it does not capture all the intricacies of turbulent flow in curved geometries, it provides a reasonable compromise between predictive capability and computational efficiency for the types of flows and geometries involved in the soft robot simulations.

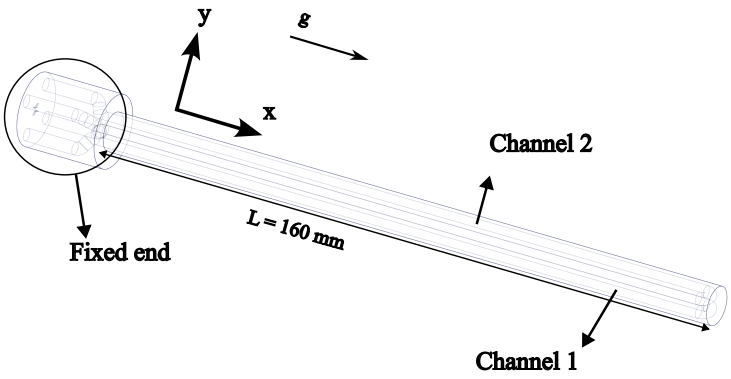
3.3 Fluid-Structure Interaction Study for The Soft Robot

3.3.1 Robot Design

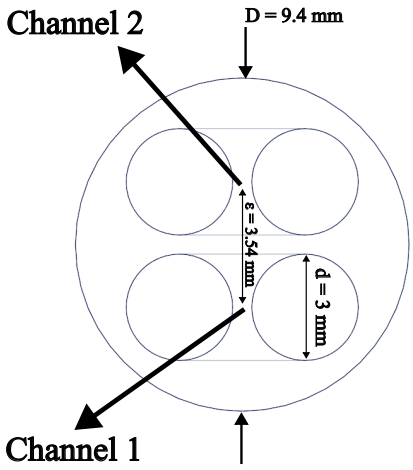
Before conducting simulations and experiments, a CAD model of the soft robot was developed to verify both its practical feasibility for fabrication and its compatibility with computational analysis. The main body of the robot features a cylinder with an external diameter of 9.4 mm and a length of 160 mm. Internally, it contains two parallel U-shaped channels carved into the body to account for actuation in both directions. Each channel has a diameter of 3 mm, and they are spaced symmetrically with respect to the centerline and 3.54 mm apart. To accommodate hydraulic fittings, an extension was incorporated into the fixed end. This compact configuration was deliberately chosen to better accommodate the spatial constraints of real-world applications such as navigating through the human body during minimally invasive procedures, where minimizing contact with surrounding tissues is critical. A schematic of the CAD model is presented in Figure 3.8.

3.3.2 CFD Analysis

For the computational modeling, only one of the internal channels (Channel 1) was considered, assuming for single-channel actuation (see Figure 3.9). To accurately account for pressure losses in the fluid domain (in accordance with the experimental setup described in Chapter 4), the downstream piping of the robot was considered. The CFD simulations were performed using ANSYS Fluent (2024R2). Prior to running the simulations, a mesh independence study was conducted to ensure that the results were not sensitive to mesh resolution. A steady inlet velocity of 4.72 m/s, equivalent to a volumetric flow rate of 2 L/min, was applied, with a relative pressure of 0 Pa at the outlet. Structured hexahedral elements were generated through global refinement with element counts ranging from 200,000 to 13 million.



(a) 3D view



(b) Side view

Figure 3.8: Schematic of the flow-driven soft robot used in this study, illustrating the two parallel U-shaped internal channels.

The total force exerted along the channel length was selected as the monitoring variable, and results were compared against those obtained from the finest mesh (13 million elements). Table 3.3 presents the exerted force for each mesh grid. A mesh with approximately 2 million elements, shown in Figure 3.10, was found to provide a good balance between accuracy and computational efficiency with regards to channel wall pressure and forces, and thus, it was selected for the CFD study.

As for the boundary conditions, various flow velocities was set on the inlet of the robot

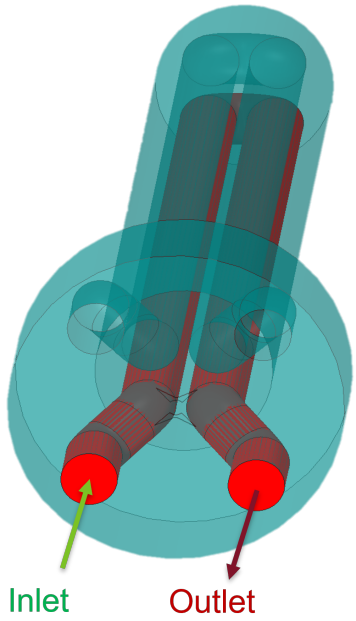


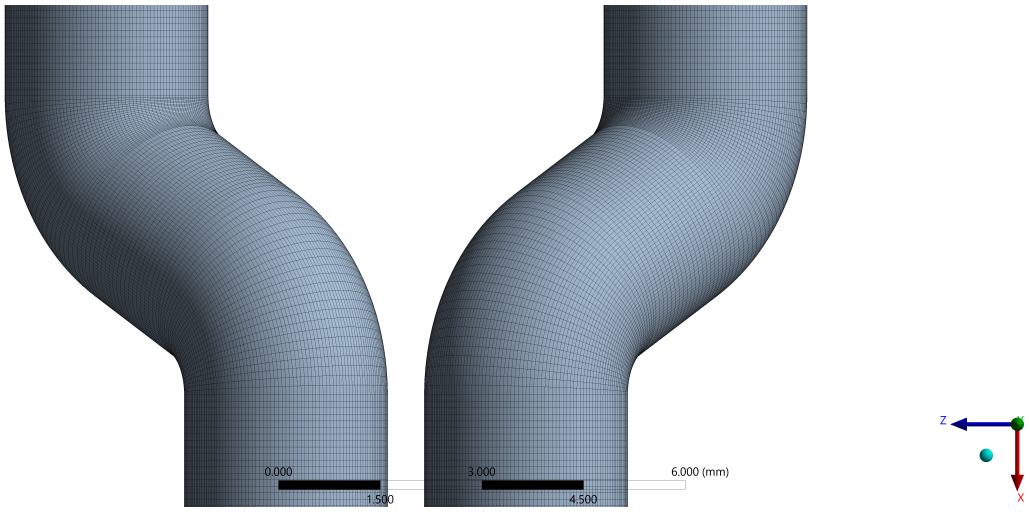
Figure 3.9: Visualization of the soft robot with two symmetrically placed channels. The region which is highlighted in red, shows the flow path where water enters and exits the robot through the channel placed the bottom, as defined in the CFD simulation setup.

Table 3.3: Mesh element study for the CFD analysis.

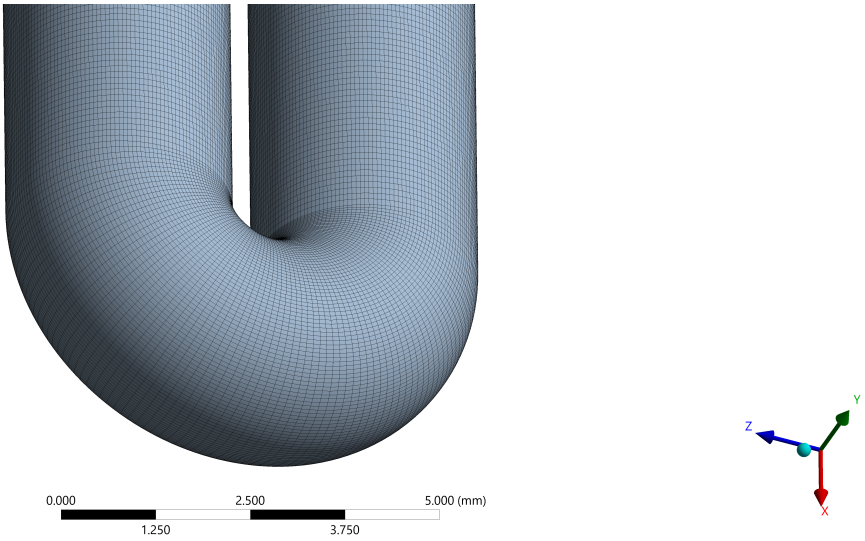
No. Elements	Y^+	F_τ (N)	**Error (%)**
12,932,788	8.5	0.827	-
8,909,824	9.6	0.823	0.47%
4,064,688	8.2	0.816	1.40%
1,947,010	**16.0**	**0.842**	**1.83%**
1,664,209	23.0	0.826	0.20%
219,744	26.2	0.707	14.58%

and static pressure of 0 Pa was applied at the discharge of the downstream to simulate for the water flow being discharged to the surrounding.

Figures 3.11 and 3.12 illustrate the velocity and static pressure fields in the mid-plane of the channel for two representative flow velocities: $U = 1.61$ m/s and $U = 4.65$ m/s. Although the magnitude of the velocity and pressure vary with flow rate, the overall flow behavior remained consistent. Upstream of the U-bend, the flow was fully developed, exhibiting an



(a) CFD mesh of the flow channel close to the clamp end

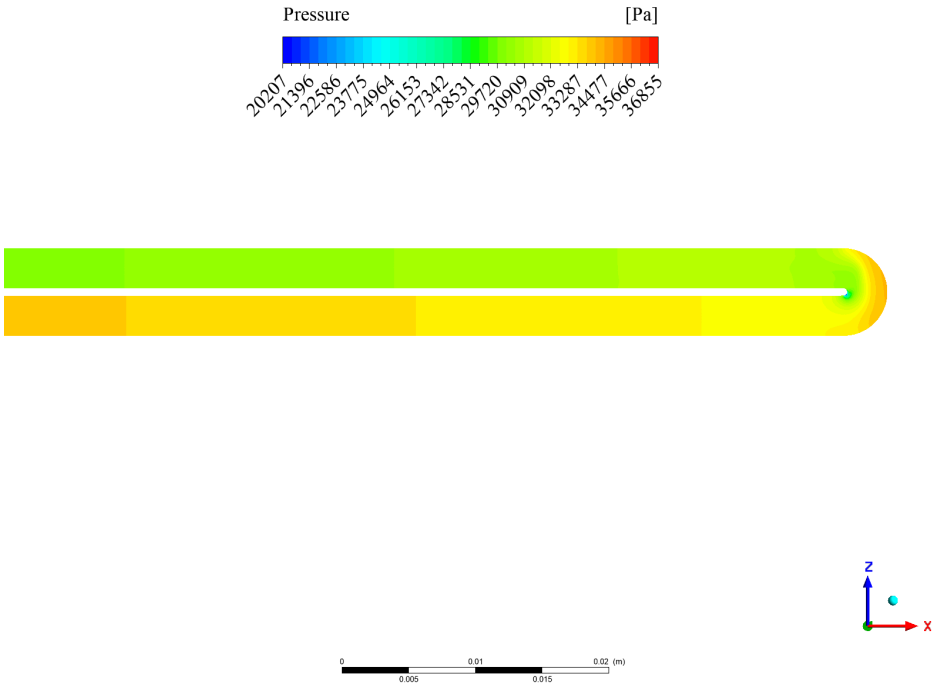


(b) CFD mesh of the flow channel close to the free end

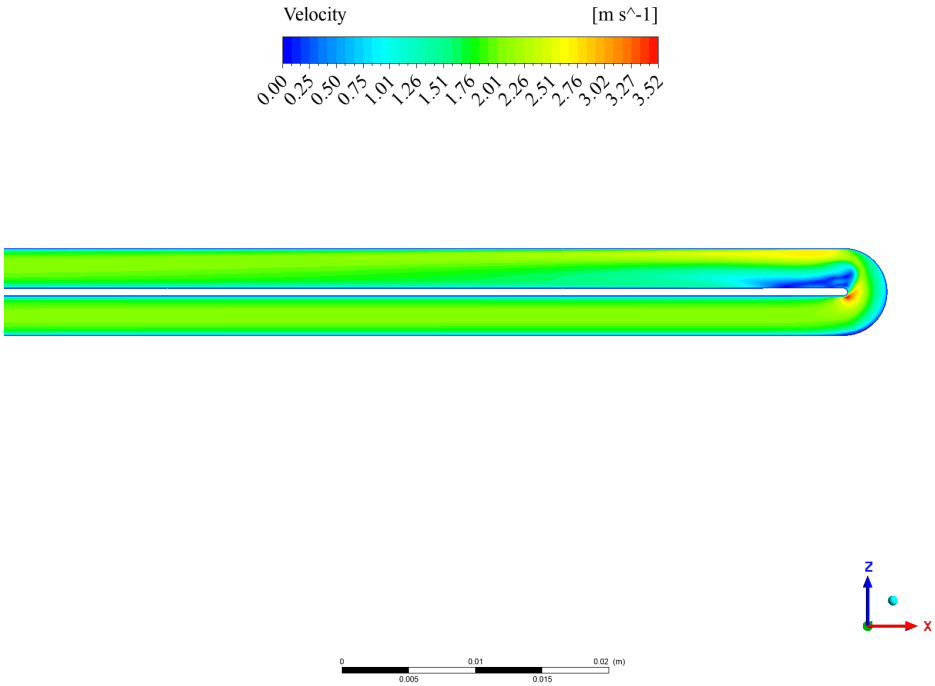
Figure 3.10: CFD mesh of the flow channel at both ends

almost constant velocity profile. This confirms the validity of the plug flow assumption in the analytical model in Chapter 2. Approximately one channel diameter before the bend, the flow began to curve in response to the geometry, initiating flow separation and the formation of a recirculation region. The Dean number, which quantifies the strength of secondary flows in curved channels, in the curved section of the flow ranges between $3222 < De < 12888$

for the tested flow velocities of $1.61 < U < 4.65$. Downstream of the bend, a jet stream developed near the outer wall of the channel, gradually dissipating as the flow re-established a fully developed profile. Symmetric secondary flows were observed up to a few diameters close to the U-bend, as illustrated in Figure 3.13, highlighting the three-dimensional nature of the flow.

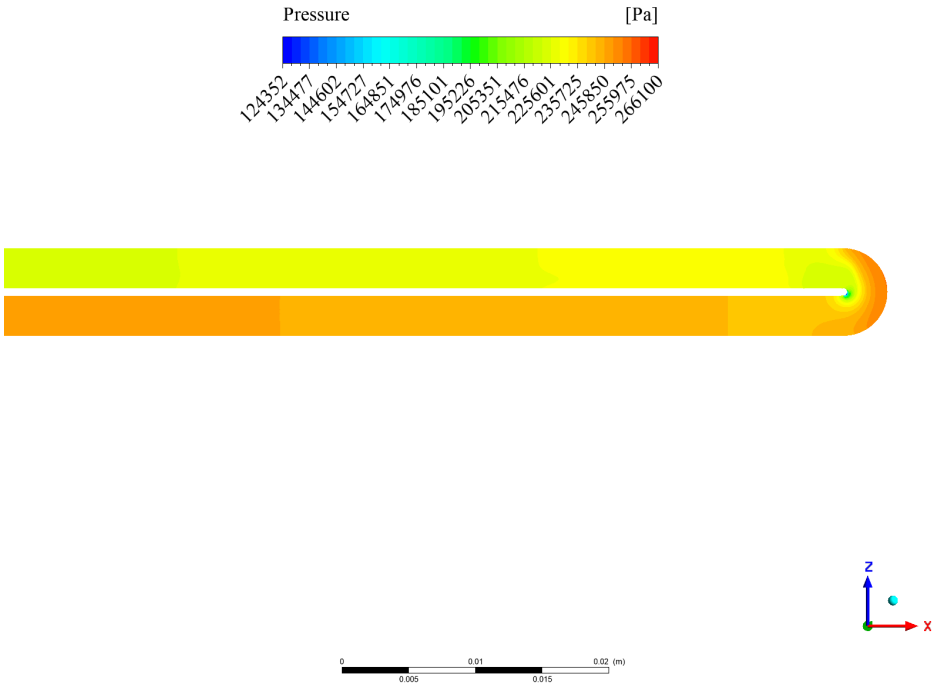


(a) Pressure

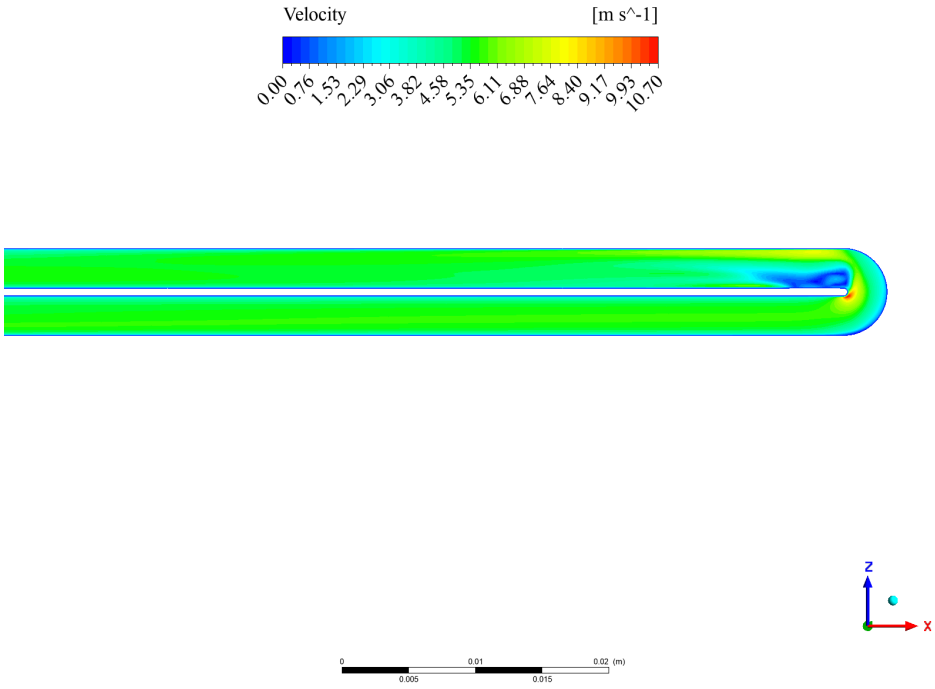


(b) Velocity

Figure 3.11: Static pressure and velocity magnitude contours at $U = 1.61\,\mathrm{m/s}$.



(a) Pressure



(b) Velocity

Figure 3.12: Static pressure and velocity magnitude contours at $U = 4.65\,\mathrm{m/s}$.

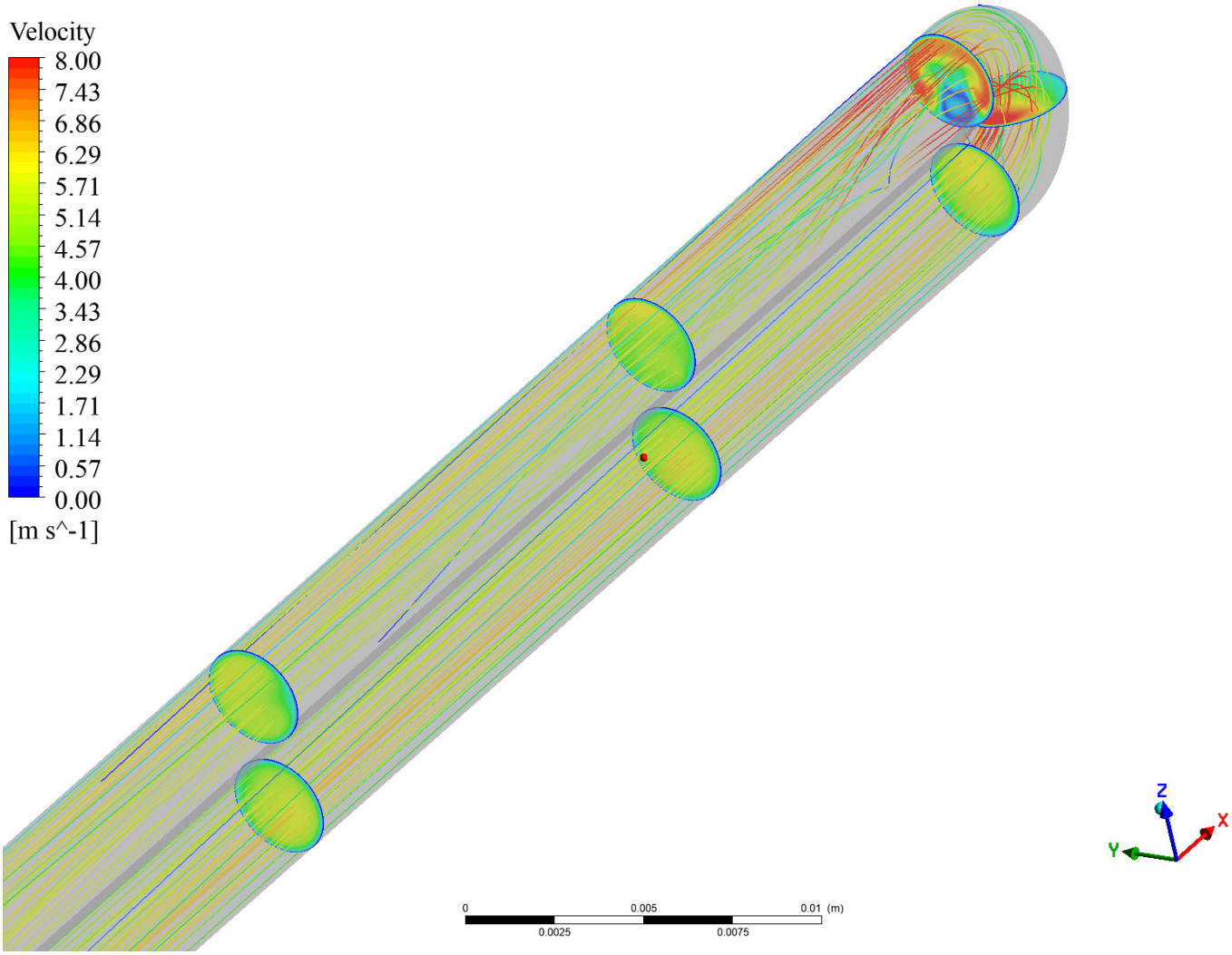


Figure 3.13: Symmetric secondary flow jets after the U-bent segment at a flow rate of $U = 4.65\,\mathrm{m/s}$. The flow transitions back to fully developed after traveling a certain distance in the downstream segment.

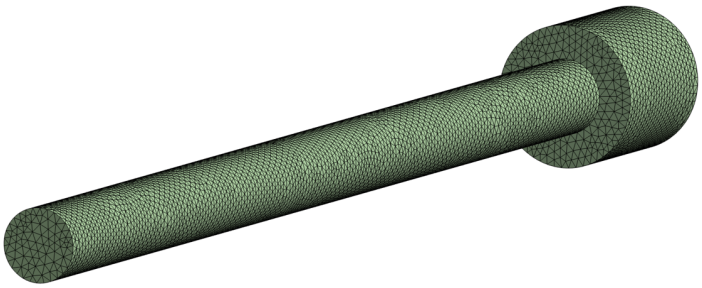


Figure 3.14: Finite-element mesh (106,000 unstructured elements) generated with the `Nonlinear Mechanics` physics preference.

3.3.3 Structural FEA Analysis

Before importing the CFD results into ANSYS Mechanical for structural finite-element analysis (FEA), the material test data from Section 4.1.2 were used to fit a third-order Yeoh constitutive model for the silicone elastomer. The Yeoh model was selected because it captures a wide range of large-deformation behaviors using only uniaxial test data [53].

For the structural simulations a mesh was generated with the `Nonlinear Mechanics` physics preference and a target element size of 1 mm (Figure 3.14). Because the structure undergoes large deformations, nonlinear adaptive remeshing was enabled to avoid excessive element distortion. The solver was configured to remesh automatically when mesh quality metrics exceeded predefined limits (skewness threshold of 0.9 and a Jacobian ratio limit of 0.1), and remeshing was allowed at every step to maintain solution stability and accuracy.

Substepping was activated so loads could be introduced gradually; the initial, minimum and maximum substep counts were set to 20, 2 and 1,000,000, respectively. The analysis used a `Large` deformation formulation and a quasi-static solution strategy, which improves convergence by applying internal channel forces slowly and thereby neglects inertial effects. Gravity was applied along the undeformed robot's longitudinal axis to reproduce the vertical, base-suspended orientation used in the experiments. The robot base was fixed to represent a clamped boundary condition, effectively modeling the structure as a cantilevered beam.

As will be shown in Chapter 5, results from the standard one-way FSI approach did not agree with the experiments. In the standard procedure, the force distribution from the CFD model is mapped directly onto the channel wall using ANSYS's `Imported Load` feature; however, we observed that the mapped load did not remain normal (or tangent) to the wall as the geometry deformed. To address this, a modified one-way FSI procedure was implemented in which follower-type pressure load was applied to the channel wall in the FEA model. This modification allows the applied pressure to follow the wall deformation and thus increases the physical fidelity of the simulation.

From Equation 2.7 it follows that the combined inlet–outlet pressure, $(P_{in} + P_{out})$, dominates the applied loading. Accordingly, the pressures at the inlet and outlet were extracted from the CFD simulations, summed to produce a constant pressure value, and applied to the channel wall as a normal follower load in the structural model. Figures 3.15 and 3.16 show the deflection of the soft robot under the flow actuation, using the modified FSI approach.

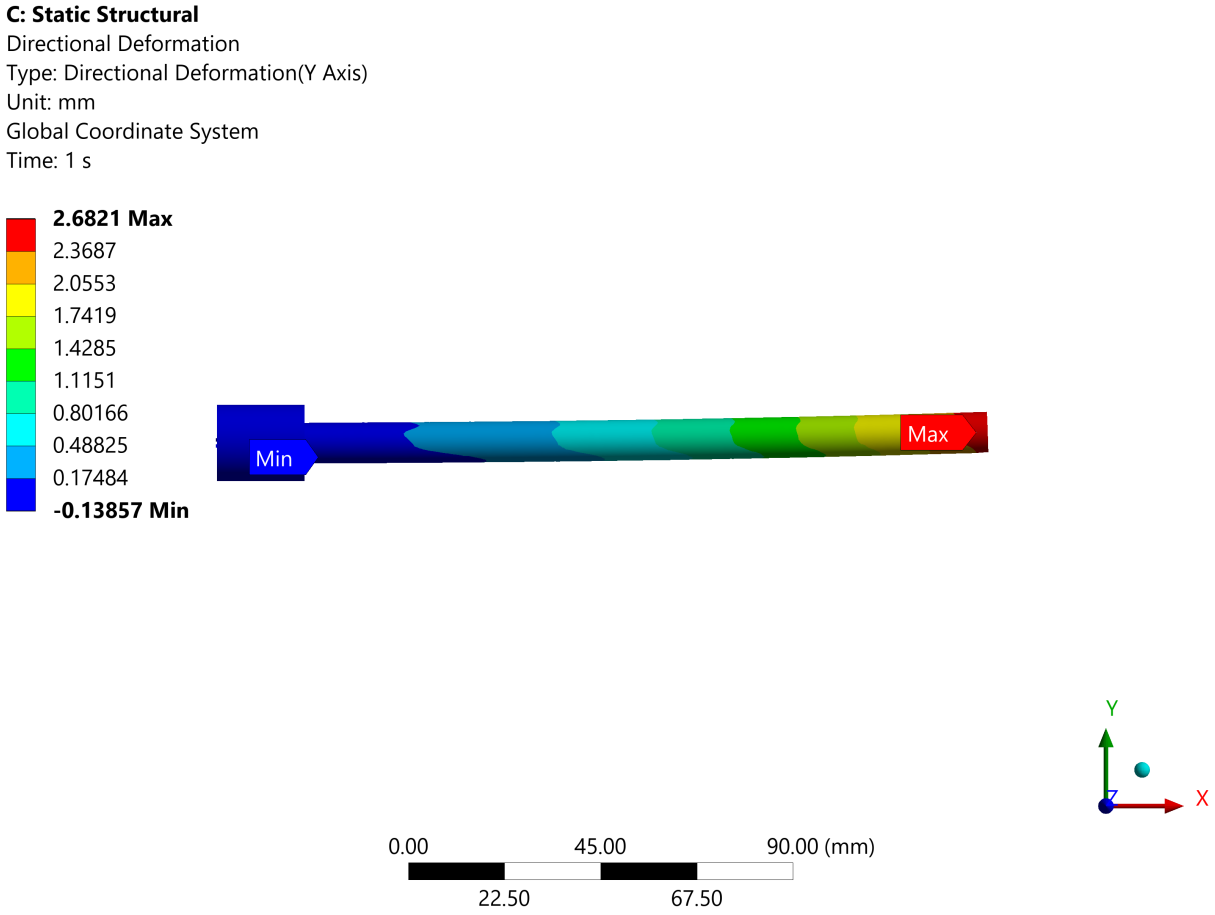


Figure 3.15: Lateral deflection contour of the flow-driven soft robot at $U = 1.61\,\mathrm{m/s}$ using the modified one-way FSI approach.

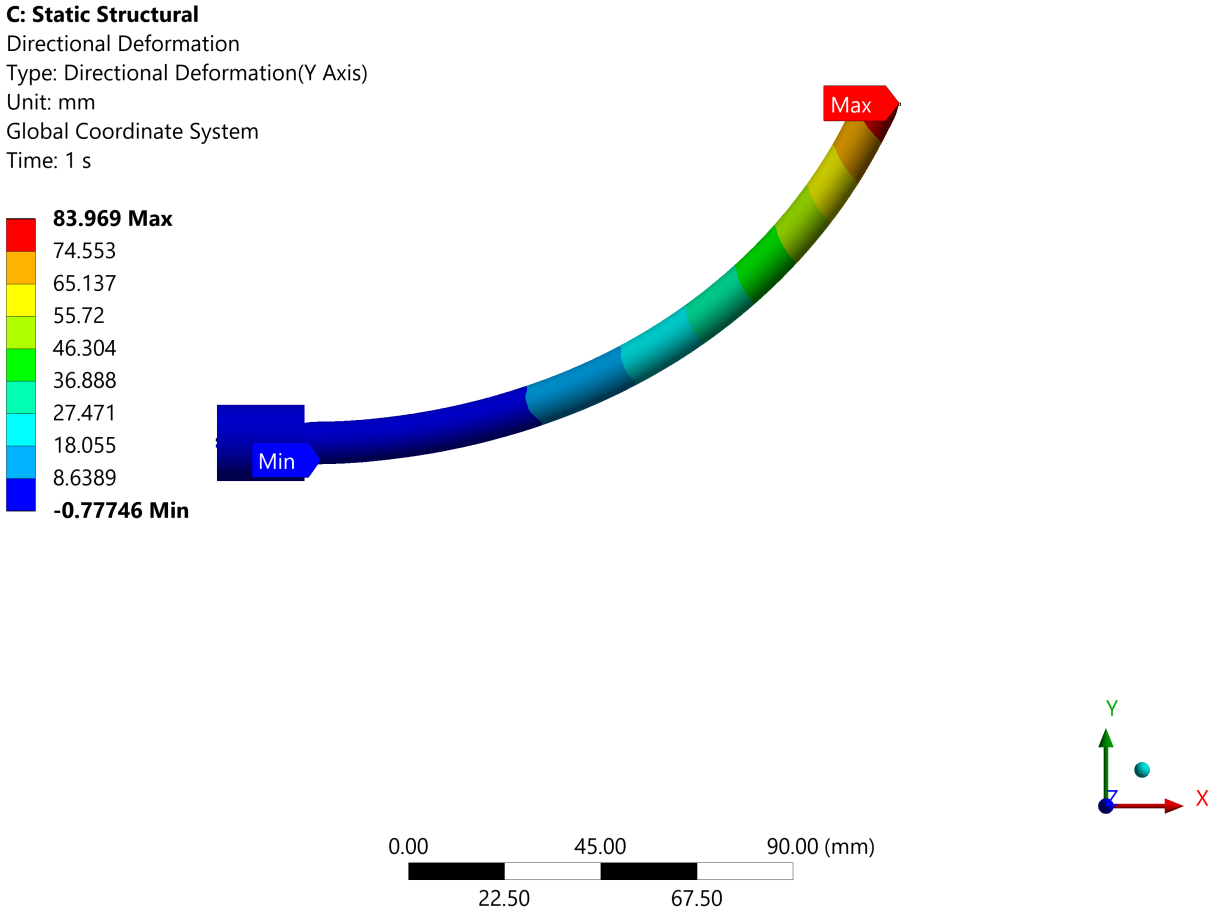


Figure 3.16: Lateral deflection contour of the flow-driven soft robot at $U = 4.65\,\mathrm{m/s}$ using the modified one-way FSI approach.

Chapter 4

Experimentation

This chapter presents the experimental framework developed to fabricate, assemble, and investigate the deformation behavior of a flow-actuated soft robot. Two distinct fabrication methods were employed: stereolithography (SLA) 3D printing, which enabled high-resolution prototyping, and a custom-designed injection molding process, introduced as an alternative approach to support repeatable and scalable manufacturing. To evaluate the performance of the robot, a dedicated test rig was constructed, incorporating a hydraulic supply system and an electronic data acquisition setup. Flow rate and deformation were continuously monitored using a calibrated flow sensor and a high-resolution camera, both interfaced with an Arduino-based control system to ensure synchronized data collection. This experimental setup enabled direct comparison with the theoretical model, developed in Chapter 2, by extracting the shape of the robot from recorded image data and provided a reliable platform for validating the robot's operating capabilities under various flow conditions.

4.1 Prototype Development

4.1.1 Additive Manufacturing

Stereolithography (SLA) 3D printing was selected for this study since it is fast, and it can produce complex, high-resolution, smooth surface finish, watertight geometries which is an essential requirement for fluidic actuation in soft robotics. SLA works by selectively curing layers of liquid photopolymer resin using a focused light source. This process enables the fabrication of highly detailed components with minimal porosity and superior dimensional fidelity. Compared to fused deposition modeling (FDM), which builds parts by extruding thermoplastic filament, SLA offers significantly finer feature resolution and smoother surfaces, making it particularly advantageous for producing soft actuators and embedded fluidic channels.

The Formlabs 3B+ SLA printer used in this study employs an inverted printing configuration, where the build platform lifts the part upward from the resin tank as each layer is cured. This setup allows precise control over layer thickness and exposure time, achieving dimensional accuracy within ± 0.1 mm and layer heights ranging from 25 to 100 microns. A key benefit of SLA in soft robotic fabrication is its ability to create sealed, continuous internal channels without leakage, which is critical for hydraulic and pneumatic actuation systems. The printer's compatibility with biocompatible and flexible resins further enhances its suitability for prototyping soft robotic components intended for minimally invasive surgical applications. Figure 4.1 shows the Formlabs 3B+ printer used in this study.

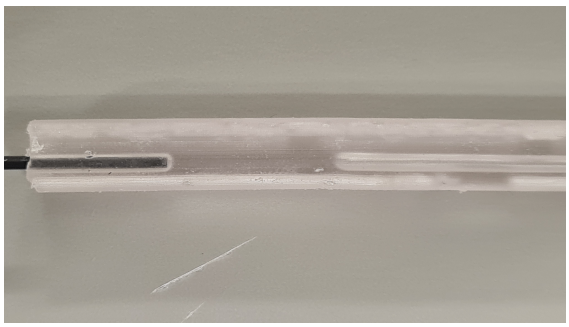
The soft robot was printed using Formlabs' Silicone 40A Resin, which offers a Shore hardness of 40A, elongation at break of 230%, tear strength of 12 kN/m, and excellent fatigue resistance, surviving over 500,000 flexing cycles at room temperature. These properties make the resin ideal for applications requiring flexibility and durability. The resin also provides



Figure 4.1: Formlabs 3B+ SLA printer used for rapid prototyping.



(a) Delamination and layer shifting



(b) Channel blockage

Figure 4.2: Common issues encountered during SLA printing with soft materials.

good chemical resistance and thermal stability from –25°C to 125°C.

Print orientation plays a critical role when working with flexible materials. Although vertical or steep-angle orientations can improve accuracy, the flexibility of Silicone 40A may lead to deformation during the peel and recoat steps of the SLA process. As the build platform lifts and re-submerges, soft geometries may stretch or shift, causing misalignment and delamination (Figure 4.2a). To mitigate this, the robot was printed at a 10-degree angle relative to the build platform, with the base positioned closer to improve stability and reduce unsupported overhangs. Support structures were generated using Formlabs' PreForm software, and approximately 40 mL of resin was consumed per print, including supports.



Figure 4.3: The 3D-printed robot in its uncured "green" state.

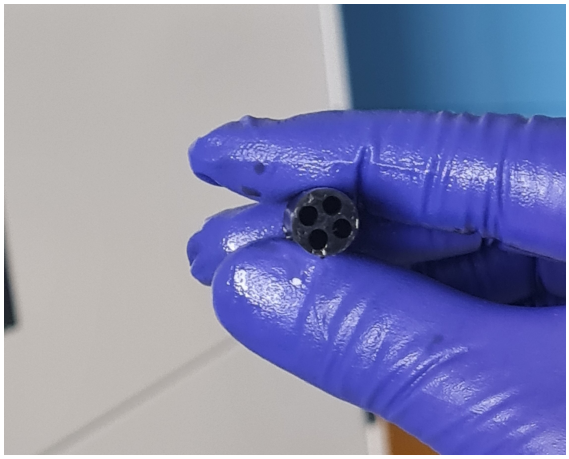
After printing, the part remained in a "green" state (Figure 4.3), meaning it had not yet achieved its full mechanical properties. Post-processing was required to develop its final strength and elasticity. Prior to curing, the part was thoroughly cleaned to remove uncured resin and ensure unobstructed internal channels. A failed print with a blocked channel due to residual resin is shown in Figure 4.2b. Cleaning was performed using a solvent mixture of 80% isopropyl alcohol (IPA, 99%) and 20% n-butyl acetate, repeated in three 10-minute cycles. Compressed air was used after each wash to flush the channels and remove residual solvent.

Support structures were removed using a sharp cutter. Due to the softness of the silicone, care was taken to avoid tearing or puncturing the part, which could compromise the embedded channels. The part was then air-dried at room temperature, away from direct sunlight to prevent premature UV exposure.

Final curing was performed using the Form Cure L system. The printed robot was fully submerged in a glass beaker filled with water at room temperature and cured at 60°C for 45



(a) Flexibility of the robot under gravity



(b) Robot cross-section

Figure 4.4: 3D-printed flow-driven soft robot fabricated with silicone resin: (a) demonstration of prototype flexibility under gravitational force, (b) cross-sectional view showing two embedded flow channels.

minutes, as recommended for Silicone 40A. This water-submersion method ensures uniform heat distribution and prevents surface oxidation. After curing, the part was allowed to cool gradually to room temperature before handling or testing. The mechanical flexibility and internal structure of the printed robot are shown in Figure 4.4, including a side view under gravity and a cross-sectional view of the embedded channels.

4.1.2 Tensile Test on Silicone Specimen Using DIC Method

To gain an accurate understanding of the mechanical behavior of the Silicone 40A resin used in the soft robot, and to enable more realistic structural simulations using nonlinear material models, a uniaxial tensile test was conducted.

A dog-bone specimen was printed using the same material and post-processing conditions as the soft robot. The specimen was made according to ASTM D638 and is shown in Figure 4.5. The test was performed using a Kezhun universal testing machine equipped with a Transcell 500 N load cell. The test speed was set to 15 mm/min.

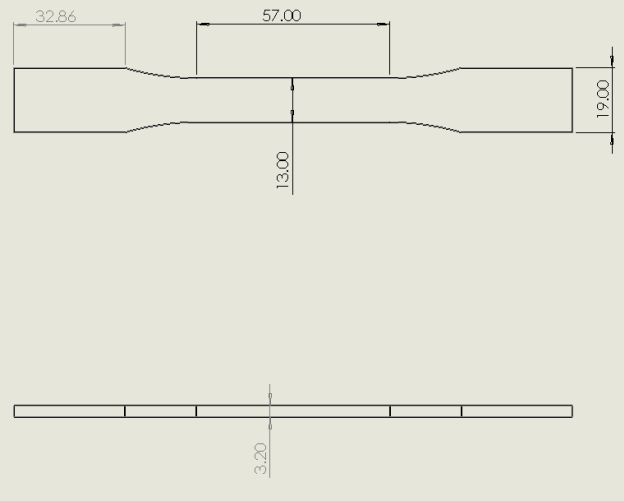


Figure 4.5: Dimensions of the dog-bone specimen used for tensile testing.

Using the machine grip displacement to calculate strain would yield incorrect results, as the displacement at the grips does not represent the actual deformation occurring at the center of the specimen. In soft materials like silicone, strain is highly non-uniform, and accurate measurement requires capturing the local deformation at the region of interest. To address this, Digital Image Correlation (DIC) was employed to measure local strain. DIC is a non-contact optical method that tracks the displacement of a speckle pattern applied to the specimen surface. By comparing images captured during deformation, the method computes full-field strain distributions with high spatial resolution. To prepare the specimen for DIC, a random speckle pattern was applied by spraying white paint dots over a dark base layer. The pattern was concentrated near the center of the specimen to ensure accurate tracking in the region of interest. The tensile test setup is shown in Figure 4.6. Figure 4.7 shows the axial strain distribution on the specimen at a specific load level. The DIC analysis provided accurate strain data at the center of the specimen, which was used to generate engineering stress-strain data and convert it to true stress-strain form using the following equations:

$$\sigma_{\text{true}} = \sigma_{\text{eng}}\big(1 + \varepsilon_{\text{eng}}\big) \tag{4.1}$$

$$\varepsilon_{\text{true}} = \ln\big(1 + \varepsilon_{\text{eng}}\big) \tag{4.2}$$

where σ_{eng} is obtained by dividing the load to the initial cross-sectional area of the specimen and ϵ_{eng} is obtained directly from the DIC analysis. The resulting stress-strain curves are presented in Figure 4.8. The data includes 53 points up to an engineering strain of approximately 0.64 mm/mm. These results will be used to define a hyperelastic material model for the Silicone 40A resin in the simulations presented in Chapter 5.

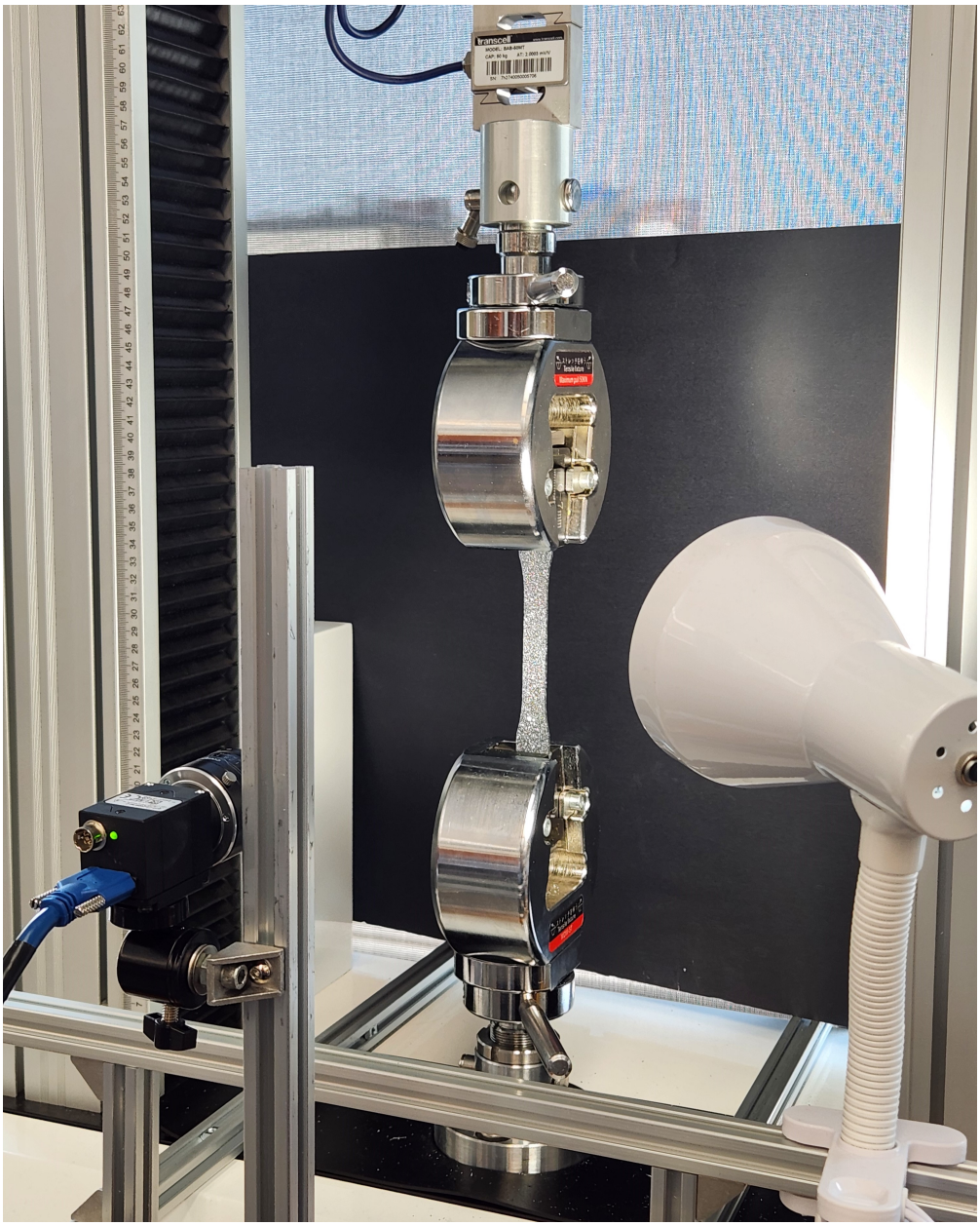


Figure 4.6: Tensile test setup with DIC system for strain measurement.

4.1.3 Injection Molding

Injection molding is a well-established technique for fabricating soft prototypes, including catheter-based systems and soft robotic platforms. Its ability to produce high-fidelity, repeatable, and biocompatible components makes it particularly suitable for medical applications [54]. Compared to additive manufacturing, injection molding offers a cost-effective

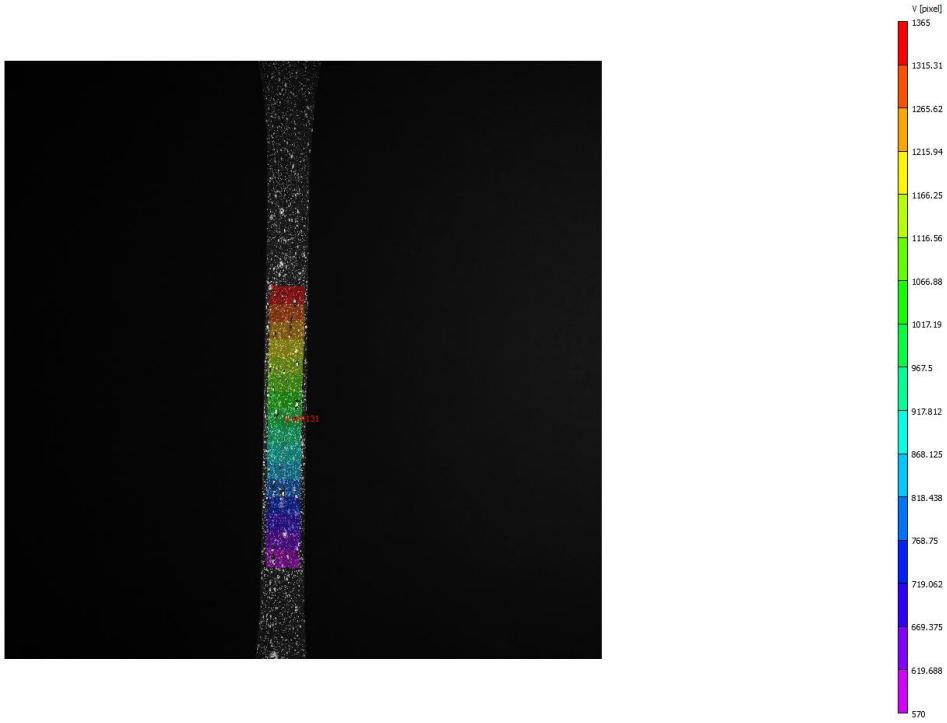


Figure 4.7: Vertical strain distribution on the specimen at F = 35 N.

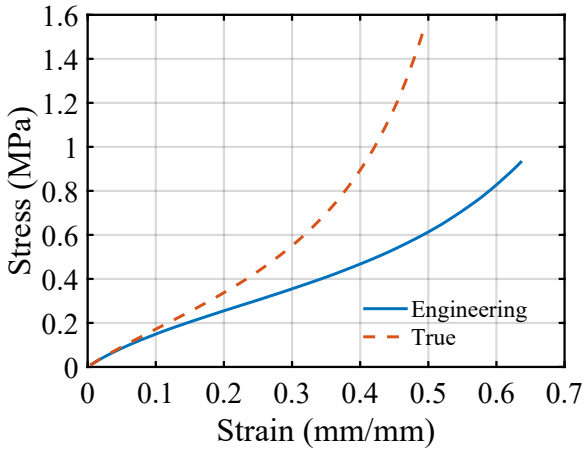


Figure 4.8: Engineering and true stress-strain curves obtained from the tensile test.

solution for large-scale production of complex elastomeric parts with consistent mechanical properties. It also supports a wide range of silicone formulations, enabling precise control

over mechanical characteristics such as hardness, elongation, tear strength, and biocompatibility.

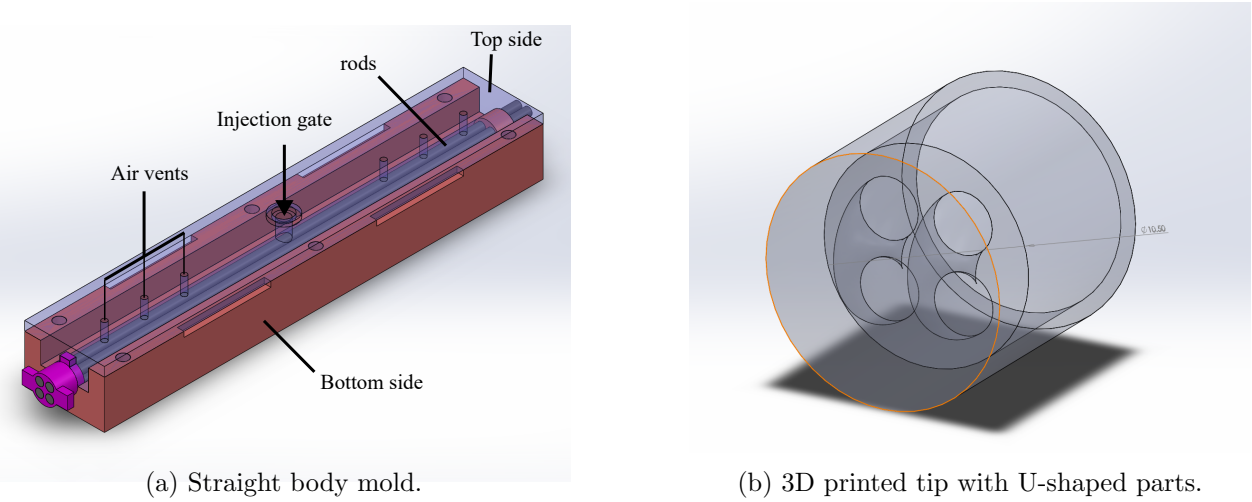
In this study, injection molding was explored as an alternative to 3D printing to evaluate its feasibility for scalable production. It enables the use of high-performance silicones that are not compatible with 3D printing and offers faster production cycles with lower per-part costs at scale.

Mold Design

The geometry of the soft robot presented challenges for conventional single-mold casting, particularly due to the U-shaped internal channels at the tip. To address this, the robot was divided into two components: a straight main body and a curved U-shaped tip. The main body was fabricated using a custom-designed mold, while the tip was produced via 3D printing.

The mold for the straight body (Figure 4.9a)section was designed with modular features to facilitate assembly and demolding. It consists of three parts: an upper half, a lower half, and alignment holes for inserting rods to form the internal channels. The injection inlet was positioned at the top center, surrounded by small air vents to allow air escape and minimize bubble formation.

For the tip component (Figure 4.9b), due to its small dimensions, a short extension was added with an internal diameter matching that of the main shaft. This design acts as a socket, allowing the main body to be inserted and bonded securely. This approach avoids gluing the cross-sections directly, which could risk obstructing the internal channels.



(a) Straight body mold.

(b) 3D printed tip with U-shaped parts.

Figure 4.9: CAD models of the mold components for silicone casting.

Mold Fabrication

The mold parts were fabricated using Formlabs Clear V4.1 resin, a transparent photopolymer that allows visual inspection during the molding process. This transparency facilitates monitoring of silicone flow and detection of air trapped inside the chamber. Clamps were used during assembly to ensure tight clearances and prevent leakage. Precision 3 mm diameter drill rods ($\pm$7.6 µm tolerance) were used to form the internal channels. Figure 4.10 shows the printed mold parts used in this process.

Casting Process

Smooth-On Mold Star 40T was selected for casting due to its Shore hardness matching that of the Formlabs Silicone. It is a transparent elastomer with a pot life of 20 minutes and cures at room temperature within three hours.

Prior to casting, a release agent (cooking oil) was applied to the mold's internal surfaces. Parts A and B of the silicone elastomer were mixed in a 1:1 weight ratio. To eliminate air bubbles introduced during mixing, the mixture was placed in a vacuum chamber for 16

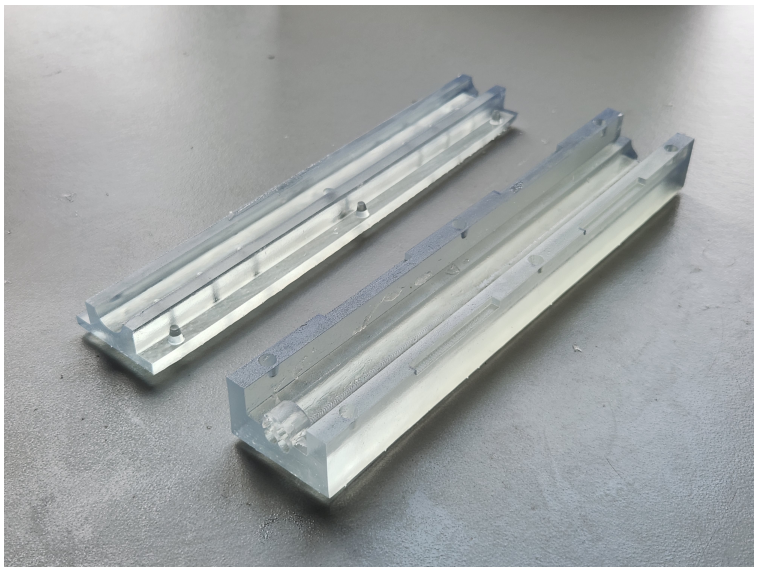


Figure 4.10: 3D-printed mold components used for silicone injection molding.

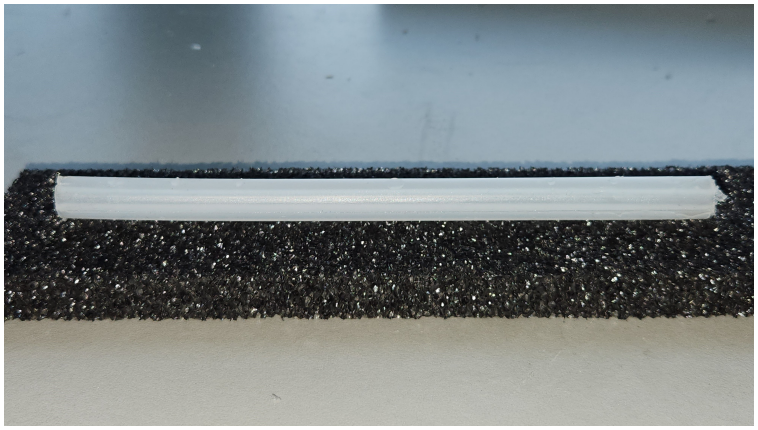


Figure 4.11: Final soft robotic component fabricated from silicone elastomer.

minutes. Additionally, the top internal surface of the mold was heated with a heat gun to further reduce the risk of air bubbles formation. The de-aired mixture was then transferred into a syringe and slowly injected into the mold.

After curing, the top half of the mold was removed, and the rods were manually extracted using compressed air. Figure 4.11 illustrates the silicone-casted component following demolding.



Figure 4.12: Final configuration of the soft robot after component fabrication and bonding

Final Assembly

After fabrication, the two components were bonded using Smooth-On Sil-Poxy, a silicone-specific adhesive. This ensured a continuous, sealed internal channel suitable for fluidic actuation. The final assembled soft robot is shown in Figure 4.12. Table 4.1 summarizes the mechanical properties of the elastomers and adhesive used.

Table 4.1: Mechanical properties of Sil-Poxy and casted elastomers compared to 3D-printed Silicone 40A resin.

Product Name	**Shore Hardness**	**Tensile Strength (psi)**	**100% Modulus (psi)**	**Elongation at Break (%)**
Formlabs Silicone	40A	725	145	230
Mold Star 40T	40A	656	108	500
Sil-Poxy Adhesive	40A	750	-	750

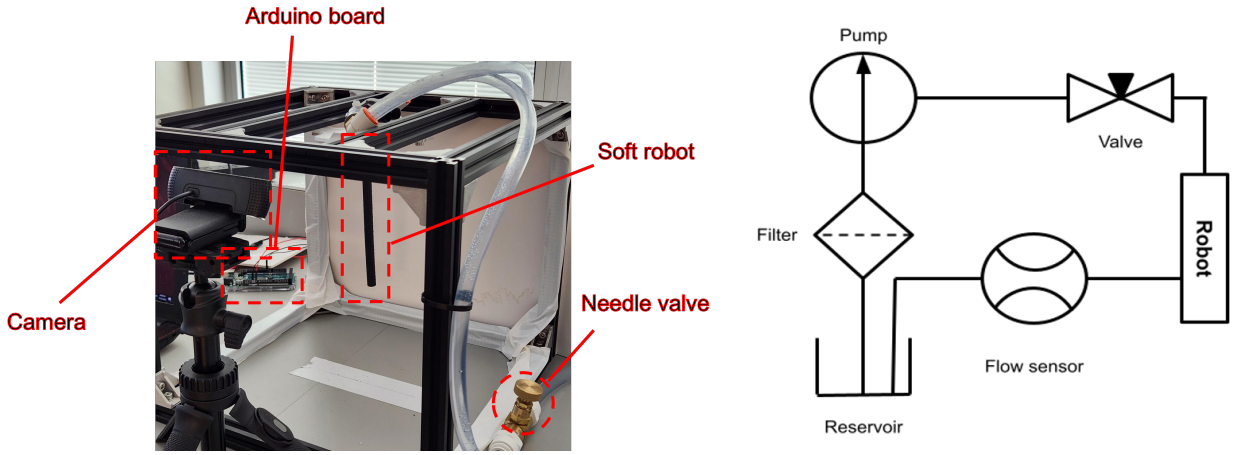


Figure 4.13: Test setup used to evaluate the deformation behavior of the soft robot.

4.2 Test Setup and Methodology

To evaluate the deformation behavior of the soft robot, a custom test rig was developed. The setup included a modular frame constructed from T-slotted aluminum bars, a custom-designed holder to clamp the robot vertically at its base, a flow loop, and an electronic data acquisition circuit. This system was designed to monitor and record both the flow rate and lateral deflection of the robot using dedicated sensors. An overview of the complete test setup is shown in Figure 4.13.

4.2.1 Equipment and Components

The flow loop was designed based on the operating conditions predicted by CFD simulations, with additional consideration for head losses introduced by tubing and fittings. A 5.5 GPM, 55 PSI water pump was selected as the primary flow source. To enable precise control of the flow rate, a needle valve was installed downstream of the pump. A water filter was also integrated to prevent particulates from entering the robot and potentially obstructing internal channels or damaging the pump.

Flow rate was measured using an OMEGA BV2000TRN050B turbine flow sensor, which operates based on the principle of a rotating turbine whose angular velocity correlates directly with fluid velocity. The sensor accommodates flow rates up to 5 L/min, is rated for pressures up to 25 bar, and outputs a frequency signal via the Hall effect. The sensor provides an accuracy of 3% of the reading and supports installation in any orientation.

The electronic control and data acquisition system was built around the Arduino MEGA 2560 Rev 3 microcontroller. It handled sensor data acquisition, signal conversion into flow rate values, and data transmission to a computer for logging and analysis. A variable DC power supply set to 9 V independently powered both the Arduino and the sensor. A custom Arduino script was developed to interface with the sensor. It reads the frequency signal, converts it into real-time flow rate values, and transmits the data via serial communication for logging purposes. The complete code is provided in Appendix A.1.

4.2.2 Flow Sensor Validation

To validate the accuracy of the flow sensor, a calibration test was conducted. The sensor was connected to the flow loop without the robot, and flow rate data were collected over 5-minute periods at various needle valve positions. Simultaneously, the volume of water discharged was measured manually over known time intervals to calculate the average flow rate. These values were then compared to the sensor readings and are shown in Table 4.2.

Seen from the table, for flow rates near the upper and lower bounds of the operating range of the sensor, the maximum relative error is 7.1%. For mid-range flow rates, the average error was 3.2%, confirming reliability of the sensor for the use in these experiments.

Table 4.2: Validation results of the Omega flow sensor.

Test #	ΔT **(s)**	ΔV **(mL)**	Q_{calc} **(L/min)** $= 0.06 \times \Delta V/\Delta T$	$Q_{\textbf{sensor}}$ **(L/min)**	**Error (%)**
1	10	840	5.04	4.78	5.2
2	30	1730	3.46	3.50	1.2
3	30	1400	2.80	2.83	1.1
4	30	600	1.20	1.22	1.7
5	60	480	0.48	0.45	7.1

4.2.3 Image-Based Deformation Tracking

To capture the deformation, a Logitech C920 camera was mounted on a stable tripod, aligned parallel to the deformation plane, and positioned to ensure the entire deformation range remained within the frame. Prior to image processing, camera calibration was conducted to determine intrinsic, extrinsic, and distortion parameters. A checkerboard calibration pattern was printed and placed near the robot, aligned with its XY plane. Using 22 images of the checkerboard at various positions within the frame, calibration was performed, yielding a mean reprojection error of 0.65 pixels. This error quantifies the average pixel distance between actual and projected checkerboard corner positions, as shown in Figure 4.14.

Image processing was carried out in MATLAB to extract the centerline of the robot over time. The workflow, illustrated in Figure 4.15, and detailed in Appendix A.2, involved importing each raw image frame, undistorting it using the calibration parameters, and applying color segmentation to isolate the robot body. The segmented robot was rendered as white pixels against a black background, from which the centerline was extracted and overlaid for subsequent analysis.

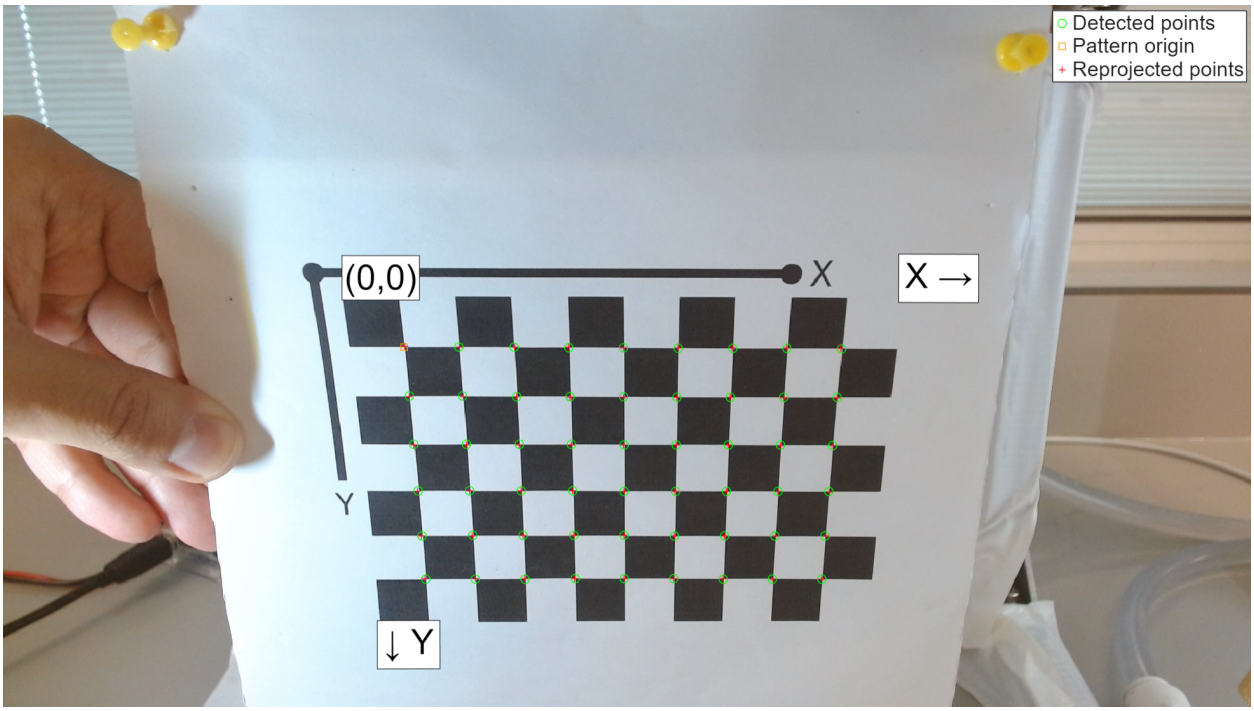


Figure 4.14: Checkerboard image for camera calibration. Axes are for image processing only and are in pixels, not system coordinates.

4.2.4 Data Acquisition Protocol

To accurately record the deformed shape of the robot as the flow rate is varied, a custom data acquisition software was developed to simultaneously capture data from both the camera and the microcontroller. The system recorded high-definition (1080p) images at an optimal rate of 30 frames per second, providing sufficient temporal and spatial resolution for analysis. Synchronization between video frames and sensor data was achieved using the computer's global clock, with both data streams sampled at the same nominal frequency. In practice, the effective recording rate was approximately 20 Hz, which remained adequate for the experimental requirements.

Each test sequence included a stabilization period of 30–60 seconds following any change in valve state, allowing the robot to reach a steady deformation condition. This ensured

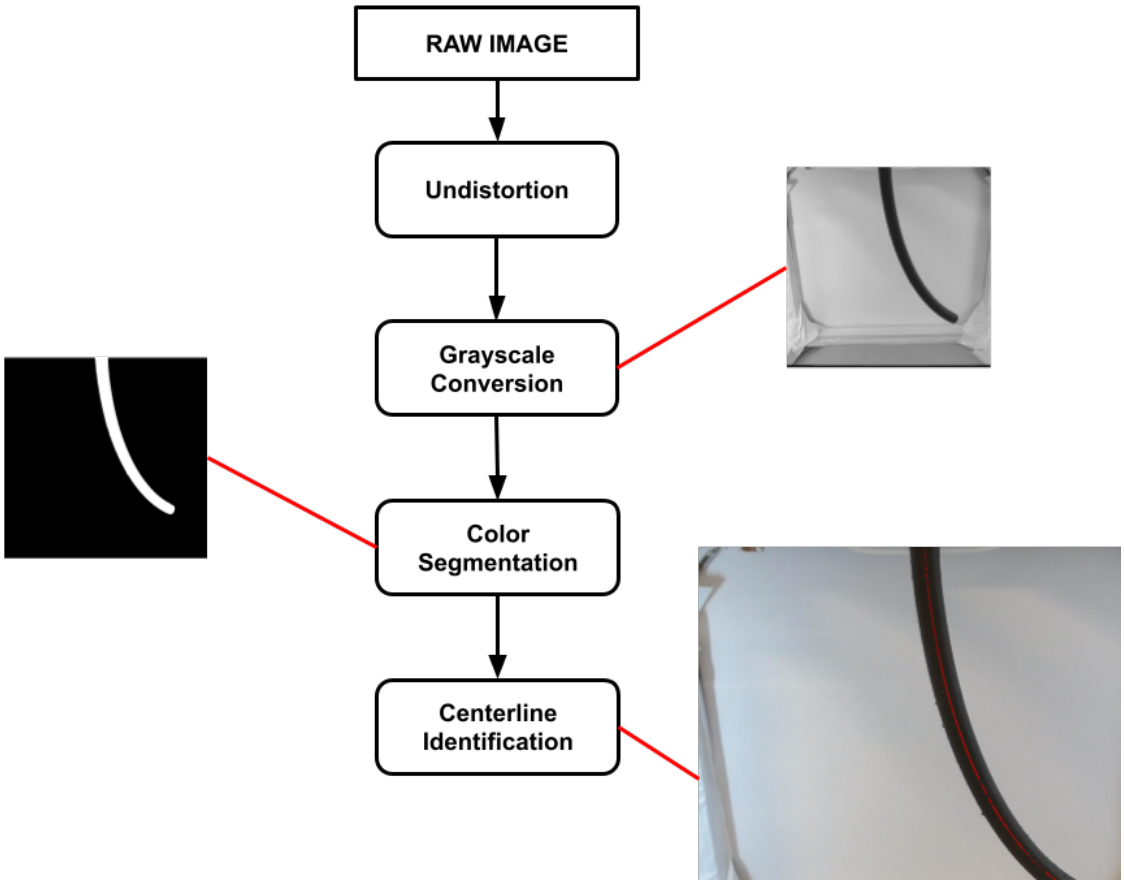


Figure 4.15: Image processing workflow used to extract the centerline of the robot from video frames.

that the captured images accurately represented the corresponding steady-state flow rate. For each valve state, the final 200 samples were used to compute the average flow rate, and the median frame within this window was selected to represent the deformation of the robot under the given flow condition.

4.3 Experimental Results

Using the MATLAB algorithm provided in Appendix A.2, the centerline of the soft robot was extracted for each tested flow rate. The script initially identified the centerline in pixel coordinates and then converted it to Cartesian coordinates in the XY plane, under the assumption of planar deformation. This process enabled accurate reconstruction of the robot's shape across varying flow rates.

Figure 4.16 presents a composite image in which multiple deformation states are superimposed to illustrate the full range of motion observed. Figures 4.17 and 4.18 display the extracted centerlines from nine separate tests, covering the velocity range during both the loading phase (forward or increasing flow velocity) and the unloading phase (backward or decreasing flow velocity). Experimental observations confirm that the soft robot can be reliably actuated in response to changes in flow rate.

The maximum tip deflection recorded across the tests was 93.84 mm on average, which corresponds to more than 58% of the robot's initial length. At the highest flow rate achieved, the tip angle remained below 90°. Furthermore, at the maximum tested flow velocity, the robot exhibited an average axial extension of 16.57%, demonstrating that extension plays a significant role in the overall deformation behavior, in addition to pure bending.

Figure 4.19 presents the dimensionless tip deflection as a function of dimensionless flow velocity, with a clear distinction between increasing and decreasing flow conditions. Although the unloading path generally produced lower tip deflections, this pattern was not consistent across all experiments. This suggests that during backward bending the robot dissipates more stored strain energy, resulting in reduced deflection during unloading. The effect is likely due to material nonlinearities or changes in the internal flow conditions within the channel during backward motion. However, additional testing is required to more conclusively determine the underlying cause of this behavior.

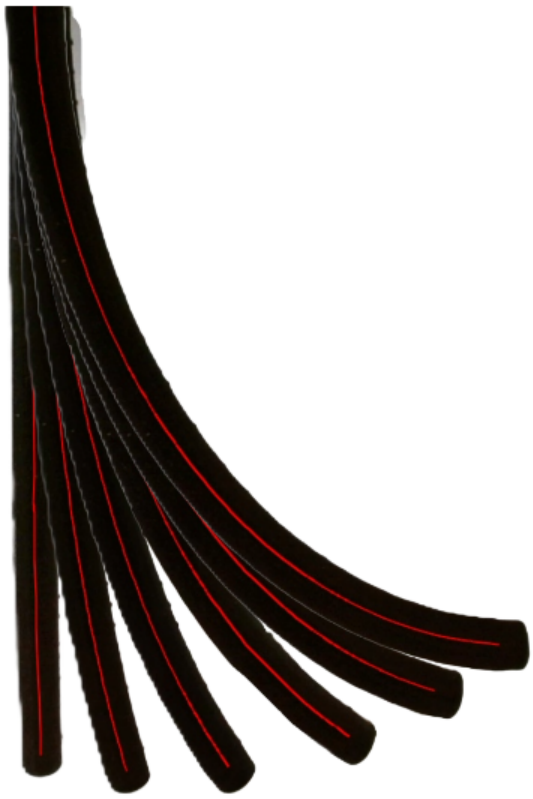
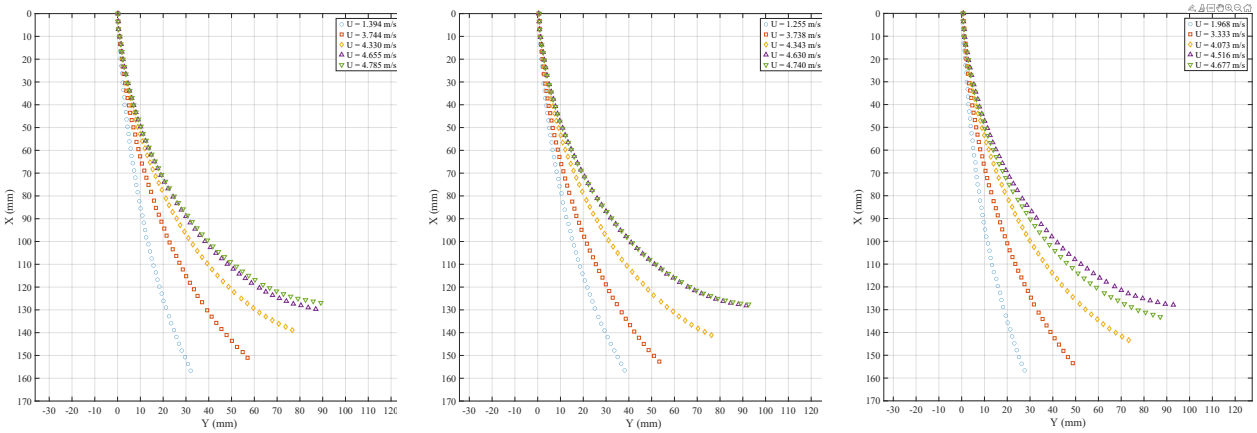


Figure 4.16: Superimposed image of robot deformation.

For Test 3, the slightly lower deflection observed at $U = 4.677$ m/s compared to $U = 4.516$ m/s may be attributed to a minor error in flow rate measurement, given the velocity difference of 0.161 m/s.

Regarding the casted robot, it failed to operate properly due to leakage between actuated and unactuated channels. This issue likely stemmed from a small gap forming between the body and tip components under high internal pressure. Tensile stresses may have caused slight displacement of the tip, and since glue was applied only around the outer surfaces (see Section 4.1.3), the internal interface remained vulnerable. This gap may have allowed pressurized fluid to escape into unactuated channels. Although resolving this issue is beyond

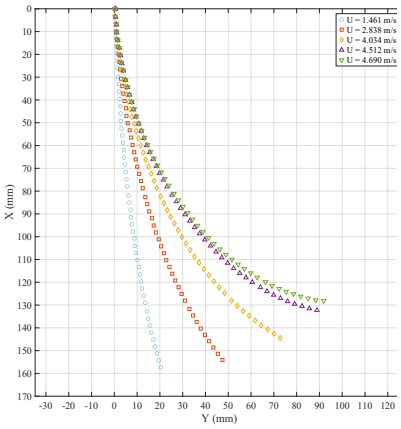
the scope of the current study, one potential solution could involve thermally bonding the body and tip components at their cross-sections under pressure, provided the materials are compatible with heat bonding.



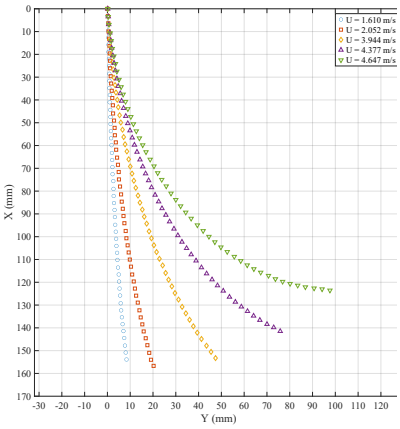
(a) Test 1

(b) Test 2

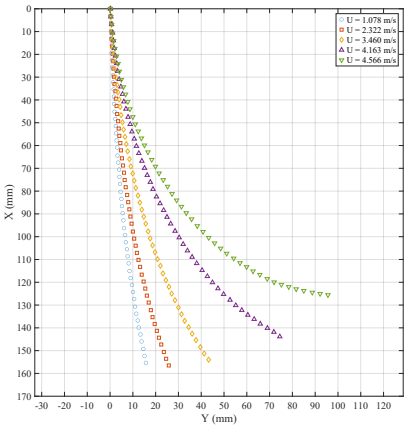
(c) Test 3



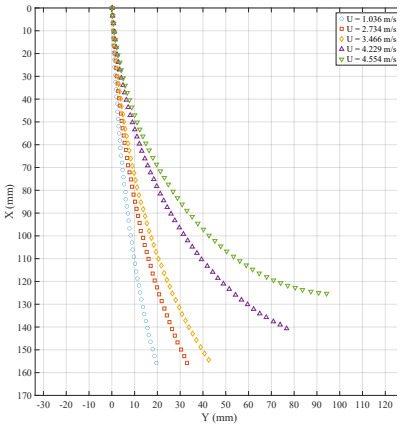
(d) Test 4



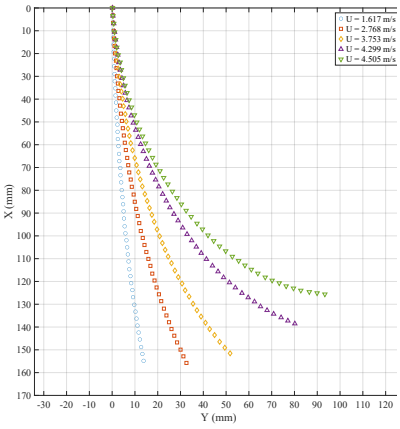
(e) Test 5



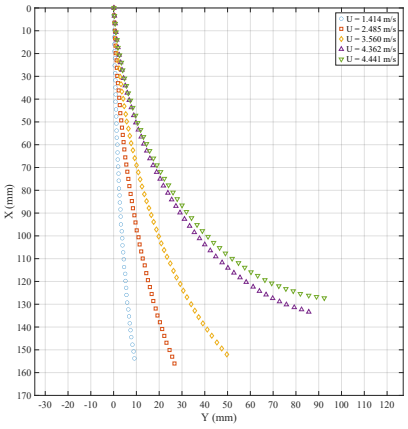
(f) Test 6



(g) Test 7

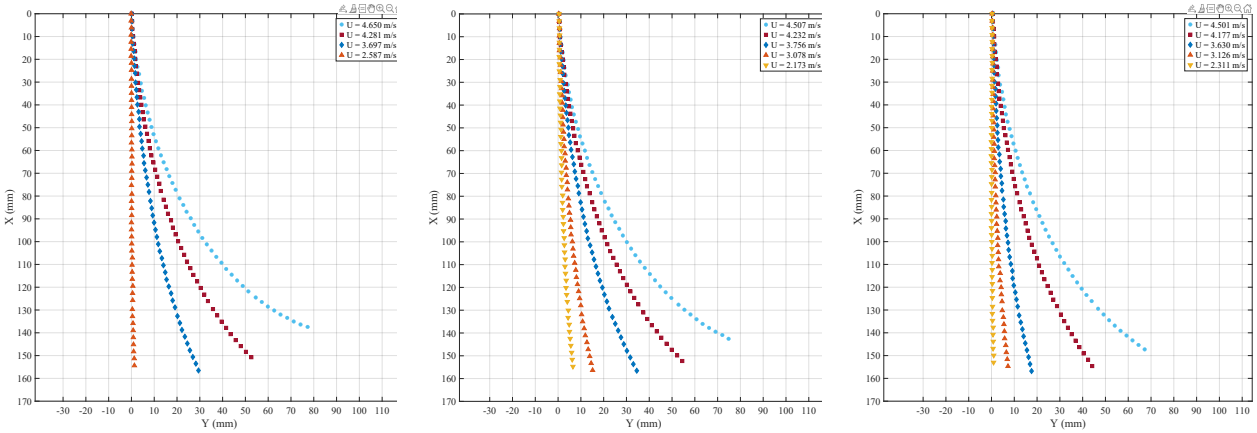


(h) Test 8



(i) Test 9

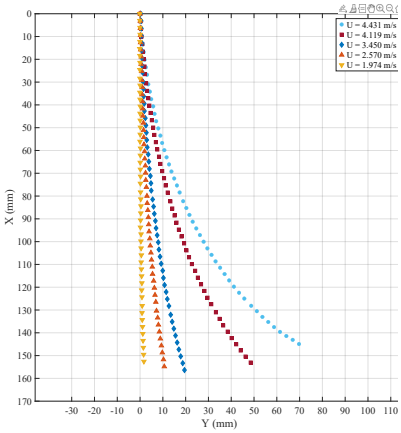
Figure 4.17: Extracted centerline of the soft robot in various flow velocities at the loading (forward) phase.



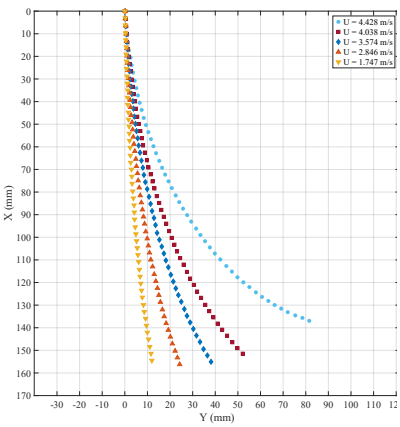
(a) Test 1

(b) Test 2

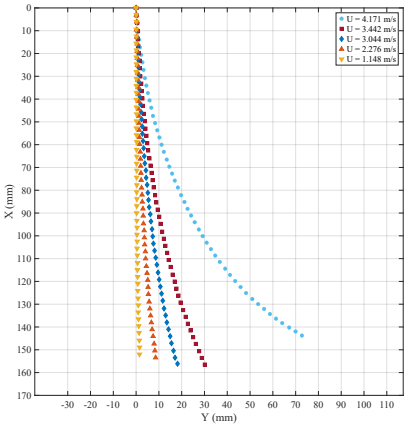
(c) Test 3



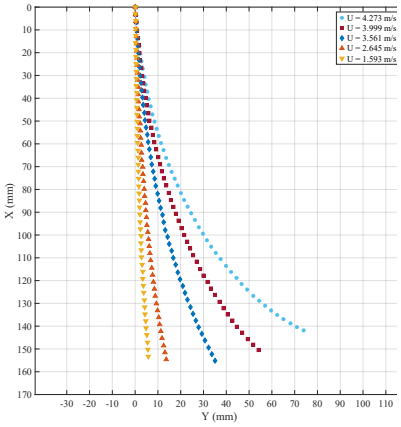
(d) Test 4



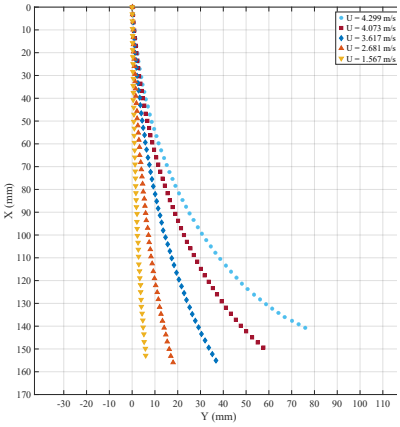
(e) Test 5



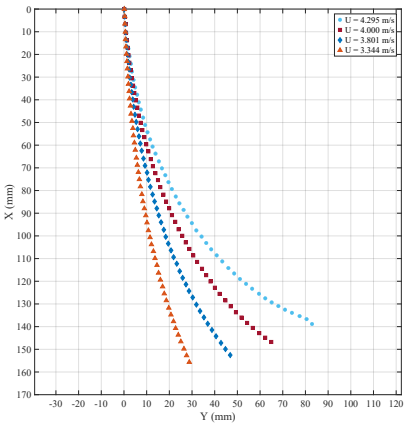
(f) Test 6



(g) Test 7

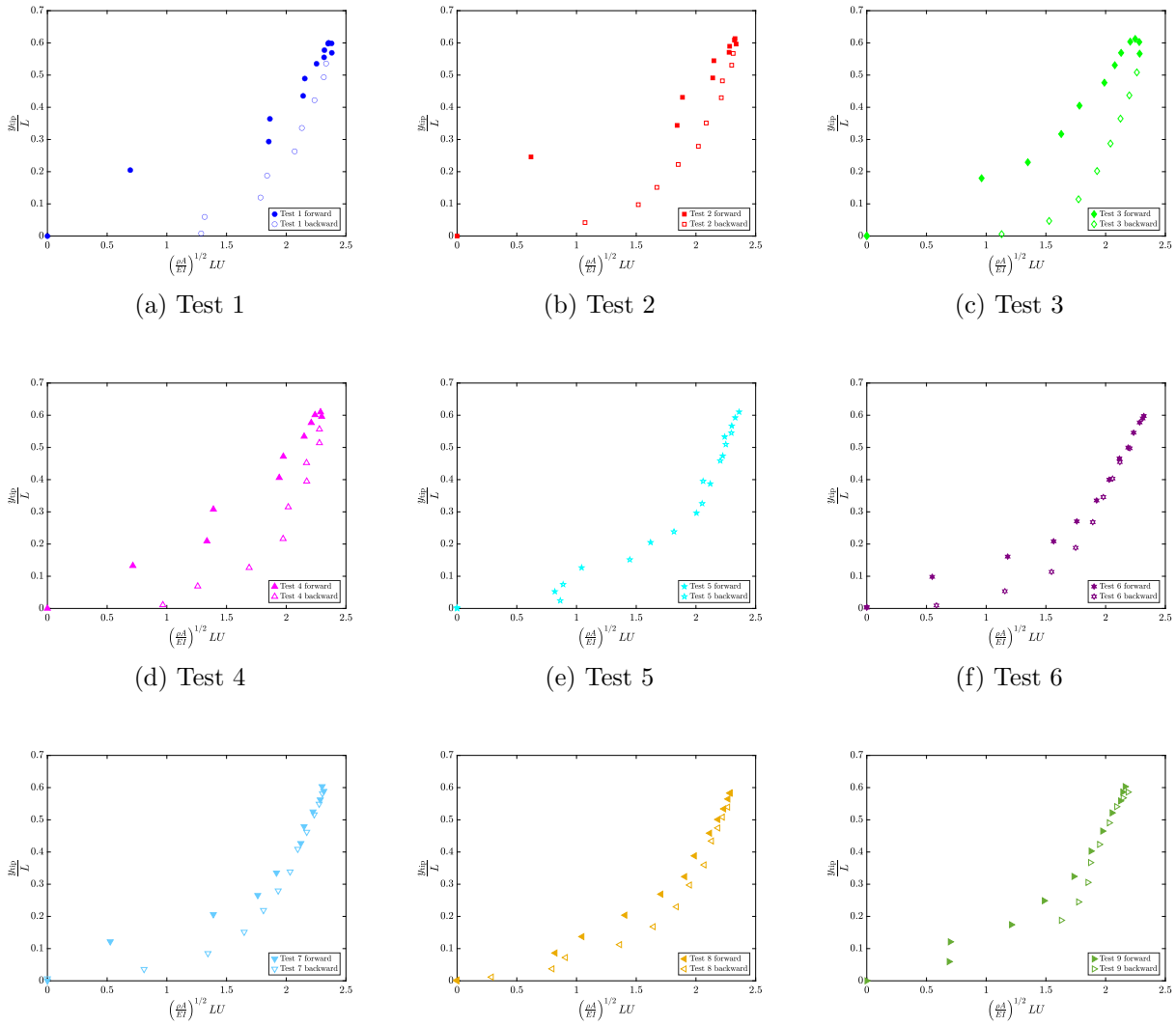


(h) Test 8



(i) Test 9

Figure 4.18: Extracted centerline of the soft robot in various flow velocities at the unloading (backward) phase.



(a) Test 1 (b) Test 2 (c) Test 3

(d) Test 4 (e) Test 5 (f) Test 6

(g) Test 7 (h) Test 8 (i) Test 9

Figure 4.19: Dimensionless tip deflection of the robot as a function of the dimensionless flow velocity. Filled symbols correspond to deflections while increasing the flow velocity, while hollow symbols indicate deflections while decreasing the flow velocity.

Chapter 5

Results Comparison and Model Evaluation

This chapter presents a comparative analysis of experimental observations, analytical predictions, and computational simulations. First, the effective bending stiffness (EI_{eff}) of the soft robot was evaluated using the analytical model developed in Chapter 2, and the resulting centerline shapes were compared with experimental profiles. Next, the limitations of the standard one-way FSI approach in ANSYS were examined, revealing its inability to capture the distributed normal follower load that significantly influences deformation. To address this, a modified one-way FSI approach was implemented, incorporating constant normal follower pressures along the channel wall. The predictions from this approach were then compared to experimental planar deformations, demonstrating improved accuracy at higher flow rates. Finally, key observations and discussions were provided to explain the influence of internal pressure distribution, material properties, and modeling assumptions on the robot's deformation behavior.

5.1 Analytical Model Validation

The following material and geometric properties were used to compute the results from the analytical model described in Chapter 2: water density $\rho = 997.62\ \mathrm{kg\,m^{-3}}$, structure density $\rho_r = 1152.8\ \mathrm{kg\,m^{-3}}$, and second moment of area $I = 278.764\ \mathrm{mm^4}$, based on the robot's circular cross-section. Using the computational model outlined in Chapter 3, the static inlet and outlet pressures corresponding to the tested flow velocities were applied to estimate the effective bending stiffness, EI_{eff}, of the centerlines during the loading phase (Figure 4.17); the variation of EI_{eff} with flow velocity is shown in Figure 5.1.

Stiffness values were obtained by fitting the theoretical model to the experimentally observed centerline shapes. Across experiments, a representative value $EI_{\mathrm{eff}} \approx 7 \times 10^{-4}\ \mathrm{N\,m^2}$ was consistently observed. This result closely matched an independent estimate computed using Young's modulus $E = 2.508$ MPa (determined from the initial slope of the stress–strain curve in Fig. 4.8) multiplied by the second moment of area, which yields $EI_{\mathrm{eff}} = 6.912 \times 10^{-4}\ \mathrm{N\,m^2}$.

Using this stiffness value, the experimentally extracted centerline shapes were compared with their theoretical counterparts (Figure 5.2). The tip position discrepancy was quantified in Table 5.1 using the following formulation:

$$Err \;=\; \frac{\sqrt{(x_{analytical} - x_{\mathrm{exp}})^2 + (y_{analytical} - y_{\mathrm{exp}})^2}}{\sqrt{x_{\mathrm{exp}}^2 + y_{\mathrm{exp}}^2}}\,, \tag{5.1}$$

where $x_{analytical}, y_{analytical}$ denote the analytical tip coordinates and $x_{\mathrm{exp}}, y_{\mathrm{exp}}$ denote the corresponding experimental values. While the analytical model captures the overall behavior of the soft robot, it underestimates the pronounced elongation observed at higher flow velocities, reflecting the limitations of the inextensibility assumption introduced in Chapter 2. In addition, assuming linear elasticity can introduce further error because the elastomer is

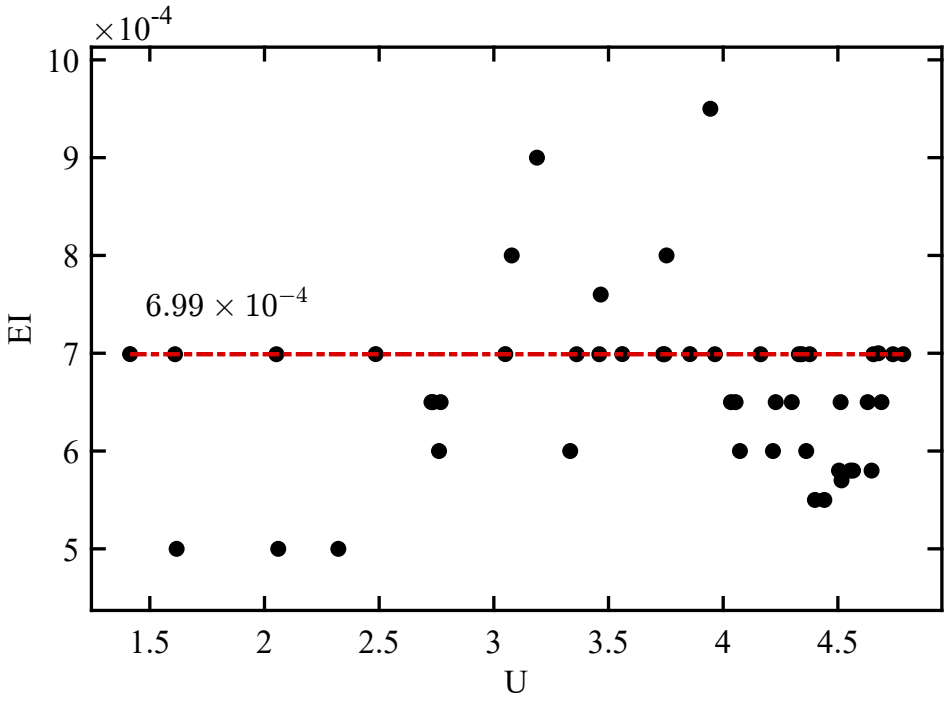


Figure 5.1: Variation of effective bending stiffness (EI_{eff}) with flow velocity. Stiffness values are derived by fitting the theoretical model to the experimentally observed centerline shape of the soft robot.

hyperelastic and exhibits nonlinear stress–strain behavior.

Despite these simplifications, the model performs well: the average tip position error across the tests is approximately 9%. In practice, particularly in applications such as minimally invasive surgery, where robots are routinely calibrated—the analytical model remains a powerful and practical basis for accurately controlling a flow-driven soft robot.

Equation 2.8 suggests the presence of a secondary moment component that could induce out-of-plane deformation. However, its magnitude depends on the pressure difference term $P_1 - P_2$, which was expected to be small. In contrast, the dominant in-plane bending moment arises from the follower tension, proportional to the pressure sum $P_1 + P_2$ (see Equation 2.7). Experimentally, deformation remained planar, indicating that out-of-plane effects were negligible or effectively suppressed.

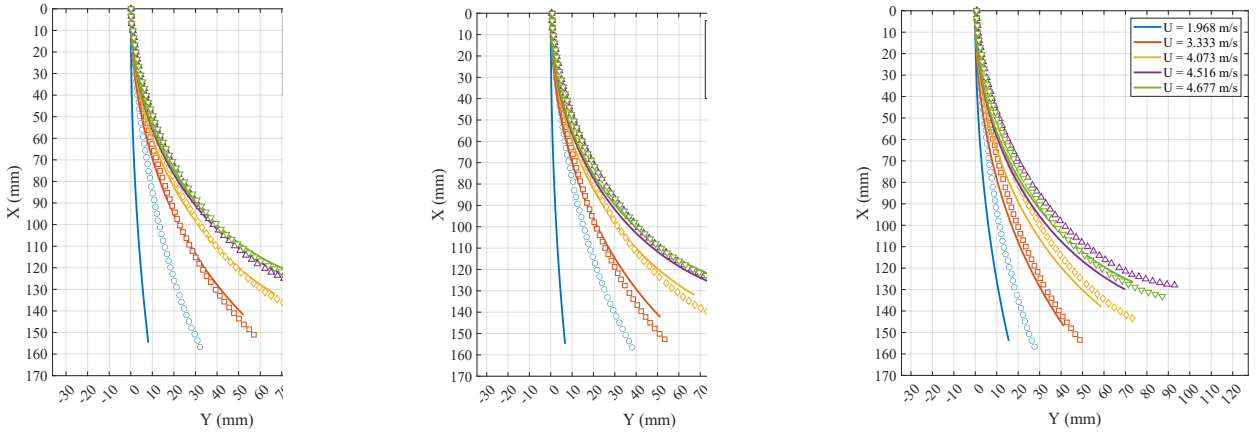
Table 5.1: Analytical vs. Experimental tip positions.

Test	$\boldsymbol{U}$ ($\mathrm{m\,s^{-1}}$)	$\boldsymbol{x_{\text{analytical}}}$ (mm)	$\boldsymbol{y_{\text{analytical}}}$ (mm)	$\boldsymbol{x_{\text{exp}}}$ (mm)	$\boldsymbol{y_{\text{exp}}}$ (mm)	**Err (%)**
	1.394	156.44	8.16	159.18	32.37	15.0
	3.744	144.45	52.37	154.32	57.63	6.8
Test 1	4.330	135.36	67.11	142.99	77.11	7.7
	4.655	128.47	75.26	133.52	87.37	8.2
	4.785	126.43	77.63	130.96	89.74	8.1
	1.255	158.89	6.32	161.11	36.84	18.5
	3.738	145.56	49.47	156.94	51.32	7.0
Test 2	4.343	135.56	64.47	145.00	73.95	8.2
	4.630	129.17	71.84	131.67	88.95	10.9
	4.740	126.39	74.74	131.39	89.74	9.9
	1.968	152.43	15.83	155.41	28.06	8.0
	3.333	145.68	41.67	152.16	49.44	6.3
Test 3	4.073	136.49	59.44	142.16	74.44	10.0
	4.516	128.92	70.83	126.76	94.44	15.0
	4.677	125.68	74.44	132.16	88.61	9.8
	1.468	156.02	8.65	159.02	20.54	7.6
	2.838	151.53	30.27	154.95	47.30	10.7
Test 4	4.034	138.57	57.03	144.53	72.43	10.2
	4.512	129.90	68.92	131.58	88.65	12.5
	4.690	125.62	73.24	127.33	91.62	11.8
	1.610	150.53	9.84	150.26	7.94	1.3
	2.052	150.00	15.78	153.16	19.91	3.4
Test 5	3.944	136.84	53.80	149.47	46.57	9.3
	4.377	129.47	64.68	138.42	74.65	8.5
	4.647	124.21	71.29	120.79	96.07	16.2
	1.078	153.13	4.08	153.88	15.41	7.3
	2.322	150.61	19.80	154.68	25.17	4.3
Test 6	3.460	143.44	42.24	152.11	42.73	5.5
	4.163	133.78	59.36	140.78	73.49	9.9
	4.566	125.09	69.65	121.81	94.04	16.0
	1.036	158.00	5.17	159.71	19.84	9.2
	2.734	154.57	28.32	159.14	32.91	4.0
Test 7	3.466	148.86	43.75	158.00	42.94	5.6
	4.229	138.29	62.94	144.00	77.22	9.4
	4.554	131.71	71.17	128.00	94.89	15.1
	1.617	156.87	10.79	157.76	13.68	1.9
	2.768	154.11	28.68	159.35	32.37	3.9
Test 8	3.753	146.02	49.21	155.80	51.05	6.1
	4.299	138.01	62.89	143.03	78.68	10.1
	4.505	133.57	68.68	130.14	91.84	14.7
	1.414	156.57	7.94	156.00	9.12	0.8
	2.485	154.57	23.82	158.57	27.06	3.2
Test 9	3.560	146.86	47.06	154.29	51.47	5.3
	4.362	134.57	67.94	135.43	89.12	13.1
	4.441	133.14	69.71	129.14	96.18	16.6

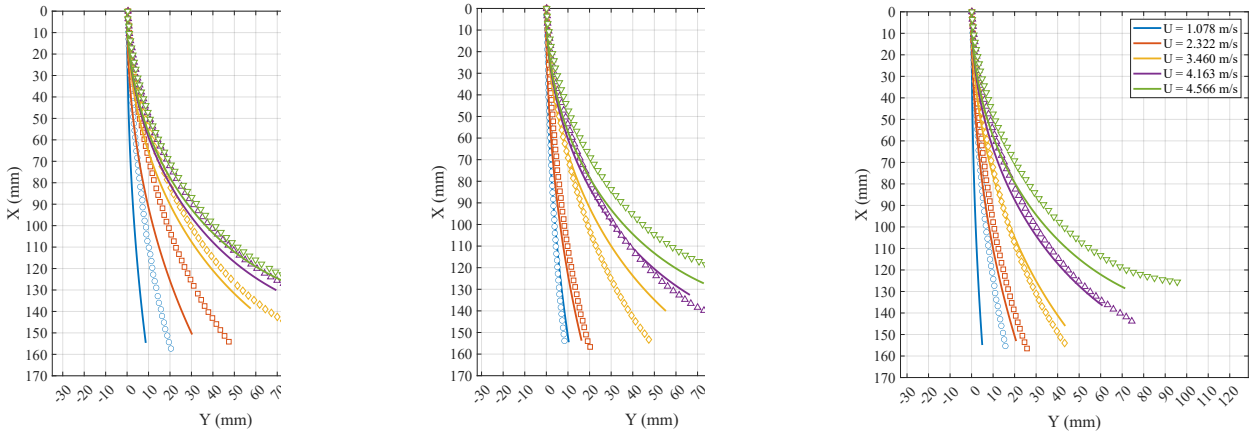
Table 5.2: CFD-derived inlet and outlet pressures for the soft robot at various flow velocities, corresponding to the experimental shapes shown in Figure 5.3.

No.	**U** ($\mathrm{m\,s^{-1}}$)	$\boldsymbol{P}_{\mathrm{in}}$ (kPa)	$\boldsymbol{P}_{\mathrm{out}}$ (kPa)	$\boldsymbol{P}_{\mathrm{sum}} = \boldsymbol{P}_{\mathrm{in}} + \boldsymbol{P}_{\mathrm{out}}$ (kPa)
1	1.414	26.606	21.311	47.917
2	2.052	62.687	52.717	115.404
3	2.485	86.357	72.191	158.547
4	3.050	122.867	102.869	225.736
5	3.855	175.680	145.233	320.913
6	4.163	214.101	179.377	393.478
7	4.740	269.544	225.494	495.038

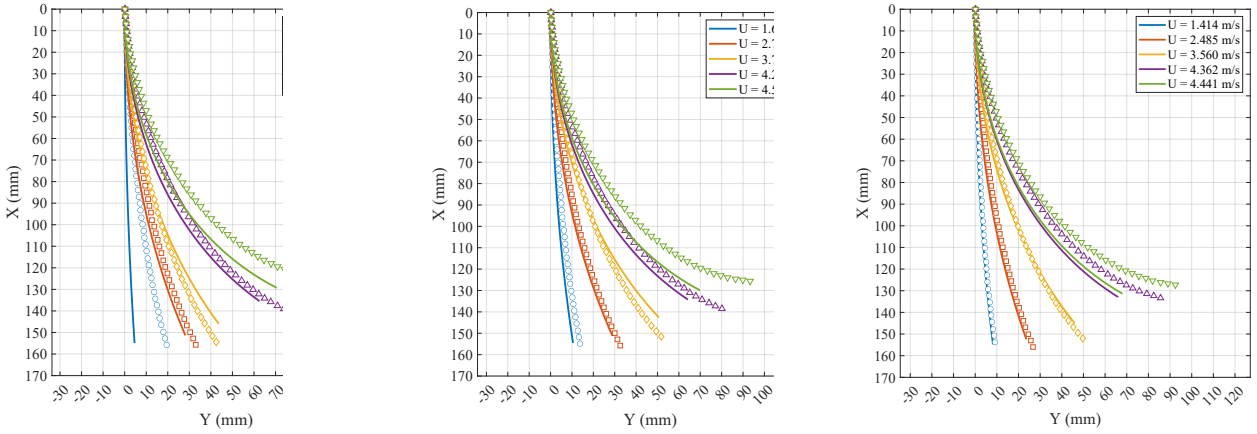
Figure 5.3 compares selected experimental deformation profiles that showed strong agreement with their corresponding theoretical predictions. The pressure values listed in Table 5.2 were used in the analytical model to generate the predicted shapes. The results reveal a clear correlation between flow rate and deformation magnitude: as the flow rate increases, the robot's curvature intensifies, culminating in a tip deflection angle of approximately 90° relative to its initial orientation.



(a) Test 1 (b) Test 2 (c) Test 3



(d) Test 4 (e) Test 5 (f) Test 6



(g) Test 7 (h) Test 8 (i) Test 9

Figure 5.2: Comparison of sample centerlines of the soft robot from tests 1 to 9 (markers) with the analytical model (solid lines) using an effective bending stiffness of $EI_{\text{eff}} = 6.912 \times 10^{-4}$ N $\cdot$ m^2.

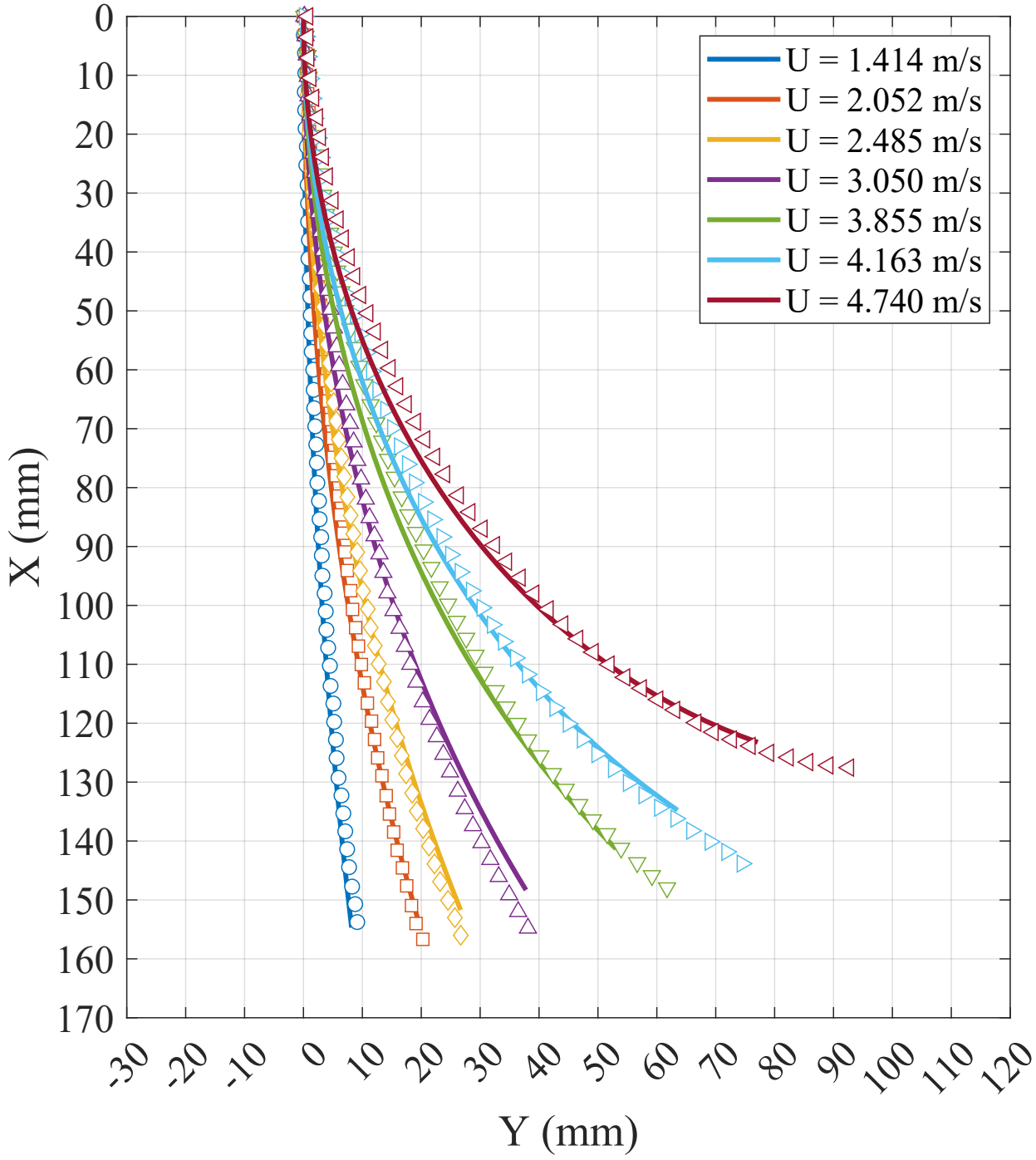


Figure 5.3: Selected experimental centerline shapes of the soft robot from tests 1 to 9 (markers), showing strong agreement with the analytical model (solid lines) using an effective stiffness of $EI_{\text{eff}} = 6.912 \times 10^{-4}\ \text{N} \cdot \text{m}^2$.

5.2 Computational Model Validation

For the one-way FSI approach, the experimental centerline plots in Figure 4.17 highlight the significant role of the distributed normal follower load, as predicted by the analytical model. In contrast, the standard one-way FSI method primarily accounts for the tensile follower force acting at the tip, while neglecting the distributed normal load along the body. This omission results in a straighter deflection profile. To investigate this effect, the analytical model was modified to include only the follower force term, excluding the normal load. The resulting shapes, shown in Figure 5.4, confirm that the standard one-way FSI approach in ANSYS fails to accurately capture the robot's behavior under realistic conditions. However, a modified FSI approach—where normal follower pressures were applied uniformly along the channel wall—successfully predicted the deformation. Figure 5.5 compares the deformation of the soft robot under various flow velocities, as observed experimentally and simulated using the modified FSI method. These centerline profiles correspond to those previously analyzed in Figure 5.3. The results indicate that this simulation approach effectively captures the material behavior and replicates the robot's deflection under flow-induced actuation. While the computational model performs inadequately at lower velocities, it provides accurate predictions at higher flow rates by incorporating material extension effects. Minor distortions observed at the free end of the simulated profiles are attributed to channel expansion, which alters the mid-axis deformation, as illustrated in Figure 5.6.

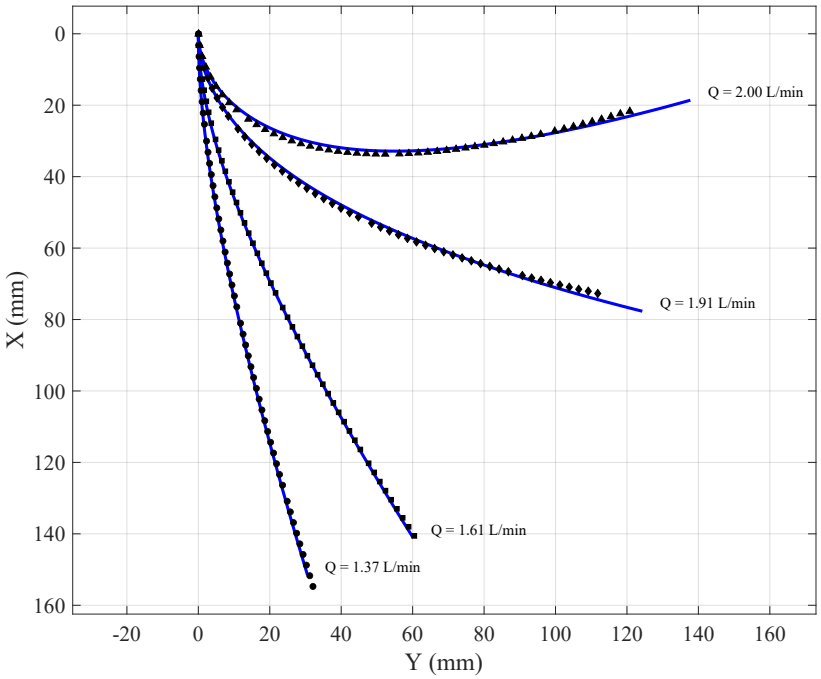


Figure 5.4: Comparison of robot centerline profiles obtained from the standard one-way FSI simulations (markers) and the modified theoretical model (solid lines) incorporating only a follower tensile force at the tip. Results are shown for flow rates of $Q = 1.37$, 1.61, 1.91, and 2.00 L/min. The close agreement between the model and simulation highlights the dominant role of the tip follower force in shaping the deformation under standard one-way FSI approach in ANSYS, which is insufficient for modeling the main case of this study.

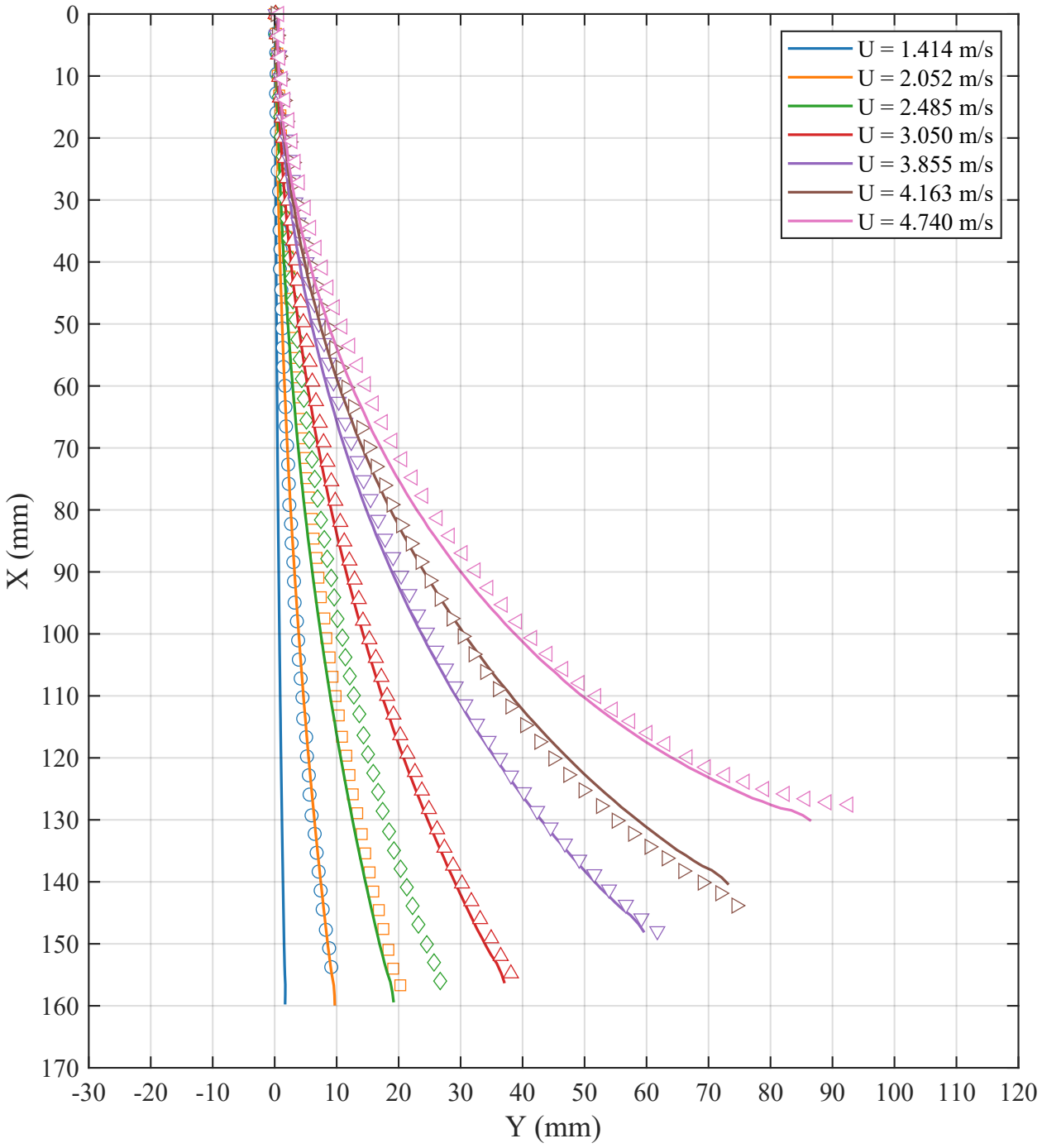


Figure 5.5: Comparison between experimentally measured robot deformation (markers) and modified FSI simulation results (solid lines) under identical flow velocities in Figure 5.3. The simulation effectively captures the robot's bending behavior, demonstrating strong agreement with experimental observations.

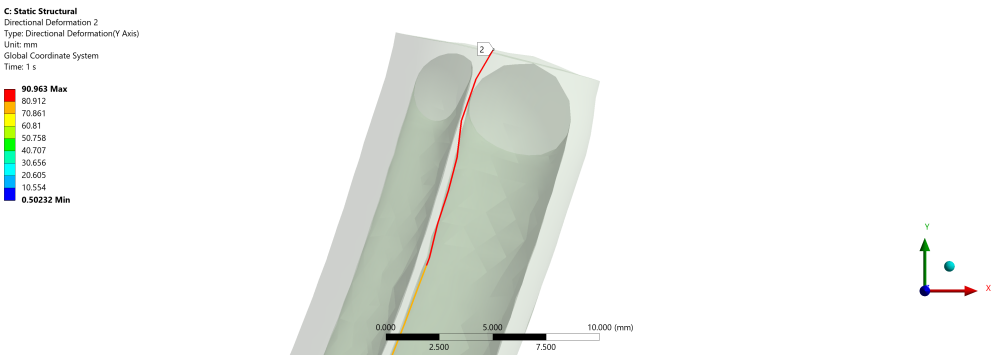


Figure 5.6: Mid-axis distortion in the FEA simulation caused by channel expansion under constant pressure loading. The deformation reflects the influence of internal pressure on the channel geometry.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

This thesis explored flow-driven actuation of soft robots, with a particular focus on applications in surgical operations and critical safety scenarios. The research was structured across three main chapters, each contributing to a comprehensive understanding of the topic.

Chapter 2: Analytical Modeling

- We began by analyzing the flow behavior within an embedded channel of a soft cylindrical actuator, focusing on the control volume at the tip of the robot, especially at the U-bend of the channel.
- Using control volume analysis and the Euler-Bernoulli beam theory, we derived a geometrically exact equation for the static lateral deflection of the soft robot under flow actuation.
- The equation was then made dimensionless to isolate and investigate the core driving

factors, encompassing both structural characteristics and flow properties.

- The theoretical model demonstrated the feasibility of achieving tip deflections exceeding 90°, validating the concept of flow-driven actuation.

Chapter 3: Computational Modeling

- A 3D CAD model was created for the flow-driven soft robot prototype containing two U-shaped channels embedded parallel against each other.
- CFD and FEA validation studies were performed to investigate the feasibility of using one-way FSI modeling for the flow-driven soft robot.
- Standard one-way FSI approach was found inadequate for capturing the true deflection behavior due to its inability to properly model the fluid pressure distribution effects.
- As an alternative, modifications were implemented in the FSI simulations by directly applying a constant pressure on the channel walls.
- Material properties obtained from DIC tensile tests were integrated into the simulation settings for improved accuracy.

Chapter 4: Experimentation

- Prototype manufacturing was explored using additive manufacturing and injection molding techniques with soft silicone elastomers.
- DIC tensile tests were conducted on silicone specimens to characterize material properties for further analysis.
- A test rig was assembled, incorporating:

  - A flow loop with flexible piping, fittings, a pump, and a needle valve for flow regulation.
  - A flow sensor for measuring flow rate.
  - A data acquisition system comprising an Arduino microcontroller, a camera, and custom software to capture robot deflection and synchronize it with flow sensor data.

- Image processing techniques were employed to extract the mid-axis deflection profile of the robot for nine tests.

- The soft robot deformed as expected to the flow velocity, bending on average 56% of its initial length at the maximum flow rate.

- A notable observation was the change in the deflection robot during loading (forward) and unloading (backward) phases, indicating strain energy dissipation, which requires further studies.

Chapter 5: Results Comparison and Model Evaluation

- Using the theoretical model, the effective stiffness of each shape of the soft robot was extracted.

- The theoretical model was compared to the experimental results using a constant EI_{eff} assumption. The model matched the centerline deflections, although it did not account for the longitudinal extension observed in practice. The tip position was predicted using the model, with average 9% error compared to the experiment.

- The computational approach, however, successfully validated robot deflection, including longitudinal extension, except at low flow velocities where discrepancies remained.

6.2 Future Work

Building upon the findings of this thesis, several avenues for future research are proposed:

Theoretical Modeling

- Incorporate hyperelastic material models to better represent the nonlinear behavior of silicone and soft elastomers.
- Develop a comprehensive model that accounts for extensibility, enabling accurate prediction of tip deflection and overall deformation.

Experimental Enhancements

- Improve the bonding technique for injection-molded samples, potentially using heat bonding combined with pressure to prevent tip component failure.
- Integrate pressure sensors into the test setup to monitor pressure losses in the flow loop, which are critical for understanding system behavior.
- Enhance image processing algorithms to achieve superior centerline shape extraction for more accurate deflection analysis.
- Replace the needle valve with a proportional valve to enable precise flow rate control, facilitating dynamic and transient behavior studies.

Simulation Advancements

- Implement two-way FSI simulations to accurately capture both static and dynamic responses of the soft robot under flow actuation. Although computationally intensive, this approach holds promise for high-fidelity modeling.

- Link simulation models with design optimization frameworks to explore various channel geometries and identify configurations that maximize performance for specific applications.

References

[1] C. C. Nguyen, M. T. Thai, T. T. Hoang, J. Davies, P. T. Phan, K. Zhu, L. Wu, M. A. Brodie, D. Tsai, Q. P. Ha *et al.*, "Development of a soft robotic catheter for vascular intervention surgery," *Sensors and Actuators A: Physical*, vol. 357, p. 114380, 2023.

[2] M. C. Yip and D. B. Camarillo, "Model-less hybrid position/force control: a minimalist approach for continuum manipulators in unknown, constrained environments," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 844–851, 2016.

[3] Y. Fu, H. Liu, W. Huang, S. Wang, and Z. Liang, "Steerable catheters in minimally invasive vascular surgery," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 5, no. 4, pp. 381–391, 2009.

[4] B.-S. Kang, C. S. Kothera, B. K. Woods, and N. M. Wereley, "Dynamic modeling of mckibben pneumatic artificial muscles for antagonistic actuation," in *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009, pp. 182–187.

[5] Z. Awada, Y. Haddab, and M. Gouttefarde, "Pneunet actuators design: Trade-offs between deformation, force, and resistance to buckling," *Sensors and Actuators A: Physical*, vol. 386, p. 116307, 2025.

[6] J. H. Argyris and S. Symeonidis, "Nonlinear finite element analysis of elastic systems

under nonconservative loading-natural formulation. part i. quasistatic problems," *Computer Methods in Applied Mechanics and Engineering*, vol. 26, no. 1, pp. 75–123, 1981.

[7] B. D. Texier and S. Dorbolo, "Deformations of an elastic pipe submitted to gravity and internal fluid flow," *Journal of Fluids and Structures*, vol. 55, pp. 364–371, 2015, publisher: Elsevier.

[8] M. Anwer and R. So, "Swirling turbulent flow through a curved pipe: Part i: Effect of swirl and bend curvature," *Experiments in Fluids*, vol. 14, no. 1, pp. 85–96, 1993.

[9] F. Stella, J. Hughes, D. Rus, and C. Della Santina, "Prescribing cartesian stiffness of soft robots by co-optimization of shape and segment-level stiffness," *Soft Robotics*, vol. 10, no. 4, pp. 701–712, 2023.

[10] K.-I. Park, C. K. Jeong, N. K. Kim, and K. J. Lee, "Stretchable piezoelectric nanocomposite generator," *Nano Convergence*, vol. 3, no. 1, p. 12, 2016.

[11] M. Roshanfar, A. Sayadi, J. Dargahi, and A. Hooshiar, "Stiffness Adaptation of a Hybrid Soft Surgical Robot for Improved Safety in Interventional Surgery," in *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. Glasgow, Scotland, United Kingdom: IEEE, Jul. 2022, pp. 4853–4859. [Online]. Available: https://ieeexplore.ieee.org/document/9871310/

[12] J. Fras and K. Althoefer, "Soft fiber-reinforced pneumatic actuator design and fabrication: Towards robust, soft robotic systems," in *Annual Conference Towards Autonomous Robotic Systems*. Springer, 2019, pp. 103–114.

[13] C. C. Nguyen, T. Teh, M. T. Thai, P. T. Phan, T. T. Hoang, H. Low, J. Davies,

E. Nicotra, N. H. Lovell, and T. N. Do, "Bidirectional soft robotic catheter for arrhythmia treatment," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 9579–9585.

[14] J. Rogatinsky, D. Recco, J. Feichtmeier, Y. Kang, N. Kneier, P. Hammer, E. O'Leary, D. Mah, D. Hoganson, N. V. Vasilyev *et al.*, "A multifunctional soft robot for cardiac interventions," *Science Advances*, vol. 9, no. 43, p. eadi5559, 2023.

[15] F. Janabi-Sharifi, A. Jalali, and I. D. Walker, "Cosserat rod-based dynamic modeling of tendon-driven continuum robots: A tutorial," *IEEE Access*, vol. 9, pp. 68 703–68 719, 2021.

[16] M. Russo, S. M. H. Sadati, X. Dong, A. Mohammad, I. D. Walker, C. Bergeles, K. Xu, and D. A. Axinte, "Continuum robots: An overview," *Advanced Intelligent Systems*, vol. 5, no. 5, p. 2200367, 2023.

[17] M. Manti, V. Cacucciolo, and M. Cianchetti, "Stiffening in soft robotics: A review of the state of the art," *IEEE Robotics & Automation Magazine*, vol. 23, no. 3, pp. 93–106, 2016.

[18] D. C. Rucker and R. J. Webster III, "Statics and dynamics of continuum robots with general tendon routing and external loading," *IEEE Transactions on Robotics*, vol. 27, no. 6, pp. 1033–1044, 2011.

[19] J. Wang and A. Chortos, "Control Strategies for Soft Robot Systems," *Advanced Intelligent Systems*, vol. 4, no. 5, p. 2100165, May 2022. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/aisy.202100165

[20] A. Chortos, J. Mao, J. Mueller, E. Hajiesmaili, J. A. Lewis, and D. R. Clarke, "Printing Reconfigurable Bundles of Dielectric Elastomer Fibers," *Advanced*

Functional Materials, vol. 31, no. 22, p. 2010643, May 2021. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/adfm.202010643

[21] L. Chang, Y. Liu, Q. Yang, L. Yu, J. Liu, Z. Zhu, P. Lu, Y. Wu, and Y. Hu, "Ionic Electroactive Polymers Used in Bionic Robots: A Review," *Journal of Bionic Engineering*, vol. 15, no. 5, pp. 765–782, Sep. 2018. [Online]. Available: http://link.springer.com/10.1007/s42235-018-0065-1

[22] J. Fras and K. Althoefer, "A fluidic actuator with an internal stiffening structure inspired by mammalian erectile tissue," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 662–668.

[23] A. De Greef, P. Lambert, and A. Delchambre, "Towards flexible medical instruments: Review of flexible fluidic actuators," *Precision engineering*, vol. 33, no. 4, pp. 311–321, 2009.

[24] N. El-Atab, R. B. Mishra, F. Al-Modaf, L. Joharji, A. A. Alsharif, H. Alamoudi, M. Diaz, N. Qaiser, and M. M. Hussain, "Soft Actuators for Soft Robotic Applications: A Review," *Advanced Intelligent Systems*, vol. 2, no. 10, p. 2000128, Oct. 2020. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/aisy.202000128

[25] H. Schulte, "The characteristics of the mckibben artificial muscle," *The application of external power in prosthetics and orthotics*, pp. 94–115, 1961.

[26] R. F. Shepherd, F. Ilievski, W. Choi, S. A. Morin, A. A. Stokes, A. D. Mazzeo, X. Chen, M. Wang, and G. M. Whitesides, "Multigait soft robot," *Proceedings of the national academy of sciences*, vol. 108, no. 51, pp. 20 400–20 403, 2011.

[27] B. Mosadegh, P. Polygerinos, C. Keplinger, S. Wennstedt, R. F. Shepherd, U. Gupta, J. Shim, K. Bertoldi, C. J. Walsh, and G. M. Whitesides, "Pneumatic networks for

soft robotics that actuate rapidly," *Advanced functional materials*, vol. 24, no. 15, pp. 2163–2170, 2014.

[28] T. TolleyMichael, F. ShepherdRobert, C. GallowayKevin, J. WoodRobert, M. WhitesidesGeorge *et al.*, "A resilient, untethered soft robot," *Soft Robotics*, 2014.

[29] S. A. Morin, R. F. Shepherd, S. W. Kwok, A. A. Stokes, A. Nemiroski, and G. M. Whitesides, "Camouflage and display for soft machines," *Science*, vol. 337, no. 6096, pp. 828–832, 2012.

[30] C. D. Onal and D. Rus, "Autonomous undulatory serpentine locomotion utilizing body dynamics of a fluidic soft robot," *Bioinspiration & biomimetics*, vol. 8, no. 2, p. 026003, 2013.

[31] J. Bishop-Moser, G. Krishnan, C. Kim, and S. Kota, "Design of soft robotic actuators using fluid-filled fiber-reinforced elastomeric enclosures in parallel combinations," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 4264–4269.

[32] Z. Zhakypov, M. Mete, J. Fiorentino, and J. Paik, "Programmable fluidic networks design for robotic origami sequential self-folding," in *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE, 2019, pp. 814–820.

[33] P. T. Phan, M. T. Thai, T. T. Hoang, N. H. Lovell, and T. N. Do, "Hfam: Soft hydraulic filament artificial muscles for flexible robotic applications," *Ieee Access*, vol. 8, pp. 226 637–226 652, 2020.

[34] M. Gepner, J. Mack, F. Giorgio-Serchi, and A. A. Stokes, "Fluidic flowbots: Intelligence embodied in the characteristics of recirculating fluid flow," in *2024 IEEE 7th International Conference on Soft Robotics (RoboSoft)*. IEEE, 2024, pp. 511–518.

[35] P. Breitman, Y. Matia, and A. D. Gat, "Fluid mechanics of pneumatic soft robots," *Soft Robotics*, vol. 8, no. 5, pp. 519–530, 2021.

[36] W. McMahan, V. Chitrakaran, M. Csencsits, D. Dawson, I. D. Walker, B. A. Jones, M. Pritts, D. Dienno, M. Grissom, and C. D. Rahn, "Field trials and testing of the octarm continuum manipulator," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.* IEEE, 2006, pp. 2336–2341.

[37] R. Deimel and O. Brock, "A novel type of compliant and underactuated robotic hand for dexterous grasping," *The International Journal of Robotics Research*, vol. 35, no. 1-3, pp. 161–185, 2016.

[38] S. Sanan, P. S. Lynn, and S. T. Griffith, "Pneumatic torsional actuators for inflatable robots," *Journal of Mechanisms and Robotics*, vol. 6, no. 3, p. 031003, 2014.

[39] A. A. Stokes, R. F. Shepherd, S. A. Morin, F. Ilievski, and G. M. Whitesides, "A hybrid combining hard and soft robots," *Soft Robotics*, vol. 1, no. 1, pp. 70–74, 2014.

[40] K. Suzumori, S. Endo, T. Kanda, N. Kato, and H. Suzuki, "A bending pneumatic rubber actuator realizing soft-bodied manta swimming robot," in *Proceedings 2007 IEEE international conference on robotics and automation*. IEEE, 2007, pp. 4975–4980.

[41] R. K. Katzschmann, A. D. Marchese, and D. Rus, "Hydraulic autonomous soft robotic fish for 3d swimming," in *Experimental Robotics: The 14th International Symposium on Experimental Robotics*. Springer, 2015, pp. 405–420.

[42] D. Malas, S. Wang, W. Huang, L. Lindenroth, W. Xia, and H. Liu, "A novel pneudraulic actuation method to enhance soft robot control," *Surgical & Interventional Engineering*, 2025, first posted / publisher page available

online. [Online]. Available: https://kclpure.kcl.ac.uk/portal/en/publications/a-novel-pneudraulic-actuation-method-to-enhance-soft-robot-contro/

[43] Y. Jung, K. Kwon, J. Lee, S. H. Ko, and et al., "Untethered soft actuators for soft standalone robotics," *Nature Communications*, vol. 15, p. 3510, 2024. [Online]. Available: https://www.nature.com/articles/s41467-024-47639-0

[44] J. Fraś, J. Czarnowski, M. Maciaś, J. Główka, M. Cianchetti, and A. Menciassi, "New stiff-flop module construction idea for improved actuation and sensing," in *2015 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2015, pp. 2901–2906.

[45] M. Cianchetti, T. Ranzani, G. Gerboni, I. De Falco, C. Laschi, and A. Menciassi, "Stiff-flop surgical manipulator: Mechanical design and experimental characterization of the single module," in *2013 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2013, pp. 3576–3581.

[46] H. Abidi, G. Gerboni, M. Brancadoro, J. Fras, A. Diodato, M. Cianchetti, H. Wurdemann, K. Althoefer, and A. Menciassi, "Highly dexterous 2-module soft robot for intra-organ navigation in minimally invasive surgery," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 14, no. 1, p. e1875, 2018.

[47] G. Decroly, B. Mertens, P. Lambert, and A. Delchambre, "Design, characterization and optimization of a soft fluidic actuator for minimally invasive surgery," *International journal of computer assisted radiology and surgery*, vol. 15, no. 2, pp. 333–340, 2020.

[48] M. Marzban, M. Packirisamy, and J. Dargahi, "Parametric study on fluid structure interaction of a 3d suspended polymeric microfluidics (spmf3)," *Microsystem Technologies*, vol. 24, no. 6, pp. 2549–2559, 2018.

[49] J. Shi, A. Shariati, S.-A. Abad, Y. Liu, J. S. Dai, and H. A. Wurdemann, "Stiffness modelling and analysis of soft fluidic-driven robots using lie theory," *The International Journal of Robotics Research*, vol. 43, no. 3, pp. 354–384, 2024.

[50] J. Till, V. Aloi, and C. Rucker, "Real-time dynamics of soft and continuum robots based on cosserat rod models," *The International Journal of Robotics Research*, vol. 38, no. 6, pp. 723–746, 2019.

[51] Y. Matia and A. D. Gat, "Dynamics of elastic beams with embedded fluid-filled parallel-channel networks," *Soft robotics*, vol. 2, no. 1, pp. 42–47, 2015.

[52] A. Kalpakli Vester, R. Örlü, and P. H. Alfredsson, "Turbulent flows in curved pipes: Recent advances in experiments and simulations," *Applied Mechanics Reviews*, vol. 68, no. 5, p. 050802, 2016.

[53] M. Roshanfar, J. Dargahi, and A. Hooshiar, "Design optimization of a hybrid-driven soft surgical robot with biomimetic constraints," *Biomimetics*, vol. 9, no. 1, 2024, publisher: MDPI.

[54] T. Torkaman, M. Roshanfar, J. Dargahi, and A. Hooshiar, "Accurate embedded force sensor for soft robots with rate-dependent deep neural calibration," in *2022 IEEE International Symposium on Robotic and Sensors Environments (ROSE)*. IEEE, 2022, pp. 1–7.

Appendix A

Source Codes

A.1 The Flow Sensor Arduino Script

```
// The sensor pin and calibration constant
#define SENSOR_PIN 2  // Flow sensor output connected to digital
    pin 2
#define K_FACTOR 6900 // Pulses per liter (provided by sensor
    manufacturer)

const float readFrequency = 30.0;  //  Hz

// Automatically calculated update interval in milliseconds
const uint32_t updateInterval = 1000.0 / readFrequency;
```

```
volatile uint32_t pulseCount = 0; // Stores the number of pulses
    detected. unint32_t: unsigned integer32 (0 to 4,294,967,295)
    standard

unsigned long previousMillis = 0; // Stores the last time the flow
     rate was calculated
float flowRate = 0.0; // Stores the calculated flow rate in LPM

// Interrupt Service Routine (ISR) - This function runs
    automatically when a pulse is detected

void countPulse() {
        pulseCount++; // Increment the pulse counter on each
            rising edge
}

void setup() {
        Serial.begin(115200);

        pinMode(SENSOR_PIN, INPUT); // Set sensor pin as input

        // Attach an interrupt to SENSOR_PIN, trigger on rising
            edge, and call countPulse()
        attachInterrupt(digitalPinToInterrupt(SENSOR_PIN),
            countPulse, RISING);
```

```
        // Store the current time to start timing the flow rate
           calculations
        previousMillis = millis();
}

void loop() {
        // Get the current time
        unsigned long currentMillis = millis();

        // Check if 1000 milliseconds has passed since last
           calculation
        if (currentMillis - previousMillis >= updateInterval) {
                // Disable interrupts while processing to ensure
                   accuracy
                detachInterrupt(digitalPinToInterrupt(SENSOR_PIN))
                   ;

                // Convert pulse count to frequency (Hz), since we
                    count pulses in 1 second
                float frequency = pulseCount / (updateInterval
                   /1000.0); // Pulses per second (Hz)

                // Convert frequency to flow rate in LPM using the
                    formula: Q = (f * 60) / K
```

```
            flowRate = (frequency * 60.0) / K_FACTOR;

            // Print the flow rate to the serial monitor
            Serial.println(flowRate);

            // Reset the pulse count for the next measurement
               period
            pulseCount = 0;

            // Update the time for the next calculation
            previousMillis = currentMillis;

            // Re-enable the interrupt to continue detecting
               pulses
            attachInterrupt(digitalPinToInterrupt(SENSOR_PIN),
                countPulse, RISING);
        }
}
```

A.2 The Image Processing Matlab Script

```
close all; clc;

% ---- Paths & image ----
imageFolder = 'C:\Users\umroot\OneDrive - Concordia University -
    Canada\Thesis project\image processing\main experiment\matlab
    codes\robot images\Main experiment tests\29 june 2025\try3\
    candid frames';
resultsFolder = fullfile(imageFolder, 'result export');
imageName = 'frame_00000';
imageFile = fullfile(imageFolder, [imageName, '.jpg']);

I = imread(imageFile);
I_undistorted = undistortImage(I, camParamSept10_3coef, '
    OutputView', 'valid');
figure; imshow(I_undistorted); title('Undistorted Image');

% ---- Binary segmentation & cleanup ----
Igray = rgb2gray(I_undistorted);
BW = ~imbinarize(Igray, 'adaptive', 'ForegroundPolarity', 'dark',
    'Sensitivity', 0.4);
BW = imfill(BW, 'holes');
BW = imopen(BW, strel('disk',3));
```

```
BW = bwareafilt(BW,1);
figure; imshow(BW); title('Cleaned Binary Image');

% ---- Distance & perimeter ----
D = bwdist(~BW);
perim = bwperim(BW);
[by, bx] = find(perim);

% ---- Parameters ----
spurIterations   = 2;     % skeleton spur prune
extensionSamples = 40;    % initial extension samples
extensionRadius  = 3;     % local refinement radius
extendStart = true;       % extend only the first endpoint
extendEnd   = false;

% ---- Skeletonize & prune ----
skel = bwskel(BW);
if spurIterations>0, skel = bwmorph(skel,'spur',spurIterations);
   end
endPts = bwmorph(skel,'endpoints');
[ey, ex] = find(endPts);

% ---- Fallback to ridge if endpoints insufficient ----
if numel(ex) < 2
warning('Skeleton endpoints <2; using medial ridge fallback.');
```

```
ridge = imregionalmax(D);
[y_r, x_r] = find(ridge);
if isempty(x_r), error('No ridge points found.'); end
pixelPoints = double([x_r, y_r]);
Dmat = squareform(pdist(pixelPoints));
[~, idx] = max(Dmat(:)); [i1,i2] = ind2sub(size(Dmat), idx);
pt1 = pixelPoints(i1,:); pt2 = pixelPoints(i2,:);
v = (pt2-pt1)/norm(pt2-pt1);
proj = (pixelPoints-pt1)*v'; [~, order] = sort(proj);
pixelPointsOrdered = pixelPoints(order,:);
perim = bwperim(BW);
[y_b,x_b] = find(perim); boundaryPoints = double([x_b,y_b]);
distB = sqrt(sum((boundaryPoints-pt2).^2,2)); [~, bi] = min(distB)
    ;
pixelPointsExtended = [pixelPointsOrdered; boundaryPoints(bi,:)];
x_pix = pixelPointsExtended(:,1); y_pix = pixelPointsExtended(:,2)
    ;
pixelPointsOrdered = pixelPointsExtended;
else
% ---- Build graph on skeleton pixels and find ordered centerline
    ----
[sy, sx] = find(skel); Nsk = numel(sx);
idxMapSk = zeros(size(skel)); idxMapSk(sub2ind(size(skel), sy, sx)
    ) = 1:Nsk;
```

```
ep_pts = [ex, ey];
if size(ep_pts,1) > 2
D_ep = squareform(pdist(ep_pts)); [~, maxidx] = max(D_ep(:)); [e1i
    ,e2i] = ind2sub(size(D_ep), maxidx);
startPt = ep_pts(e1i,:); endPt = ep_pts(e2i,:);
else
startPt = ep_pts(1,:); endPt = ep_pts(2,:);
end


neigh = [-1 -1; -1 0; -1 1; 0 -1; 0 1; 1 -1; 1 0; 1 1];
ii = []; jj = []; w = [];
for k = 1:Nsk
xk = sx(k); yk = sy(k);
for n = 1:8
nx = xk + neigh(n,1); ny = yk + neigh(n,2);
if nx>=1 && nx<=size(skel,2) && ny>=1 && ny<=size(skel,1)
j = idxMapSk(ny,nx);
if j>0
ii(end+1)=k; jj(end+1)=j; w(end+1)=hypot(sx(k)-nx, sy(k)-ny);
end
end
end
end
Gs = graph(ii,jj,w);
```

```
startIdx = idxMapSk(startPt(2), startPt(1));
endIdx   = idxMapSk(endPt(2),   endPt(1));
if startIdx==0, [~, startIdx] = min((sx-startPt(1)).^2 + (sy-
   startPt(2)).^2); end
if endIdx==0,   [~, endIdx]   = min((sx-endPt(1)).^2 + (sy-endPt
   (2)).^2); end

pathSk = shortestpath(Gs, startIdx, endIdx);
if isempty(pathSk), error('No path on skeleton.'); end
orderedSkel = [sx(pathSk), sy(pathSk)];

ext_start = []; ext_end = [];
if extendStart, ext_start = extend_endpoint(orderedSkel(1,:), bx,
   by, D, extensionSamples, extensionRadius); end
if extendEnd,   ext_end   = extend_endpoint(orderedSkel(end,:), bx
   , by, D, extensionSamples, extensionRadius); end

if ~isempty(ext_start)
ext_start_rev = flipud(ext_start);
if ~isempty(ext_start_rev) && isequal(round(ext_start_rev(end,:)),
    round(orderedSkel(1,:)))
ext_start_rev(end,:) = [];
end
else
ext_start_rev = [];
```

```
end
if ~isempty(ext_end) && ~isempty(ext_end) && isequal(round(ext_end
    (1,:)), round(orderedSkel(end,:)))
ext_end(1,:) = [];
end

pixelPointsExtended = [ext_start_rev; orderedSkel; ext_end];
x_pix = pixelPointsExtended(:,1); y_pix = pixelPointsExtended(:,2)
    ;
pixelPointsOrdered = pixelPointsExtended;
end

% --- Ensure base-to-tip ordering (y increases downward) ---
if exist('y_pix','var') && exist('pixelPointsOrdered','var')
if y_pix(1) > y_pix(end)
pixelPointsOrdered = flipud(pixelPointsOrdered);
x_pix = pixelPointsOrdered(:,1); y_pix = pixelPointsOrdered(:,2);
end
else
error('Centerline extraction failed.');
end

% --- Map to world coordinates ---
rotMat = camParamSept10_3coef.RotationMatrices(:,:,1);
transVec = camParamSept10_3coef.TranslationVectors(1,:);
```

```
tform = rigidtform3d(rotMat', transVec);
intrinsics = camParamSept10_3coef.Intrinsics;
worldPointsAll = img2world2d(pixelPointsOrdered, tform, intrinsics
    );
x_sorted = worldPointsAll(:,1); y_sorted = worldPointsAll(:,2);

% ---- Arc length & spline fit ----
dx = diff(x_sorted); dy = diff(y_sorted);
s_raw = [0; cumsum(sqrt(dx.^2 + dy.^2))]; s_norm = s_raw / s_raw(
    end);
[s_norm_unique, ia] = unique(s_norm, 'stable');
x_sorted = x_sorted(ia); y_sorted = y_sorted(ia);

x_spline = spline(s_norm_unique, x_sorted);
y_spline = spline(s_norm_unique, y_sorted);

s_query = linspace(0,1,200); x_fit = ppval(x_spline, s_query);
    y_fit = ppval(y_spline, s_query);

% ---- Save deformation variable ----
deformationXY = [y_fit(:), x_fit(:)];
deformVarName = ['deformationXY_' imageName];
deformFile = fullfile(resultsFolder, [deformVarName, '.mat']);
save(deformFile, 'deformationXY');
```

```
% ---- Plots & overlay ----
figure; plot(s_query, x_fit, 'b-', 'LineWidth', 2); xlabel('
    Normalized Arc Length s'); ylabel('Lateral Deflection [units]')
    ; grid on;
figure; plot(y_fit, x_fit, 'LineWidth', 2); xlabel('X [units]');
    ylabel('Y [units]'); grid on;


figure; imshow(I_undistorted); hold on; plot(x_pix, y_pix, 'r.-');


outFile = fullfile(resultsFolder, ['Centerline_' imageName '.jpg'
    ]);
segments = [];
if numel(x_pix) >= 2
segments = [x_pix(1:end-1), y_pix(1:end-1), x_pix(2:end), y_pix(2:
    end)];
end
if ~isempty(segments)
if size(I_undistorted,3) == 1, I_rgb = repmat(I_undistorted,1,1,3)
    ; else I_rgb = I_undistorted; end
try
I_with_line = insertShape(I_rgb, 'Line', segments, 'Color', 'red',
     'LineWidth', 3);
catch
frame = getframe(gca); I_with_line = frame.cdata;
end
```

```
else
I_with_line = I_undistorted;
end
imwrite(I_with_line, outFile, 'jpg', 'Quality', 95);

fprintf('Saved: %s\n', outFile);
fprintf('Option A completed (extendStart=%d, extendEnd=%d).\n',
    extendStart, extendEnd);

% ---------- Local functions ----------
function extPts = extend_endpoint(epxy, bx, by, D, samples, r)
regionR0 = max(round(samples), 40); regionRmax = 600; expandFactor
    = 1.6;
requireExactBoundary = true; globalFallback = true; imgH = size(D
    ,1); imgW = size(D,2);
regionR = regionR0; success = false; extPts = [];

while regionR <= regionRmax
xmin = max(1, round(epxy(1)-regionR)); xmax = min(imgW, round(epxy
    (1)+regionR));
ymin = max(1, round(epxy(2)-regionR)); ymax = min(imgH, round(epxy
    (2)+regionR));
Dwin = D(ymin:ymax, xmin:xmax);
[ry, rx] = find(Dwin > 0);
```

```
if isempty(rx), regionR = ceil(regionR*expandFactor); continue;
   end
xg = rx + xmin - 1; yg = ry + ymin - 1; nodeCoords = [xg, yg];
   Nloc = size(nodeCoords,1);
idxMapLocal = zeros(size(Dwin)); idxMapLocal(sub2ind(size(Dwin),
   ry, rx)) = 1:Nloc;
neigh = [-1 -1; -1 0; -1 1; 0 -1; 0 1; 1 -1; 1 0; 1 1];
ii = zeros(Nloc*8,1); jj = zeros(Nloc*8,1); ww = zeros(Nloc*8,1);
   ptr = 0;
for k = 1:Nloc
xk = nodeCoords(k,1); yk = nodeCoords(k,2); dk = D(yk,xk);
for n = 1:8
xn = xk + neigh(n,1); yn = yk + neigh(n,2);
if xn>=xmin && xn<=xmax && yn>=ymin && yn<=ymax
lx = xn - xmin + 1; ly = yn - ymin + 1; j = idxMapLocal(sub2ind(
   size(Dwin), ly, lx));
if j>0, dn = D(yn,xn); meanD = (dk+dn)/2; ptr = ptr+1; ii(ptr)=k;
   jj(ptr)=j; ww(ptr)=1/(meanD+eps); end
end
end
end
if ptr==0, regionR = ceil(regionR*expandFactor); continue; end
ii = ii(1:ptr); jj = jj(1:ptr); ww = ww(1:ptr); Gw = graph(ii,jj,
   ww);
```

```
dstart = (nodeCoords(:,1)-epxy(1)).^2 + (nodeCoords(:,2)-epxy(2))
    .^2; [~, startNode] = min(dstart);
maskInside = bx>=xmin & bx<=xmax & by>=ymin & by<=ymax;
if ~any(maskInside), regionR = ceil(regionR*expandFactor);
    continue; end
bx_local = bx(maskInside); by_local = by(maskInside); nb = numel(
    bx_local);
bNodeIDs = zeros(nb,1);
for j = 1:nb, [~, idxNearest] = min((nodeCoords(:,1)-bx_local(j))
    .^2 + (nodeCoords(:,2)-by_local(j)).^2); bNodeIDs(j)=idxNearest
    ; end
bNodeIDs = unique(bNodeIDs);
distVec = distances(Gw, startNode);
reachable = ~isinf(distVec(bNodeIDs));
if ~any(reachable), regionR = ceil(regionR*expandFactor); continue
    ; end
bIDs_reach = bNodeIDs(reachable); [~, minpos] = min(distVec(
    bIDs_reach)); targetNode = bIDs_reach(minpos);
pathIdx = shortestpath(Gw, startNode, targetNode);
if isempty(pathIdx), regionR = ceil(regionR*expandFactor);
    continue; end
extPts_candidate = nodeCoords(pathIdx,:);
lastNodeCoord = extPts_candidate(end,:);
d2boundary = min((bx-lastNodeCoord(1)).^2 + (by-lastNodeCoord(2))
    .^2);
```

```
if requireExactBoundary
if d2boundary <= 16, extPts = extPts_candidate; success = true;
   break; else regionR = ceil(regionR*expandFactor); continue; end
else
extPts = extPts_candidate; success = true; break;
end
end

if success
[~, ia] = unique(round(extPts(:,1:2)), 'rows', 'stable'); extPts =
    extPts(ia,:); return;
end

if globalFallback
[g_ry, g_rx] = find(D > 0); Nglob = numel(g_rx);
idxMapGlob = zeros(size(D)); idxMapGlob(sub2ind(size(D), g_ry,
   g_rx)) = 1:Nglob;
gNodeCoords = [g_rx, g_ry]; ii = zeros(Nglob*8,1); jj = zeros(
   Nglob*8,1); ww = zeros(Nglob*8,1); ptr = 0;
for k = 1:Nglob
xk = gNodeCoords(k,1); yk = gNodeCoords(k,2); dk = D(yk,xk);
for n = 1:8
xn = xk + neigh(n,1); yn = yk + neigh(n,2);
if xn>=1 && xn<=size(D,2) && yn>=1 && yn<=size(D,1)
j = idxMapGlob(yn,xn);
```

```
if j>0, dn = D(yn,xn); ptr = ptr+1; ii(ptr)=k; jj(ptr)=j; ww(ptr)
    =1/((dk+dn)/2+eps); end
end
end
end
ii = ii(1:ptr); jj = jj(1:ptr); ww = ww(1:ptr); Gglob = graph(ii,
    jj,ww);
dstartg = (gNodeCoords(:,1)-epxy(1)).^2 + (gNodeCoords(:,2)-epxy
    (2)).^2; [~, startG] = min(dstartg);
nb = numel(bx); bNodeIDs = zeros(nb,1);
for j = 1:nb, [~, idxNearest] = min((gNodeCoords(:,1)-bx(j)).^2 +
    (gNodeCoords(:,2)-by(j)).^2); bNodeIDs(j)=idxNearest; end
bNodeIDs = unique(bNodeIDs);
distVec = distances(Gglob, startG); reachable = ~isinf(distVec(
    bNodeIDs));
if ~any(reachable)
extPts = straightline_extension_fallback(epxy, bx, by, D, samples,
    r);
[~, ia] = unique(round(extPts(:,1:2)), 'rows', 'stable'); extPts =
    extPts(ia,:); return;
end
bIDs_reach = bNodeIDs(reachable); [~, minpos] = min(distVec(
    bIDs_reach)); targetNode = bIDs_reach(minpos);
pathIdx = shortestpath(Gglob, startG, targetNode);
if isempty(pathIdx)
```

```
extPts = straightline_extension_fallback(epxy, bx, by, D, samples,
    r);
[~, ia] = unique(round(extPts(:,1:2)), 'rows', 'stable'); extPts =
    extPts(ia,:); return;
end
extPts = gNodeCoords(pathIdx,:);
[~, ia] = unique(round(extPts(:,1:2)), 'rows', 'stable'); extPts =
    extPts(ia,:); return;
end

extPts = straightline_extension_fallback(epxy, bx, by, D, samples,
    r);
[~, ia] = unique(round(extPts(:,1:2)), 'rows', 'stable'); extPts =
    extPts(ia,:);
end

function extPts = straightline_extension_fallback(epxy, bx, by, D,
    samples, r)
dists = (bx - epxy(1)).^2 + (by - epxy(2)).^2; [~, bi] = min(dists
   ); bpt = [bx(bi), by(bi)];
nS = max(round(samples), 3); xs = linspace(epxy(1), bpt(1), nS);
   ys = linspace(epxy(2), bpt(2), nS);
extPts = zeros(nS,2);
for t = 1:nS
```

```
x0 = round(xs(t)); y0 = round(ys(t)); xmin = max(1, x0-r); xmax =
    min(size(D,2), x0+r);
ymin = max(1, y0-r); ymax = min(size(D,1), y0+r); localD = D(ymin:
    ymax, xmin:xmax);
if isempty(localD), extPts(t,:) = [xs(t), ys(t)]; else [~, midx] =
     max(localD(:)); [mrow,mcol] = ind2sub(size(localD), midx);
extPts(t,:) = [xmin + mcol - 1, ymin + mrow - 1]; end
end
[~, ia] = unique(round(extPts(:,1:2)), 'rows', 'stable'); extPts =
     extPts(ia,:);
end
```

Appendix B

Simulation Results for Turbulence Model Comparison

Table B.1: CFD and experimental results for the **k-ε turbulence model**, separated by Outer and Inner Bend regions. Experimental data reproduced from [8].

Position	Outer Bend			Inner Bend		
	CFD	**EXP**	**Error (%)**	**CFD**	**EXP**	**Error (%)**
-14	−0.123	−0.411	70.181	−0.123	−0.423	71.027
-9	−0.241	−0.503	51.989	−0.241	−0.508	52.463
-6	−0.307	−0.528	41.916	−0.307	−0.536	42.783
-4	−0.349	−0.569	38.694	−0.349	−0.595	41.373
-2.5	−0.380	−0.600	36.673	−0.380	−0.627	39.394
-1.5	−0.380	−0.621	38.814	−0.380	−0.660	42.43
-1	−0.408	−0.636	35.80	−0.413	−0.675	38.759
-0.5	−0.408	−0.639	36.111	−0.434	−0.707	38.572
22.5°	−0.325	−0.616	47.306	−0.627	−0.844	25.703
45°	−0.365	−0.630	42.05	−0.662	−0.892	25.802
67.5°	−0.442	−0.670	34.024	−0.728	−0.951	23.415
112.5°	−0.584	−0.791	26.1549	−0.881	−1.107	20.399
135°	−0.655	−0.883	25.823	−0.955	−1.178	18.892
157.5°	−0.726	−0.987	26.426	−1.026	−1.224	16.152
0.5	−0.988	−1.149	14.027	−1.018	−1.245	18.251
1	−1.016	−1.182	14.064	−1.022	−1.251	18.271
1.5	−1.032	−1.202	14.175	−1.033	−1.259	17.989
4	−1.092	−1.249	12.554	−1.088	−1.266	14.026
6	−1.140	−1.304	12.576	−1.138	−1.326	14.176
9	−1.212	−1.351	10.269	−1.212	−1.381	12.27
14	−1.331	−1.447	8.017	−1.331	−1.477	9.887

Table B.2: CFD and experimental results for the **k-ω SST turbulence model**, separated by Outer and Inner Bend regions. Experimental data reproduced from [8].

Position	Outer Bend			Inner Bend		
	CFD	**EXP**	**Error (%)**	**CFD**	**EXP**	**Error (%)**
-14	−0.139 309	−0.411	66.105	−0.139 306	−0.423	67.067
-9	−0.265	−0.503	47.373	−0.265	−0.508	47.891
-6	−0.330 437	−0.528	37.417	−0.330 435	−0.536	38.352
-4	−0.372 614	−0.569	34.514	−0.372 613	−0.595	37.376
-2.5	−0.403 675	−0.6	32.721	−0.403 719	−0.627	35.611
-1.5	−0.423 654	−0.621	31.779	−0.424 763	−0.66	35.642
-1	−0.431 678	−0.636	32.126	−0.437 166	−0.675	35.235
-0.5	−0.431 24	−0.639	32.514	−0.458 359	−0.707	35.168
22.5°	−0.3413	−0.616	44.594	−0.643 96	−0.844	23.702
45°	−0.3835	−0.63	39.128	−0.678 08	−0.892	23.982
67.5°	−0.467 74	−0.67	30.189	−0.752 41	−0.951	20.882
112.5°	−0.614 51	−0.791	22.312	−0.916 64	−1.107	17.196
135°	−0.692 27	−0.883	21.60	−0.995 98	−1.178	15.451
157.5°	−0.771 38	−0.987	21.846	−1.072 64	−1.224	12.366
0.5	−1.045 87	−1.149	8.976	−1.076 34	−1.245	13.547
1	−1.075 32	−1.182	9.025	−1.082 33	−1.251	13.483
1.5	−1.092 59	−1.202	9.102	−1.094 05	−1.259	13.102
4	−1.160 47	−1.249	7.088	−1.157 88	−1.266	8.540
6	−1.211 97	−1.304	7.058	−1.210 33	−1.326	8.723
9	−1.287 37	−1.351	4.71	−1.286 68	−1.381	6.83
14	−1.408 64	−1.447	2.651	−1.408 57	−1.477	4.633

Table B.3: CFD and experimental results for the **RSM Stress-ω turbulence model**, separated by Outer and Inner Bend regions. Experimental data reproduced from [8].

Position	Outer Bend			Inner Bend		
	CFD	**EXP**	**Error (%)**	**CFD**	**EXP**	**Error (%)**
-14	−0.110 012	−0.411	73.233	−0.110 012	−0.423	73.992
-9	−0.222 479	−0.503	55.77	−0.222 479	−0.508	56.205
-6	−0.282 248	−0.528	46.544	−0.282 248	−0.536	47.342
-4	−0.320 937	−0.569	43.596	−0.320 937	−0.595	46.061
-2.5	−0.349 521	−0.6	41.747	−0.349 562	−0.627	44.248
-1.5	−0.367 928	−0.621	40.752	−0.368 978	−0.66	44.094
-1	−0.375 224	−0.636	41.003	−0.380 532	−0.675	43.625
-0.5	−0.374 08	−0.639	41.458	−0.400 858	−0.707	43.302
22.5°	−0.281 71	−0.616	54.268	−0.582 48	−0.844	30.986
45°	−0.322 36	−0.63	48.832	−0.606 35	−0.892	32.023
67.5°	−0.3949	−0.67	41.06	−0.674 47	−0.951	29.078
112.5°	−0.516 52	−0.791	34.7	−0.819 15	−1.107	26.003
135°	−0.583 59	−0.883	33.908	−0.882 77	−1.178	25.062
157.5°	−0.652 04	−0.987	33.937	−0.946 59	−1.224	22.664
0.5	−0.912 11	−1.149	20.617	−0.938 86	−1.245	24.59
1	−0.938 03	−1.182	20.64	−0.942 59	−1.251	24.653
1.5	−0.952 51	−1.202	20.756	−0.950 57	−1.259	24.498
4	−1.009 33	−1.249	19.189	−1.003 16	−1.266	20.761
6	−1.054 58	−1.304	19.127	−1.051 64	−1.326	20.691
9	−1.123 16	−1.351	16.865	−1.121 37	−1.381	18.8
14	−1.228 97	−1.447	15.068	−1.227 68	−1.477	16.88

Table B.4: CFD and experimental results for the **RSM Stress-BSL** turbulence model, separated by Outer and Inner Bend regions. Experimental data reproduced from [8].

Position	Outer Bend			Inner Bend		
	CFD	**EXP**	**Error (%)**	**CFD**	**EXP**	**Error (%)**
-14	−0.1098	−0.411	73.286	−0.1098	−0.423	74.043
-9	−0.221 67	−0.503	55.93	−0.221 67	−0.508	56.364
-6	−0.281 64	−0.528	46.66	−0.281 64	−0.536	47.456
-4	−0.320 67	−0.569	43.643	−0.320 67	−0.595	46.106
-2.5	−0.349 55	−0.6	41.742	−0.349 59	−0.627	44.244
-1.5	−0.368 16	−0.621	40.714	−0.3692	−0.66	44.061
-1	−0.375 58	−0.636	40.947	−0.380 83	−0.675	43.581
-0.5	−0.374 58	−0.639	41.381	−0.401 22	−0.707	43.25
22.5°	−0.2837	−0.616	53.945	−0.584 76	−0.844	30.716
45°	−0.322 84	−0.63	48.755	−0.610 26	−0.892	31.586
67.5°	−0.396 43	−0.67	40.831	−0.676 09	−0.951	28.908
112.5°	−0.519 12	−0.791	34.371	−0.822 22	−1.107	25.725
135°	−0.585 64	−0.883	33.677	−0.885 41	−1.178	24.838
157.5°	−0.6534	−0.987	33.799	−0.948 25	−1.224	22.529
0.5	−0.909 16	−1.149	20.874	−0.936 91	−1.245	24.746
1	−0.934 76	−1.182	20.918	−0.940 38	−1.251	24.83
1.5	−0.948 91	−1.202	21.056	−0.948 68	−1.259	24.648
4	−1.0046	−1.249	19.568	−0.999 78	−1.266	21.028
6	−1.049 29	−1.304	19.533	−1.046 95	−1.326	21.044
9	−1.115 79	−1.351	17.41	−1.114 58	−1.381	19.292
14	−1.218 89	−1.447	15.764	−1.218 34	−1.477	17.513

Table B.5: CFD and experimental results for the **k-ω SST turbulence model** with refined mesh, separated by Outer and Inner Bend regions. Experimental data reproduced from [8].

Position	Outer Bend			Inner Bend		
	CFD	**EXP**	**Error (%)**	**CFD**	**EXP**	**Error (%)**
-14	−0.144 53	−0.411	64.836	−0.144 53	−0.423	65.833
-9	−0.265 17	−0.503	47.282	−0.265 17	−0.508	47.801
-6	−0.329 57	−0.528	37.581	−0.329 57	−0.536	38.513
-4	−0.371 11	−0.569	34.778	−0.371 11	−0.595	37.628
-2.5	−0.401 54	−0.6	33.076	−0.401 58	−0.627	35.952
-1.5	−0.4291	−0.621	30.901	−0.421 17	−0.66	36.186
-1	−0.434 37	−0.636	31.703	−0.422 21	−0.675	37.45
-0.5	−0.428 68	−0.639	32.914	−0.455 01	−0.707	35.642
22.5°	−0.339 26	−0.616	44.925	−0.642 74	−0.844	23.846
45°	−0.387 04	−0.63	38.565	−0.680 25	−0.892	23.739
67.5°	−0.465 87	−0.67	30.467	−0.755 84	−0.951	20.521
112.5°	−0.615 16	−0.791	22.23	−0.917 25	−1.107	17.141
135°	−0.691 34	−0.883	21.705	−0.995 65	−1.178	15.48
157.5°	−0.768 13	−0.987	22.175	−1.071 17	−1.224	12.486
0.5	−1.044 74	−1.149	9.074	−1.078	−1.245	13.414
1	−1.074 59	−1.182	9.087	−1.083 08	−1.251	13.423
1.5	−1.091 44	−1.202	9.198	−1.093 88	−1.259	13.116
4	−1.153 95	−1.249	7.61	−1.151 75	−1.266	9.024
6	−1.2002	−1.304	7.96	−1.1989	−1.326	9.585
9	−1.266 83	−1.351	6.23	−1.266 42	−1.381	8.297
14	−1.376 07	−1.447	4.902	−1.376 05	−1.477	6.835

Table B.6: CFD and experimental results for the **URANS k-ω SST turbulence model**, separated by Outer and Inner Bend regions. Experimental data reproduced from [8].

Position	Outer Bend			Inner Bend		
	CFD	**EXP**	**Error (%)**	**CFD**	**EXP**	**Error (%)**
-14	−0.140 87	−0.411	35.725	−0.140 87	−0.423	66.698
-9	−0.265 25	−0.503	47.266	−0.265 25	−0.508	47.785
-6	−0.330 45	−0.528	37.415	−0.330 45	−0.536	38.349
-4	−0.372 47	−0.569	34.54	−0.372 47	−0.595	37.401
-2.5	−0.403 36	−0.6	32.773	−0.403 41	−0.627	35.661
-1.5	−0.423 17	−0.621	31.857	−0.424 22	−0.66	35.725
-1	−0.431 17	−0.636	32.206	−0.436 43	−0.675	35.344
-0.5	−0.430 89	−0.639	32.569	−0.457 16	−0.707	35.338
22.5°	−0.3425	−0.616	44.399	−0.645 82	−0.844	23.481
45°	−0.386 75	−0.63	38.611	−0.681 86	−0.892	23.559
67.5°	−0.465 61	−0.67	30.506	−0.754 59	−0.951	20.653
112.5°	−0.616 97	−0.791	22.002	−0.9164	−1.107	17.218
135°	−0.692 65	−0.883	21.557	−0.994 69	−1.178	15.561
157.5°	−0.7678	−0.987	22.209	−1.0692	−1.224	12.647
0.5	−1.037 87	−1.149	9.672	−1.069 28	−1.245	14.114
1	−1.067 11	−1.182	9.72	−1.074 62	−1.251	14.099
1.5	−1.083 93	−1.202	9.823	−1.085 81	−1.259	13.756
4	−1.148 59	−1.249	8.039	−1.146 41	−1.266	9.446
6	−1.196 67	−1.304	8.231	−1.195 15	−1.326	9.868
9	−1.266 48	−1.351	6.256	−1.265 85	−1.381	8.338
14	−1.378 69	−1.447	4.721	−1.378 63	−1.477	6.66