Task-Driven Automated Design and Optimisation of Soft Pneumatic Bellow Actuators



Yao Yao

Jesus College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Michaelmas 2025

Acknowledgements

In this section, I would like to express my sincere gratitude to all those who have contributed to the successful completion of my doctoral thesis, including my supervisor Prof. Perla Maiolino, my collaborators, colleagues, friends, and family. This journey would not have been possible without their guidance, support, and encouragement, which have shaped both my academic development and personal growth throughout my time at Oxford.

First and foremost, I would like to express my deepest gratitude to my supervisor, Prof. Perla Maiolino. I am sincerely thankful for the opportunities she has given me, from supporting my MSc studies at Oxford in 2020 to allowing me to transfer to the DPhil programme in 2023, which made this doctoral journey possible. Throughout my DPhil, Perla provided continuous and dedicated support for my research, including hands-on guidance, patient teaching, and valuable opportunities for collaboration and professional development. Her mentorship helped me grow into a more independent researcher, while her trust allowed me the freedom to explore and define my own research direction. Beyond research guidance, Perla has taught me, through her own practice, what it means to pursue meaningful science, emphasising rigour, integrity, and impact beyond publication metrics. I am also deeply grateful for her personal understanding during difficult periods of my DPhil, when her encouragement made a significant difference to both my work and well-being.

I am also grateful to my collaborators, Dr Liang He, Dr Alessandro Albini, Mr Shaohong Zhong, Dr Marco Pontin, Mr Maximilian Westermann, and Dr David Howard, for their valuable contributions to my research. Their expertise, constructive discussions, and thoughtful feedback played an important role in shaping this work, and I greatly appreciated the collaborative spirit and professionalism they brought to our joint efforts.

I would like to thank my colleagues and lab mates, Wenye Ouyang, Oliver Shorthose, Sihan Wang, Fengyu Zhong, Ningzhe Hou, Giammarco Caroleo, and Ellis Capp, for creating a supportive and stimulating working environment. I am also deeply thankful to my friends Dr Li Shen, Ms Zhaoxi Shen, Ms Ruyue Wang, Dr Yana Lishkova, Ms Yueru Chi, and Dr Zipei Song, who accompanied me throughout my time in Oxford. Their care, understanding, and help—especially during the most challenging periods—meant more to me than words can fully express.

My heartfelt gratitude also goes to my dear friend Ms Yi Ding. Over the past five years of my DPhil, as well as the year before, she generously took care of my dog, Chouchou, with great dedication and love. Her generosity and steadfast support became an invaluable source of strength during my time in Oxford.

I would like to give my most sincere thanks to my partner, Dr Yuwen Chen. We met when I first arrived in Oxford, and his presence has accompanied me throughout my entire DPhil journey. He has offered unwavering encouragement, support, and love, standing by me through joyful, difficult, and ordinary days alike. As an outstanding scholar and an exceptional person, he has been a constant source of inspiration and motivation. I feel deeply grateful to walk this path alongside him, and I look forward to sharing the future together.

Above all, I owe my deepest gratitude to my family, especially my father, Mr Tusheng Yao, and my mother, Ms Jie Zhang. Their selfless support, understanding, and even indulgence have been the greatest source of encouragement throughout my academic journey, and I am profoundly thankful for their love and trust.

This thesis is also dedicated to my grandfather, Mr Jincheng Zhang, who was a school teacher and passed away during my DPhil. His belief in education has profoundly influenced me, and his deep trust in me gave me immense strength throughout my journey. I hope this work would have made him proud.

Abstract

Soft robotic systems have gained increasing attention due to their inherent compliance, safety, and adaptability, making them well-suited for interactions with humans and unstructured environments. Among these, bellow soft pneumatic actuator (SPA)s offer a promising solution for task-oriented applications that require large deformation, structural compliance, and geometry-level customisation. However, the design of these actuators for intended task applications remains difficult to systematise due to strong nonlinearities and tight coupling between geometry, actuation, and task interaction, often relying on empirical trial-and-error and case-specific tuning. This reliance not only limits design efficiency but also constrains the broader application of such actuators in diverse tasks, including shape-matching, manipulation, and locomotion. To address these challenges, this DPhil aims to develop an automated, task-driven design approach that integrates computational modelling and optimisation, enabling a systematic translation from task-level objectives to manufacturable bellow SPA designs.

This thesis presents our contributions toward the aforementioned objective in a step-by-step manner. Firstly, to expedite the design process and reduce reliance on trial and error, an open-source analysis toolbox for 3D-printed bending bellow SPAs was developed. The toolbox, featuring an interactive graphical user interface (GUI), supports rapid, human-guided exploration of bellow SPA designs through a fast analytical model while retaining physically grounded performance evaluation via finite element method (FEM)-based simulation. Secondly, to expand the design space beyond uniform bending, a modular actuator representation was introduced to support more complex actuator geometries and diverse deformation patterns for task-oriented applications. Building on this representation, a FEM-based surrogate modelling approach was developed to enable optimisation-based inverse design for shape-matching objectives, allowing actuator geometries to be generated automatically from task specifications. This approach was also implemented as an open-source GUI-based toolbox, supporting both kinematic prediction for customised actuator designs and automated actuator generation based on input target shapes. Thirdly, building on the shape-matching optimisation framework, an investigation was performed on different modelling and optimisation combinations in terms of computational cost, convergence behaviour, and achieved design quality,

providing practical insight into optimisation-based task-driven design. Finally, this thesis proposes a task-driven design pipeline that supports generalisation across parameterised actuator designs and task scenarios. By combining high-fidelity modelling, reduced-order representations, and fast task-level simulation, the pipeline enables task-driven design and optimisation to be performed without reconstructing the entire modelling workflow for each new design instance or task application.

Overall, this thesis provides the soft robotics community with a systematic task-driven design methodology that bridges task objectives and manufacturable soft actuator designs. By improving design efficiency and consistency while supporting a wider range of task-oriented applications, the proposed methodology is not limited to bellow SPAs, but instead offers general-purpose modelling and optimisation tools that the research community can readily build upon.

Contents

List of Figures **ix**

1 Introduction **1**

1.1 Motivation 1
1.2 Objective and our contributions 3
1.3 Outline 6
1.4 Publications 8
1.4.1 Papers directly related to my DPhil topic 8
1.4.2 Other papers contribution 9

2 Literature Review **10**

2.1 Soft Robots 10
2.1.1 Soft Actuators 11
2.1.2 Empirical Design Approaches 13
2.1.3 Applications 14
2.2 Simulation and Modelling Approaches for Soft Robot Design 15
2.2.1 Analytical Models 15
2.2.2 Finite Element and High-Fidelity Simulations 17
2.2.3 Data-Driven and Surrogate Models 18
2.2.4 Complementary and Emerging Modelling Approaches 19
2.2.5 Summary and Comparison 21
2.3 Design Optimisation of Soft Robots 22
2.3.1 Actuator-level Optimisation 22
2.3.2 Task-level Optimisation 26
2.4 Research Gap Analysis 30
2.4.1 Research Gap 1 31
2.4.2 Research Gap 2 31
2.4.3 Research Gap 3 32
2.4.4 Research Gap 4 32

3 A Simulation Toolbox for 3D-Printed Bellow SPAs 33
3.1 Positioning with respect to the state-of-the-art (SOTA) 33
3.1.1 Problem 33
3.1.2 Objective 34
3.1.3 Contributions 34
3.2 Assumptions, Validity Domain and Limitations 35
3.3 Contributed Paper 36
3.4 Supporting Information 43
3.4.1 Toolbox Repository 43
3.4.2 Supporting Video 43

4 An Optimisation Toolbox for Bellow SPAs in Shape Matching 44
4.1 Positioning with respect to the SOTA 44
4.1.1 Problem 44
4.1.2 Objective 45
4.1.3 Contributions 45
4.2 Assumptions, Validity Domain and Limitations 46
4.3 Contributed Paper 46
4.4 Supporting Information 62
4.4.1 Toolbox Repository 62
4.4.2 Supporting Video 62
4.4.3 Supplementary Data 62

5 An Optimisation Investigation for Bellow SPAs in Shape-Matching 73
5.1 Positioning with respect to the SOTA 73
5.1.1 Problem 73
5.1.2 Objective 74
5.1.3 Contributions 75
5.2 Assumptions, Validity Domain and Limitations 75
5.3 Contributed Paper 76
5.4 Supporting Information 84
5.4.1 Additional Experimental Results 84
5.4.2 Shape-Matching Optimisation: Low-Dimensional 85
5.4.3 Shape-Matching Optimisation: High-Dimensional 87

6 A Generalised Task-Driven Design Pipeline for Soft Actuators 91
6.1 Positioning with respect to the SOTA 91
6.1.1 Problem 91
6.1.2 Objective 92
6.1.3 Contributions 93
6.2 Assumptions, Validity Domain and Limitations 93
6.3 Contributed Paper 94
6.4 Supporting Information 112
6.4.1 Supporting Material 112
6.4.2 Supporting Video 121

7 Conclusion 122

Appendices

A List of Supporting Videos 127

B Copyright Permissions 128
B.1 Permissions Granted by Publishers 129
B.2 Permissions of other papers 132

References 134

List of Figures

2.1 Different soft actuators. **(A)** shows a fiber-reinforced actuator, reproduced with permission [7]. **(B)** shows a PneuNet bending actuator, reproduced with permission [8]. **(C)** shows actuators based on Shape memory alloys, reproduced with permission [9]. **(D)** shows dielectric elastomer actuators, reproduced with permission [10]. **(E)** shows a bellow SPA, reproduced with permission [11]. 11

2.2 Different simulation and modeling approaches for soft actuators. **(A)** uses a model based on beam theory, reproduced from [36]. **(B)** shows a Cosserat rod representation, reproduced from [37]. **(C)** uses an energy-based model, reproduced from [38]. **(D)** uses FEM, reproduced from [39]. **(E)** shows a scheme of a neural network. **(F)** shows a geometric model based on piecewise-constant curvature, reproduced from [40]. **(G)** shows a process of pseudo-rigid body model (PRBM), reproduced from [41]. **(H)** uses a voxel-based model, reproduced from [42]. **(I)** uses a model based on the material point method (MPM), reproduced from [43]. **(J)** shows a process of constructing model using SOFA, reproduced from [44]. 16

2.3 Gradient-based design optimisation for soft actuators. **(A)** uses an energy-based analytical modelling of a fibre-reinforced soft actuator, enabling extension, expansion, twisting, and bending for trajectory-matching optimisation using a nonlinear least-squares algorithm in MATLAB (lsqnonlin). The left panel shows the parameterised actuator designs, and the right panel illustrates the trajectory optimisation process, reproduced with permission [50]. **(B)** shows a topology optimisation of a SPA to achieve desired deformation behaviour based on FEM simulation, implemented using the Solid Isotropic Material with Penalisation approach and the Method of Moving Asymptotes algorithm. The upper panel illustrates the design domain and optimisation setup, while the lower panel shows the optimisation workflow, reproduced with permission [86]. 23

2.4 Gradient-free design optimisation for soft actuators. **(A)** presents a framework for the design and optimisation of a SPA using agenetic algorithm (GA) coupled with FEM simulation to achieve target deformation responses. The left panel illustrates the framework components, and the right panel outlines the optimisation workflow, reproduced with permission [58]. **(B)** shows a multi-objective design optimisation framework for a soft finger, where a surrogate artificial neural network (ANN)—trained on simulation data from simulation open framework architecture (SOFA)—is optimised using a genetic algorithm (NSGA-II) to achieve desired reaction force and centre-of-gravity alignment, reproduced with permission [89]. 24

2.5 Task-level optimisation frameworks built upon high-fidelity simulations. **(A)** shows a reinforcement learning (RL)-based design and control co-optimisation framework for soft robot crawlers, employing proper orthogonal decomposition (POD)-reduced FEM model for simulating physically consistent locomotion to optimise travel distance. The upper left panel depicts the use of POD for model reduction, the upper right shows the design space and representative simulated and real-world designs, and the lower panel outlines the joint optimisation process, reproduced with permission [99]. **(B)** presents a co-design method for a soft hand, utilising SOFA-based simulation to jointly optimise the morphology and control for object grasping with the particle filter optimisation method. The upper panel displays the design and control parameter spaces, the middle panel illustrates the simulation of grasping phases, and the lower panel outlines the optimisation workflow, reproduced with permission [98]. 27

2.6 Task-level optimisation frameworks using fast proxy simulations. **(A)** presents a benchmark platform for the co-optimisation of voxel-based soft robots modelled as 2D mass–spring systems, employing evolutionary and RL algorithms to optimise both morphology and control across multiple locomotion and manipulation tasks. The left panel provides an overview of the benchmark platform and its integration with co-design algorithms, and the right panel illustrates the diverse optimisation tasks enabled by the framework, reproduced with permission [102]. **(B)** illustrates a modelling and co-optimisation framework based on the MPM for a multi-link soft robotic arm to achieve target poses. The upper panel shows the simulation process using MPM, while the lower panel compares the resulting configurations under different stiffness settings: fixed stiffness at 300% and 100% of the initial Young's modulus (Poses I and II), and a co-optimised stiffness distribution (Pose III). Only the co-optimised design achieves the target configuration, reproduced with permission [43]. **(C)** shows a computational design synthesis approach for automated generation of virtual soft robots, where designs are represented by PRBM combined with RL-based optimisation for locomotion. The upper left panel displays the building blocks of the spatial grammar, the right panel presents the grammar rules used for design generation, and the lower left panel shows representative optimised designs, reproduced with permission [103]. 28

5.S1 Additional optimisation-algorithm results (set 2). Panel labels match those in Fig. 4. 84

5.S2 Additional optimisation-algorithm results (set 3). Panel labels match those in Fig. 4. 84

5.S3 $N = 1$. **(A)** Segmented arc. **(B)** Shapes designed with the optimised parameters at each optimisation level (colours correspond to **(C)**). **(C)** Objective values for the levels. 85

5.S4 $N = 2$. **(A)** Segmented arcs. **(B)** Shapes designed with the optimised parameters at each optimisation level (colours correspond to **(C)**). **(C)** Objective values for the levels. 86

5.S5 $N = 3$. **(A)** Segmented arcs. **(B)** Shapes designed with the optimised parameters at each optimisation level (colours correspond to **(C)**). **(C)** Objective values for the levels. 87

5.S6 $N = 5$. **(A)** Segmented arcs. **(B)** Shapes designed with the optimised parameters at each optimisation level (colours correspond to **(C)**). **(C)** Objective values for the levels. 88

5.S7 $N = 10$. **(A)** Segmented arcs. **(B)** Shapes designed with the optimised parameters at each optimisation level (colours correspond to **(C)**). **(C)** Objective values for the levels. 89
5.S8 $N = 15$. **(A)** Segmented arcs. **(B)** Shapes designed with the optimised parameters at each optimisation level (colours correspond to **(C)**). **(C)** Objective values for the levels. 90

B.1 Permissions from IEEE for reusing figures for thesis. 129
B.2 Permissions from Advanced Intelligent Systems licensed with CC BY 4.0 for reusing figures for thesis. 130
B.3 Permissions from the Association for Computing Machinery for reusing figures for thesis. 131
B.4 Permissions from the Proceedings of the National Academy of Sciences for reusing figures for thesis. 132
B.5 Permissions for papers via RightsLink. 132
B.6 Permissions for papers via Marketplace 133

Acronyms

ANN artificial neural network. x, 24

BO bayesian optimisation. 5, 24–26, 74, 75, 85

CAD computer aided design. 4

FEM finite element method. iv, ix, x, 3–5, 7, 16–18, 20, 21, 23–27, 29, 31–36, 45, 46, 74, 75, 85, 93, 123

GA genetic algorithm. x, 5, 24–26, 74, 75, 87

GUI graphical user interface. iv, 3, 4, 34

MPM material point method. ix, xi, 16, 19, 21, 28, 29

NN neural networks. 4, 5, 45, 46, 75, 87

PCC piecewise constant curvature. 19, 21, 45

POD proper orthogonal decomposition. x, 18, 27

PRBM pseudo-rigid body model. ix, xi, 5, 16, 19, 21, 28

PSO particle swarm optimisation. 24–26, 74

RL reinforcement learning. x, xi, 27, 28

SOFA simulation open framework architecture. x, 20, 21, 24, 27

SOTA state-of-the-art. vii, viii, 6, 30, 33, 44, 73, 91, 122

SPA soft pneumatic actuator. iv, v, ix, x, 2–7, 10–12, 15, 17, 23, 24, 33–35, 45, 46, 74, 75, 122

1 Introduction

Contents

1.1 Motivation **1**
1.2 Objective and our contributions **3**
1.3 Outline **6**
1.4 Publications **8**
1.4.1 Papers directly related to my DPhil topic 8
1.4.2 Other papers contribution 9

1.1 Motivation

Soft robots have gained increasing interest due to their ability to safely interact with uncertain environments and humans. Their compliant bodies enable behaviours that emerge from the interplay between morphology, material properties, and distributed actuation, reducing the need for precise external control. Unlike rigid robots—whose behaviour is primarily dictated by high-bandwidth control loops—soft robots frequently exploit body–environment coupling: shape change, passive deformation, and contact interactions contribute directly to task execution. This distinction is important not because soft robots lack control, but because their body carries part of the computation.

Given these advantages, soft robots have been explored for agriculture, food handling, surgery, and field applications where robustness and adaptable deformation are essential. Within the broader family of soft actuators, SPAs stand out for their simple operation, broad customisability, and ability to generate large deformations. In particular, bellow SPAs rely on macroscopic geometric deformations through corrugations rather than high material strain, making them compatible with 3D-printing processes. This is especially beneficial because current 3D-printable elastomers exhibit much lower elongation-at-break than silicone rubbers. The geometry-driven deformation of bellow SPAs enables efficient fabrication of customised actuators and systems, enhancing precision and repeatability.

Although advances in fabrication and materials have made soft actuators increasingly accessible, designing them for intended task applications remains fundamentally challenging. Their behaviour emerges from the coupled effects of geometry, materials, actuation, and environmental interaction, such that task performance is an emergent outcome rather than the result of any single design factor.

As a result, task-driven design naturally starts from task requirements—such as target deformation, force output, or interaction with the environment—and seeks actuator structures capable of realising these behaviours. In practice, however, this inverse mapping from task objectives to design parameters is highly nonlinear and unintuitive, making task-level performance difficult to predict from design choices. Though designers may anticipate qualitative tendencies, this difficulty becomes more pronounced as tasks demand richer deformation patterns or as actuator structures grow in complexity, causing the design space to expand beyond what can be explored through intuition or direct inspection. Once this inverse design step becomes necessary, the challenge shifts to how suitable designs are identified efficiently in practice. Furthermore, when actuator geometries or task contexts change, a design process developed for one setting may no longer apply, introducing the need for approaches that can adapt without being reconstructed from scratch.

Together, these considerations suggest that task-driven soft-actuator design benefits from systematic computational support, allowing designers to move from

high-level requirements toward manufacturable geometries in a principled way. The next section outlines the specific objectives of this research.

1.2 Objective and our contributions

The objective of this DPhil research is to develop computational methods that automate the task-driven design of soft actuators, enabling efficient and accurate transitions from high-level task objectives to viable actuator designs. Here, "efficient" refers to the computational resources required per design study, including both model evaluations and any upfront data generation, while "accurate" concerns the fidelity of model predictions with respect to experimentally measured actuator behaviour. Achieving this objective streamlines the process from task specification to actuator design, reducing reliance on human effort and delivering designs that would accomplish the target tasks. In doing so, the resulting design methods aim to remain sufficiently accurate to inform design decisions and sufficiently efficient to support iterative design within practical time and resource limits. To address this objective, several research questions were identified and investigated in this thesis.

Q1. How can we support the designers to translate their intuition to achieve certain soft actuator behaviour without iterative prototyping and testing?

Modelling and simulation have become essential for the automated design of soft actuators, enabling computational evaluation of design candidates and reducing reliance on repetitive physical prototyping. Dedicated and accessible tools allow designers to iteratively explore design parameters and assess performance efficiently. However, existing modelling tools for soft robots remain limited, facing a fundamental trade-off between computational efficiency and accuracy, and are often tailored to silicone-casting rather than 3D-printed fabrication methods. Therefore, this DPhil first develops a simulation toolbox for 3D-printed bellow SPAs that translates design intuition into predictive actuator models. The toolbox provides an interactive GUI for customising geometry and printing materials, combining a beam-theory-based model for rapid workspace estimation with high-fidelity FEM

simulations for accurate evaluation of deformation, blocked force, and stress–strain behaviour. This establishes a fully digital design workflow for bellow SPAs. Details of this work can be found in Chapter 3.

Q2. How can we address the challenges that arise when task complexity expands the design space beyond intuition-guided exploration?

The main limitation of the previous work was that the bellow SPA design was restricted to uniform bending, resulting in a limited design space and reduced capability for complex task-driven behaviours. As task requirements become more complex, relying on human intuition guided by simulation feedback is no longer sufficient, and efficient exploration of the large, nonlinear design space requires algorithmic optimisation with iterative evaluation. To overcome this limitation, the actuator design was extended with a modular structure incorporating internal constraints, enabling module-level customisation and interconnection to realise a wider range of deformation shapes beyond simple bending. Building on this modular foundation, an inverse design approach was introduced, in which shape matching is adopted as the task representation to provide a direct and interpretable geometric objective for relating task requirements to actuator configurations. To support inverse design, a two-step modelling approach was developed: a modular FEM for accurate shape estimation, and a neural networks (NN)-based surrogate model for fast iterative evaluation during optimisation. These components are integrated into an interactive, GUI-based toolbox that automatically converts target shapes into ready-to-print computer aided design (CAD) models. Further details of this work are presented in Chapter 4.

Q3. How do different combinations of modelling and optimisation choices trade off computational efficiency and design performance in soft actuator design optimisation?

Optimisation becomes central to task-driven soft actuator design under inverse design, yet different algorithms interact with computational models in distinct ways, leading to trade-offs between computational cost, exploration behaviour, and achievable design quality. While data-driven surrogate models enable fast

evaluation but require costly dataset generation, FEM-based simulations provide high accuracy at increasing computational expense as design complexity grows. To examine these trade-offs, this work systematically compares two representative global optimisation algorithms, bayesian optimisation (BO) and GA, paired with FEM and a NN-based surrogate model, respectively. Shape matching is used as a consistent and interpretable task objective to evaluate optimisation behaviour across different model–algorithm combinations. The combinations are benchmarked in terms of total computational cost and optimisation performance, including shape-matching accuracy and convergence behaviour. The results offer practical guidance on selecting suitable model–algorithm pairings based on design-space dimensionality and computational resource constraints. Further details of this work are presented in Chapter 5.

Q4. How can task-driven soft actuator design be supported in a reusable manner across parameterised actuator designs and task scenarios?

Although the previous chapters enable accurate and efficient design of bellow SPAs for specific tasks, the resulting modelling and optimisation pipelines remain tightly coupled to particular actuator designs and task scenarios. This reflects a broader limitation in task-driven soft robot design: models are often constructed for a fixed geometry, actuation mechanism, or task context, making them difficult to reuse when these conditions change. In soft robotic systems, this limitation is amplified by the strong coupling between actuation, large deformation, and environmental interaction. While high-fidelity continuum models such as FEM are broadly applicable across designs, their computational cost limits their use in repeated task-level evaluation. In contrast, reduced-order and surrogate models offer efficiency but are typically valid only for specific designs or operating regimes. To address this trade-off, the final part of this thesis develops a unified task-driven design pipeline that integrates multiple modelling layers, including FEM, PRBM, and surrogate representations. This pipeline supports task-level evaluation under interaction and loading, and enables actuator designs across different geometries and actuation mechanisms to be

optimised directly with respect to task outcomes. The development and validation of this task-driven design pipeline are presented in Chapter 6.

1.3 Outline

This thesis consists of seven chapters: an introduction, a literature review, four core chapters presenting the scientific contributions, and a conclusion. The main chapters follow an integrated format, with each corresponding to a published paper. To provide additional context and improve readability, each chapter begins by positioning the work with respect to the SOTA, followed by the research objective, main contributions, and a discussion of the assumptions, validity domain, and limitations. The remainder of the chapter presents the published paper containing the technical details.

Chapter 1: Introduction

The motivation of the thesis is stated followed by a list of research questions identified and our contributions toward them. A publication list is attached as well.

Chapter 2: Literature Review

This chapter reviews prior research relevant to the task-driven design of soft robots. It begins with an overview of soft actuator design, their traditional empirically-driven design approaches, and the need for computational automated methods motivated by various applications. The chapter then systematically reviews computational design methodologies, structured into two core components: (1) modelling approaches for predicting soft actuator behaviour; and (2) optimisation and task-driven design methods leveraging such models to achieve task requirements.

Chapter 3: A Simulation Toolbox for 3D-Printed Bellow SPAs

Here we present a conference paper titled "A Simulation-Based Toolbox to Expedite the Digital Design of Bellow Soft Pneumatic Actuators" published on *2022 IEEE 5th International Conference on Soft Robotics (RoboSoft).* The work has been presented

in an oral session at the IEEE 5th International Conference on Soft Robotics in 2022, in Edinburgh, UK. The work introduces the first dedicated simulation toolbox for 3D-printed bellow SPAs, that allows users to customise geometric parameters, select 3D printing materials, and simulate actuator performance using both a fast analytical model for instant predictions and high-fidelity FEM simulations for accurate results. Therefore, it provides a digital workflow that allows designers to translate their intuition into manufacturable actuator designs.

Chapter 4: An Optimisation Toolbox for Bellow SPAs in Shape Matching

Here we present a journal paper titled "SPADA: A Toolbox of Designing Soft Pneumatic Actuators for Shape Matching Based on Surrogate Modeling" published in *Robotics Reports*. This work has been presented in a workshop at the IEEE International Conference on Robotics and Automation in 2023, in London, UK. This work introduces a modular actuator representation for bellow SPAs combined with surrogate modelling to enable task-driven inverse design beyond intuition-guided exploration. Building on this formulation, SPADA is developed as an end-to-end shape-matching design toolbox that automatically maps target deformation profiles to manufacturable actuator designs.

Chapter 5: An Optimisation Investigation for Bellow SPAs in Shape Matching

Here we present a conference paper titled "Design Optimization for Bellow Soft Pneumatic Actuators in Shape-Matching" published on *2023 IEEE 6th International Conference on Soft Robotics (RoboSoft)*. The work has been presented in a poster session at the IEEE 6th International Conference on Soft Robotics in 2023, in Singapore. This work systematically evaluates optimisation strategies for bellow SPAs in shape matching, comparing two representative model–algorithm combinations. By analysing trade-offs between computational cost, optimisation performance, and achieved design quality, the study provides practical guidance for selecting suitable model–algorithm pairs in scalable actuator design.

Chapter 6: A Generalised Task-Driven Design Pipeline for Soft Actuators

Here we present a journal paper titled "Generalized Task-Driven Design of Soft Robots via Reduced-Order Finite Element Method-Based Surrogate Modeling", published in *Advanced Intelligent Systems*. This work presents a unified task-driven design pipeline that supports generalisation across parameterised actuator designs and task scenarios by combining high-fidelity finite element modelling, reduced-order representations, and design-conditioned surrogate models. By embedding the resulting models into a rigid-body simulation environment, the proposed approach enables efficient task-level simulation and optimisation under contact, gravity, and external loading. The pipeline is validated across multiple actuator types and representative task applications.

Chapter 7: Conclusions

In this chapter, we summarise the contributions of this thesis, reflect on the key lessons learned, and discuss potential directions for future research.

1.4 Publications

1.4.1 Papers directly related to my DPhil topic

- Y. Yao, L. He and P. Maiolino, A Simulation-Based Toolbox to Expedite the Digital Design of Bellow Soft Pneumatic Actuators, *2022 IEEE 5th International Conference on Soft Robotics (RoboSoft)*, Edinburgh, United Kingdom, 2022, pp. 29-34.

- Y. Yao, Y. Chen, L. He and P. Maiolino, Design Optimisation for Bellow Soft Pneumatic Actuators in Shape-Matching, *2023 IEEE 6th International Conference on Soft Robotics (RoboSoft)*, Singapore, 2023.

- Y. Yao, L. He and P. Maiolino, SPADA: A Toolbox of Designing Soft Pneumatic Actuators for Shape Matching Based on Surrogate Modeling, *Robotics Reports*, vol. 2, no. 1, pp. 1–14, 2024.

- Y. Yao, D. Howard and P. Maiolino, Generalized Task-Driven Design of Soft Robots via Reduced-Order FEM-based Surrogate Modeling *Advanced Intelligent Systems*, 2026, 0, e70406.

1.4.2 Other papers contribution

- S. Wang, L. He, Y. Yao, C. Liu, and P. Maiolino, 2023. A Toolbox for Designing 3D-printing-ready Pneumatic Circuits for Controlling Soft Robots. *Advanced Intelligent Systems*, 5: 2300394.

- N. E. Veronese, A. Albini, Y. Yao, P. Rocco, P. Maiolino, F. Forni. On Feedback Error Learning for Adaptive Soft Robot Control, *2024 IEEE 7th International Conference on Soft Robotics (RoboSoft)*, San Diego, CA, USA, 2024, pp. 793-799.

- Y. Yao, M. Westermann, M. Pontin, A. Albini, and P. Maiolino, JAMMit! Monolithic 3D-Printing of a Bead Jamming Soft Pneumatic Arm, *2025 IEEE 7th International Conference on Soft Robotics (RoboSoft)*, Lausanne, Switzerland, 2025, pp. 1-6

- S. Zhong, Y. Yao, P. Maiolino, 'An End-to-End Framework for Modelling Pneumatic Soft Robots Based on Differentiable Finite Element Methods, *IEEE 7th Robotics and Automation Letter* vol. 10, no. 12, pp. 13003-13010, Dec. 2025.

2
Literature Review

This chapter reviews prior research relevant to the task-driven design of soft robots, structured into three main sections. The first section provides an overview of soft robot design, with a focus on the growing adoption of SPAs, particularly bellow SPAs, highlights the limitations of traditional empirically-driven design approaches, and motivates the need for computational methodologies to enable systematic and application-oriented soft robot design. This discussion establishes the motivation and significance of this DPhil research. The second section reviews modelling approaches used to describe soft actuator behaviour, which vary in fidelity, computational efficiency, and scope of implementation. The third section focuses on design optimisation frameworks that leverage these models to achieve task-level performance objectives. These two sections contextualise the research contributions of this DPhil in the research field.

2.1 Soft Robots

Conventional rigid robots excel in precision, repeatability, and high load capacity, but their rigidity often limits safe interaction with humans and adaptability to unstructured environments. Soft robots, by contrast, leverage material compliance to achieve safe interaction, adaptability, and resilience in tasks where rigid robots

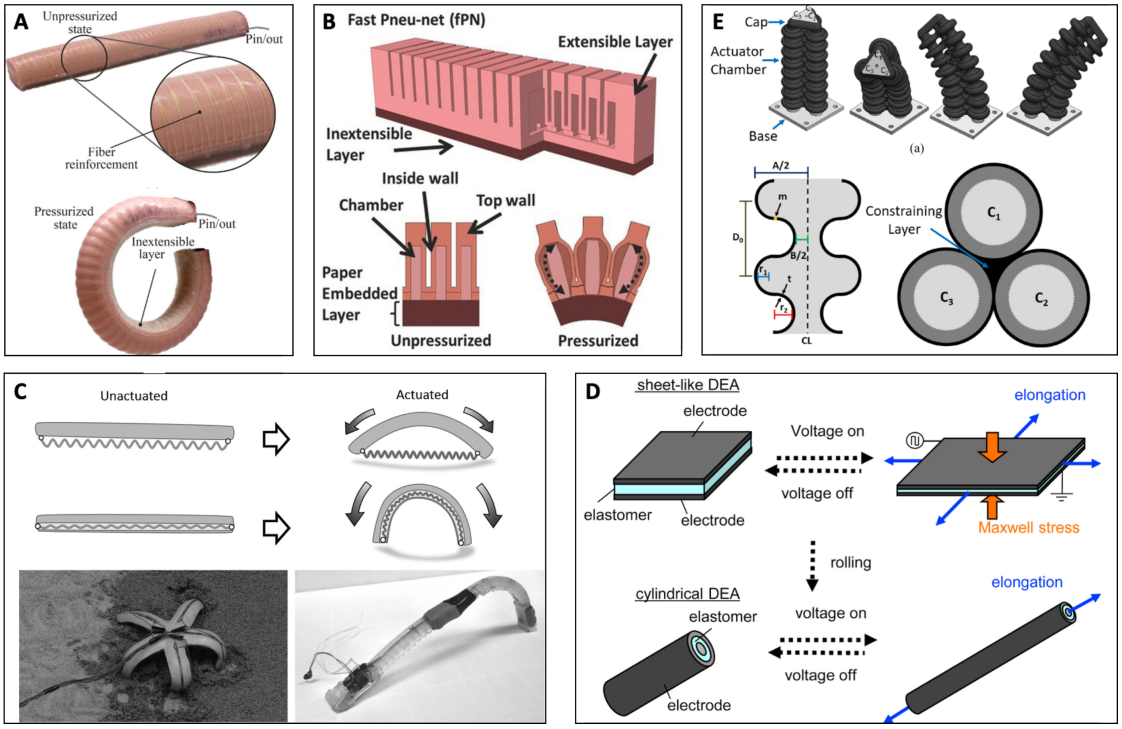


Figure 2.1: Different soft actuators. **(A)** shows a fiber-reinforced actuator, reproduced with permission [7]. **(B)** shows a PneuNet bending actuator, reproduced with permission [8]. **(C)** shows actuators based on Shape memory alloys, reproduced with permission [9]. **(D)** shows dielectric elastomer actuators, reproduced with permission [10]. **(E)** shows a bellow SPA, reproduced with permission [11].

struggle [1–3]. This motivates the exploration of diverse soft actuator designs that can expand the capabilities of robots into areas such as manipulation of delicate objects, locomotion in constrained spaces, and medical interventions [4–6]. This section provides an overview of the soft actuators used by these robots, highlights the limitations imposed by the traditional empirical design, and summarises their applications in different fields.

2.1.1 Soft Actuators

Soft actuators play a central role in soft robotic systems by generating motion and force, and by determining how they interact with the environment [1]. Representative examples are illustrated in Fig. 2.1. Early developments focused on pneumatic artificial muscles (PAMs), composed of elastomeric tube and braid fiber mesh, which mimic the contraction of biological muscles [12, 13]. PAMs, which later evolved into

the broader class of fiber-reinforced soft actuators [14], can generate large forces relative to their weight and are easy to control via compressed air, but their range of motion is limited, and their fabrication requires careful assembly of the mesh, which complicates customisation and hinders manufacturing precision.

To enable more versatile deformations, pneumatic networks (PneuNets) were introduced, consisting of patterned chambers within elastomer bodies that expand asymmetrically under pressurisation [8, 15]. PneuNets provide large bending motions and have been widely adopted in soft grippers and manipulators [16]. However, they typically rely on mould-based casting methods, which limit geometric freedom and make scaling up for complex or customised tasks challenging.

Other actuation mechanisms have also been explored. Electro-active polymers (EAPs) offer lightweight and compact actuation with high bandwidth, but often require high voltages and produce relatively small strains [17]. Shape memory alloys (SMAs) can achieve significant contraction and compact integration, but their slow response times, hysteresis, and energy inefficiency hinder broader adoption [18]. Magnetorheological elastomers (MREs) [19] and dielectric elastomers [20] expand the range of actuation principles but introduce complexities in material preparation and control electronics.

Bellow-type or bellow-inspired soft pneumatic actuators (SPAs) [11, 21] represent a category of SPAs that leverages the corrugated architecture to transfer small, distributed material strains into large macroscopic structural deformations via folding and unfolding of the walls, reducing dependence on material stretch. This approach is particularly compatible with additive manufacturing, where printable elastomers typically have lower elongation at break than cast silicones. Therefore, bellow SPAs offer a large range of motion and are compatible with 3D printing, enabling rapid prototyping and reproducible fabrication, making them a suitable platform for investigating task-driven design challenges. Yet, the effectiveness of such actuators ultimately depends on the design methodology. Early soft actuator development was dominated by empirical, intuition-based practices, which are reviewed in the next subsection.

2.1.2 Empirical Design Approaches

The initial development of soft actuators was largely driven by empirical and intuition-based practices. The most basic soft robots were typically created to achieve a single, specific behaviour, which could be directly translated into physical requirements—for instance, a target bending angle or output force [14, 22–25]—and realised through simple actuator structures. This process inherently relied on manual trial-and-error prototyping, where actuator geometry, material selection, or parametric dimensions were iteratively adjusted until the desired motion or force was observed. Such approaches enabled rapid proof-of-concept demonstrations of new actuation principles, but they were largely restricted to simple behaviours and quickly became infeasible as more was demanded of the robot.

To design more sophisticated systems, researchers frequently drew inspiration from biology, mimicking soft animals and anatomical features such as snakes [26], octopus [27], human hands [28], etc. Bioinspired heuristics offered a wealth of ideas for morphology and motion patterns, thereby accelerating the exploration of new robotic capabilities. However, these methods remained fundamentally empirical: while biological analogies suggested candidate forms and behaviours, actual task performance depended on nonlinear interactions among geometry, material properties, actuation, and the task environment, making it difficult to anticipate how design choices translate into task-level behaviour. As a result, the resulting designs were often suboptimal and case-dependent, with limited generalisability to new tasks or conditions.

Overall, while empirical design approaches laid the foundation for the first generation of soft robots, they also underscored the need for more systematic methodologies to bridge actuator design with task requirements. This limitation becomes even clearer when examining the early applications of soft robots, which are reviewed in the next subsection.

2.1.3 Applications

The emergence of soft actuators rapidly stimulated a wide range of applications, reflecting the strong interest in exploiting compliance and adaptability for tasks where rigid robots are limited. Research has been particularly active in domains such as manipulation, locomotion, and wearable or biomedical devices. These examples highlight how task requirements have motivated diverse soft robot designs, which are reviewed below.

Soft grippers and manipulators are the most widely explored applications, driven by the demand for safe and adaptive grasping [4]. In many cases, object properties such as size and weight are translated into actuator requirements (e.g., deformation range, blocked force), which are then realised through iterative adjustment of geometry and materials [16, 29–31]. This strategy has enabled a rich variety of gripper designs capable of handling delicate or irregular objects.

In locomotion and exploration, soft robots frequently mimic organisms such as worms, snakes, and caterpillars, reproducing their body shapes and gait patterns to achieve crawling, burrowing, or swimming [26, 32, 33]. These bioinspired strategies have demonstrated compelling behaviours in controlled settings and highlight the ability of soft bodies to traverse unstructured environments.

Wearable and biomedical devices provide another major motivation, where soft actuators enable safe physical interaction with the human body [6]. Applications such as exosuits or continuum surgical tools highlight the importance of compliance and adaptability [31, 34, 35]. In these cases, actuator specifications are often derived directly from the intended task—for example, the joint torque needed to assist limb motion, or the bending range required for a surgical tool—followed by further adjustments to meet these targets.

Collectively, these applications demonstrate both the growing interest in task-driven soft robot design and the inherent limitations of empirical strategies. While this approach has enabled a rich diversity of proof-of-concept demonstrations, it provides only limited guidance beyond the specific conditions for which designs are tuned. Moreover, the reliance on iterative prototyping and indirect task-to-actuator

mappings makes the process inefficient and yields designs with limited transferability, particularly as task requirements grow more complex. These limitations underscore the need for computational design methodologies, which aim to provide more systematic and predictive pathways from task objectives to actuator designs. Such methods integrate physical models of soft actuator behaviour with algorithmic optimisation to reduce inefficiency and enhance generalisability across task settings and actuator configurations. The recent advances in this direction are reviewed in the following two sections.

2.2 Simulation and Modelling Approaches for Soft Robot Design

The previous discussion has highlighted the importance of computational methods to accelerate and expand the design exploration of soft actuators toward meeting task objectives. Computational design of soft robots fundamentally relies on models that capture the nonlinear relationship between actuator geometry, material properties, loading, and resulting deformations. This section reviews the wide spectrum of modelling approaches that have been developed, differing in fidelity, computational cost, and applicability to bellow-SPAs. Representative examples of these approaches are illustrated in Fig. 2.2.

2.2.1 Analytical Models

Early studies often employed a variety of analytical models, such as beam theory [11, 45, 46], rod model [47–49], and energy-based methods [7, 50, 51]. These models provide explicit relations between actuation, deformation, and force for simplified geometries, offering valuable interpretability and computational efficiency. These analytical formulations were particularly valuable in the early development of soft actuators, as they provided fast evaluations for design intuition and parameter studies. However, they typically rely on simplified geometric representations and idealised loading assumptions. For instance, beam and rod formulations assume slender or uniform cross-sections and approximate distributed actuation as averaged

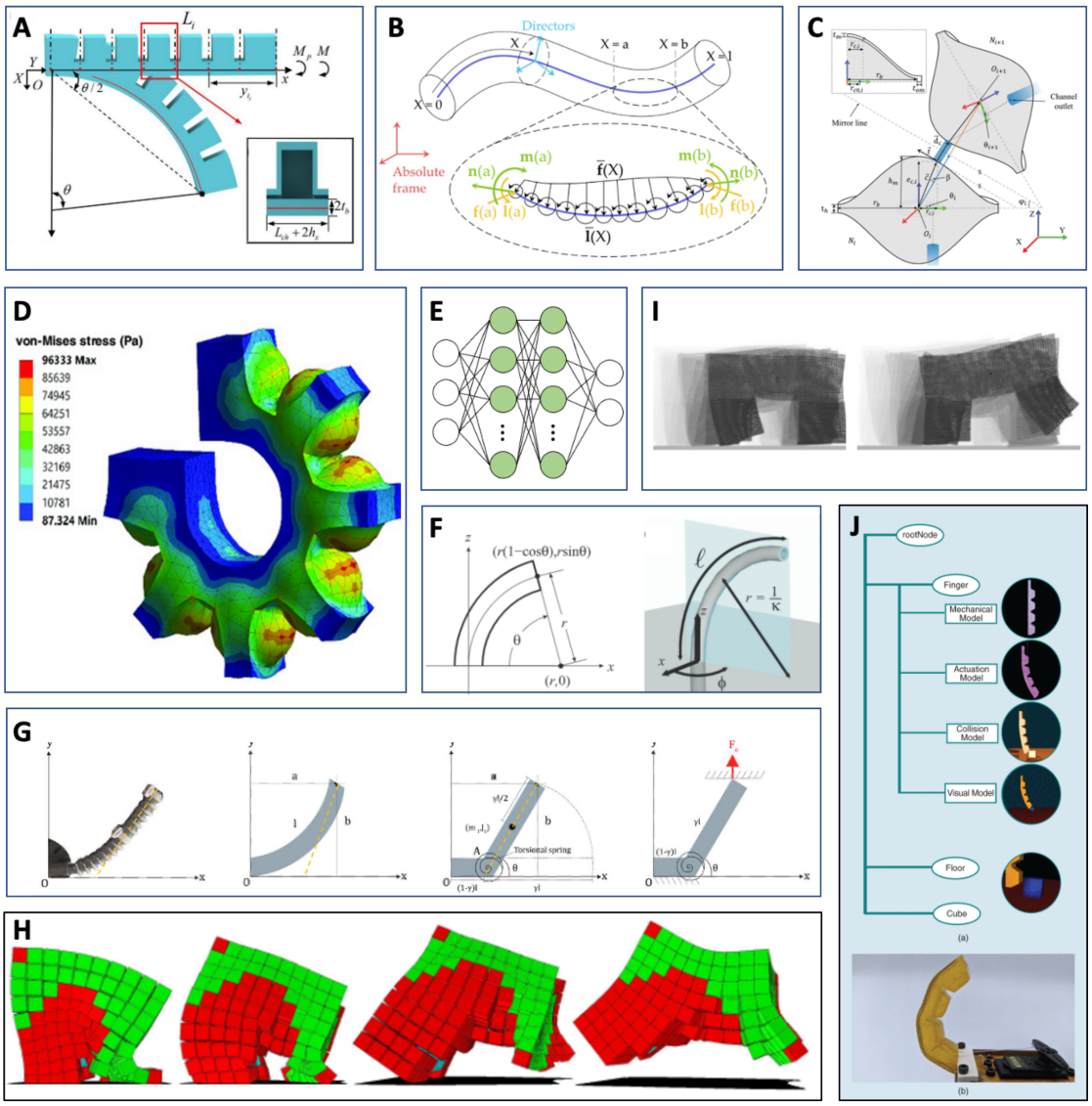


Figure 2.2: Different simulation and modeling approaches for soft actuators. **(A)** uses a model based on beam theory, reproduced from [36]. **(B)** shows a Cosserat rod representation, reproduced from [37]. **(C)** uses an energy-based model, reproduced from [38]. **(D)** uses FEM, reproduced from [39]. **(E)** shows a scheme of a neural network. **(F)** shows a geometric model based on piecewise-constant curvature, reproduced from [40]. **(G)** shows a process of PRBM, reproduced from [41]. **(H)** uses a voxel-based model, reproduced from [42]. **(I)** uses a model based on the MPM, reproduced from [43]. **(J)** shows a process of constructing model using SOFA, reproduced from [44].

axial loads, which limits their applicability to actuators with complex corrugated geometries such as bellows [36, 52]. Similarly, energy-based analytical approaches, though effective for fibre-reinforced and small-to-moderate deformations [50, 51], require predefined deformation modes and geometric simplifications that cannot

fully capture the multidirectional expansion behaviour characteristic of bellow-SPAs [38]. Consequently, the predictive accuracy of these analytical methods diminishes when applied to actuators exhibiting spatially varying deformation, complex 3D geometries, or significant external effects such as gravity and contact interactions that are intrinsic to real-world tasks.

2.2.2 Finite Element and High-Fidelity Simulations

As analytical models struggle to generalise beyond simplified geometries and loading assumptions, the community progressively turned to high-fidelity numerical simulation for predictive analysis. Among these, the finite element method (FEM) has become the standard tool for modelling soft actuators [39, 53–55]. By discretising the structure into finite elements and solving the governing equations in an element-wise manner, FEM accurately captures nonlinear material behaviour, large deformations, distributed actuation, and complex boundary conditions such as contact and gravity. This enables quantitative prediction across a wide range of geometries and loading conditions. However, practical challenges remain significant. FEM simulations are computationally expensive, with runtime ranging from tens of seconds to several minutes per evaluation depending on model complexity, which limits their usability for iterative design or optimisation [56–58]. Moreover, despite the availability of commercial solvers such as COMSOL Multiphysics® [59], ANSYS Mechanical™ [60], and Abaqus® [61], effective use of FEM requires specialised expertise and deep understanding of constitutive modelling and numerical stability—particularly when dealing with highly nonlinear materials or large-deformation regimes [39, 53]. Convergence difficulties also increase rapidly as geometric or material complexity grows, making simulation of very soft, highly deformable actuators such as bellow-SPAs especially demanding. Consequently, most FEM-based studies focus on narrow design spaces or fixed geometries, which constrains the ability to perform customised design exploration beyond predefined parameter ranges, limiting their usefulness when designers wish to investigate new configurations or task-specific requirements [57].

Although FEM serves as a general-purpose and physically grounded modelling framework, its high computational demand and operational complexity have motivated the exploration of alternative representations that maintain predictive fidelity while enabling efficient computation.

2.2.3 Data-Driven and Surrogate Models

In parallel with physics-based approaches, an alternative paradigm has emerged in which actuator behaviour is inferred directly from data rather than derived from first-principles equations. These data-driven or surrogate models learn input–output mappings—such as pressure–deformation relationships—from datasets obtained through experiments or simulations [37, 62]. Early implementations employed simple regression or polynomial fitting to approximate actuator responses, while more recent efforts have adopted reduced-order techniques such as Proper Orthogonal Decomposition (POD) or machine-learning-based surrogates—particularly neural networks and Gaussian processes—to capture nonlinear and coupled effects [63–65].

These methods enable rapid evaluation of actuator behaviours such as deformation and force responses, offering substantial acceleration in design and control tasks. However, their predictive performance depends strongly on the quality and coverage of the training data, making them sensitive to underrepresented operating conditions and limiting their extrapolation capability beyond the sampled design space. Beyond data coverage, purely data-driven models typically lack explicit physical constraints that enforce consistency with underlying mechanics, making it difficult to maintain coherent behaviour when the same actuator is evaluated under different task formulations or interaction conditions.

Despite these limitations, data-driven modelling has become a valuable complement to entirely physics-based frameworks, forming a crucial foundation for computational design methodologies that integrate simulation, optimisation, and learning-based prediction.

2.2.4 Complementary and Emerging Modelling Approaches

Beyond the aforementioned modelling paradigms, several complementary approaches have also been explored to balance fidelity, computational efficiency, and model accessibility.

Geometric models describe actuator deformation using function-based or piecewise constant curvature (PCC) curves [66, 67]. They provide fast and intuitive kinematic estimation and are often integrated with other models for control and planning. However, their reliance on simplified curvature assumptions and the neglect of external disturbances limit their applicability in task-level simulations where actuator–environment interactions play a significant role.

PRBMs approximate flexible structures with rigid links connected by compliant joints [66, 68, 69], offering lightweight computation and compatibility with rigid-body simulators such as PyBullet [70] and MuJoCo [71]. While effective for task- and environment-level simulations, they require dedicated parameterisation of joint stiffness to represent distributed actuation and nonlinear deformation—a requirement that is difficult to satisfy for bellow-SPAs, where actuation, deformation, and external interactions are strongly coupled [41].

Voxel-based models [42, 72, 73] approximate soft bodies using mass–spring or lumped-parameter networks, where actuation is emulated by local voxel expansion or contraction. These models provide efficient and intuitive implementations for approximate kinematic and dynamic behaviour, and are often integrated with task-level simulations [74]. However, accurately capturing complex geometries or continuous material distributions requires dense voxelisation, which would significantly increase computation. Furthermore, mapping voxel-based actuation to real pneumatic behaviour remains challenging, limiting their their fidelity for precise actuator design [75].

The MPM [76, 77] provides a continuum-based formulation capable of handling large deformations, contact, and topological changes, making it attractive for highly deformable or impact-rich scenarios [43]. Nevertheless, accurately approximating

the corrugated geometries of bellow-SPAs requires fine particle discretisation and small time steps, leading to high computational cost and stability challenges.

Differentiable simulation [78, 79] has recently been investigated as an approach to enable end-to-end, gradient-based optimisation for soft robot design and control [80]. However, its applicability is often constrained by the requirement that both the simulation process and task objectives be fully differentiable, which is difficult to satisfy for many soft-robot tasks involving contact, interaction, and large deformations, where non-smooth physical events naturally arise. More generally, adopting differentiable simulation in a task-driven design context typically entails modelling the complete physical process at the task level, encompassing full geometry, actuation, deformation, and environmental interaction. This leads to a fundamental trade-off between model fidelity and computational cost, as higher-fidelity representations rapidly become expensive when embedded within iterative design or optimisation loops. As a result, existing approaches often rely on simplified models [81, 82] or specialised computational acceleration techniques [83] to maintain tractability.

In parallel with these developments, open-source dedicated simulation frameworks for soft robotics, such as SOFA, provide integrated environments for implementing and coupling multiple modelling paradigms within a common computational platform [44, 84, 85]. This enables mechanical modelling, FEM solvers, collision handling, and control modules to be combined within a single simulation workflow. However, most implementations rely on simplified meshes or linear material models to maintain real-time performance, compromising simulation accuracy. Moreover, SOFA remains code-oriented, requiring familiarity with programming and physical modelling. Consequently, setting up complex simulation environments or boundary conditions is generally less intuitive than in commercial FEM software.

While modelling establishes the foundation for predicting actuator behaviour, optimisation defines how these models are used to search for designs that meet performance objectives. The next section reviews optimisation strategies that employ these models for actuator- and task-level design exploration.

2.2.5 Summary and Comparison

This section presents Table 2.1, summarises the main modelling approaches reviewed in this section, highlighting their respective advantages, limitations, and typical use cases.

The literature indicates that modelling approaches for soft actuators involve fundamental trade-offs between physical fidelity, computational efficiency, and flexibility with respect to changing designs and task requirements. High-fidelity simulations provide accurate physical grounding but are costly to deploy repeatedly, while simplified or data-driven models improve efficiency at the expense of generality or interpretability. These trade-offs suggests no single modelling approach is sufficient to support task-driven design across varying actuator configurations and task contexts, and motivates the practical design workflows to be combining different modelling abstractions to balance these competing requirements under realistic computational constraints.

Table 2.1: Comparison of representative simulation and modelling approaches for soft robots.

Approach	Advantages	Limitations
Analytical Models [7, 11, 45–51]	Physically interpretable; fast computation.	Simplified geometry and loading assumptions; limited to simple deformation or interactions.
FEM [39, 53–58]	High fidelity; handles nonlinear material and large deformation; broadly validated.	Computationally expensive; requires solver expertise; convergence issues with complex models.
Data-driven/Surrogate Models [37, 62–65]	Fast evaluation; broadly applicable.	Strongly data-dependent; low physical interpretability.
Geometric Models (e.g., PCC, function-based) [66, 67]	Simple and efficient; useful for kinematic estimation and control.	Inaccurate when unexpected external forces/contact are applied.
PRBMs [41, 66, 68–71]	Lightweight; compatible with rigid-body simulators; suitable for diverse task simulations.	Require joint stiffness parameterisation; limited for distributed actuation and nonlinear deformation.
Voxel-based Models [42, 72–75]	Intuitive representation; readily integrated with task-level simulations.	Dense voxelisation needed for complex shapes; limited physical realism and manufacturability.
MPM [43, 76, 77]	Handles large deformation and contact.	Computationally expensive and stability challenges for complex models.
Differentiable FEM [78–83]	Supports gradient-based optimisation.	Solver convergence and computational overhead limit applicability to highly nonlinear cases.
SOFA [44, 84, 85]	Open-source dedicated simulation framework integrating multiple modelling paradigms	Simplified meshes for real-time use; code-oriented setup.

These modelling approaches provide the foundation for the design optimisation strategies discussed in the next section.

2.3 Design Optimisation of Soft Robots

Building upon the modelling foundations discussed above, design optimisation in soft robotics concerns how actuator geometries, materials, and actuation parameters can be systematically identified to achieve desired behaviours or task performance under practical constraints. In this context, optimisation extends beyond the choice of numerical algorithms and should be understood as part of a broader design pipeline that couples a chosen parameterisation, a modelling approach for performance evaluation, and a search strategy. In practice, limitations in modelling capability and computational resources have strongly shaped how optimisation has been formulated, influencing both the level at which objectives are defined and the types of optimisation strategies that can be applied in practice. This section reviews how optimisation has been formulated in soft robot design under different modelling and computational constraints, including actuator-level approaches based on proxy performance metrics and task-level approaches that evaluate design performance directly within simulated task contexts.

2.3.1 Actuator-level Optimisation

Early optimisation-based approaches to soft actuator design predominantly operated at the actuator level. Owing to the limited availability and high computational cost of task-level simulation, task requirements were typically translated by designers into actuator-level performance metrics, such as target deformation profiles, bending angles, stiffness characteristics, or blocked force. Optimisation was then performed with respect to these proxy metrics, under the assumption that improvements at the actuator level would lead to satisfactory task performance.

Gradient-based optimisation methods rely on explicit sensitivity information between design parameters and performance objectives, enabling efficient convergence when such information is available. They are commonly applied when analytical

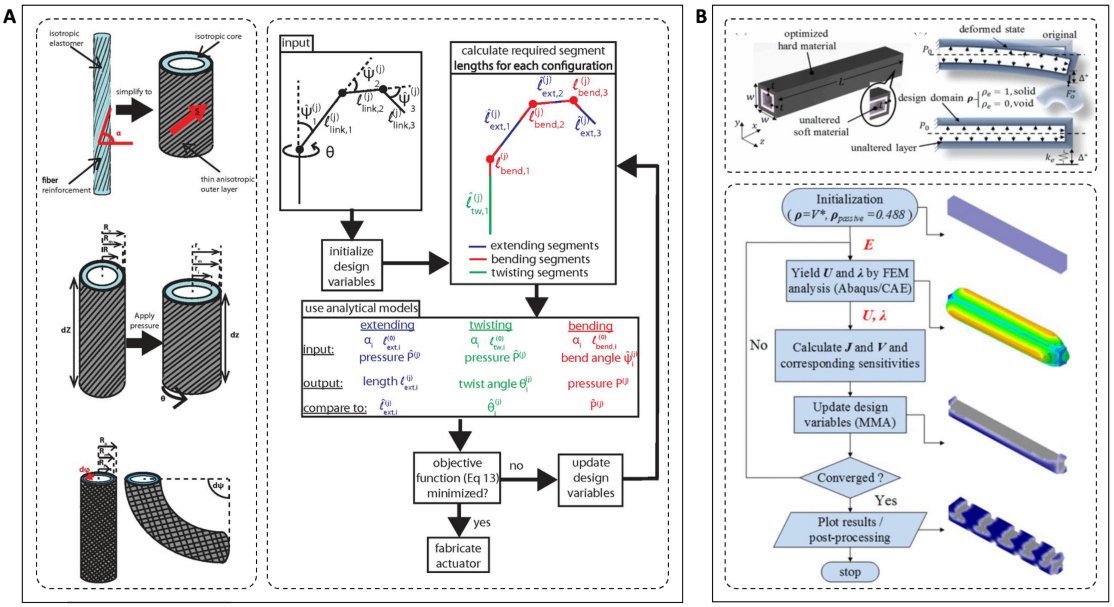


Figure 2.3: Gradient-based design optimisation for soft actuators. **(A)** uses an energy-based analytical modelling of a fibre-reinforced soft actuator, enabling extension, expansion, twisting, and bending for trajectory-matching optimisation using a nonlinear least-squares algorithm in MATLAB (lsqnonlin). The left panel shows the parameterised actuator designs, and the right panel illustrates the trajectory optimisation process, reproduced with permission [50]. **(B)** shows a topology optimisation of a SPA to achieve desired deformation behaviour based on FEM simulation, implemented using the Solid Isotropic Material with Penalisation approach and the Method of Moving Asymptotes algorithm. The upper panel illustrates the design domain and optimisation setup, while the lower panel shows the optimisation workflow, reproduced with permission [86].

models or differentiable simulation frameworks are available, as gradients can be directly computed via analytical derivation or automatic differentiation. Early examples combined analytical models with gradient-based solvers to refine actuator geometries for prescribed motions or deformation profiles [38, 50, 87]. Similarly, topology optimisation frameworks have been applied to soft structures to determine material layouts that maximise deformation efficiency under pressure loading [86, 88]. Representative examples of both cases are shown in Fig. 2.3. While gradient-based methods offer rapid convergence, their applicability is constrained by the need for differentiable formulations and relatively low-dimensional design spaces. For soft actuators with highly nonlinear material behaviour, complex geometries, or strong geometric–material coupling, the resulting optimisation landscapes are often nonconvex, causing gradient-based methods to converge to local optima and

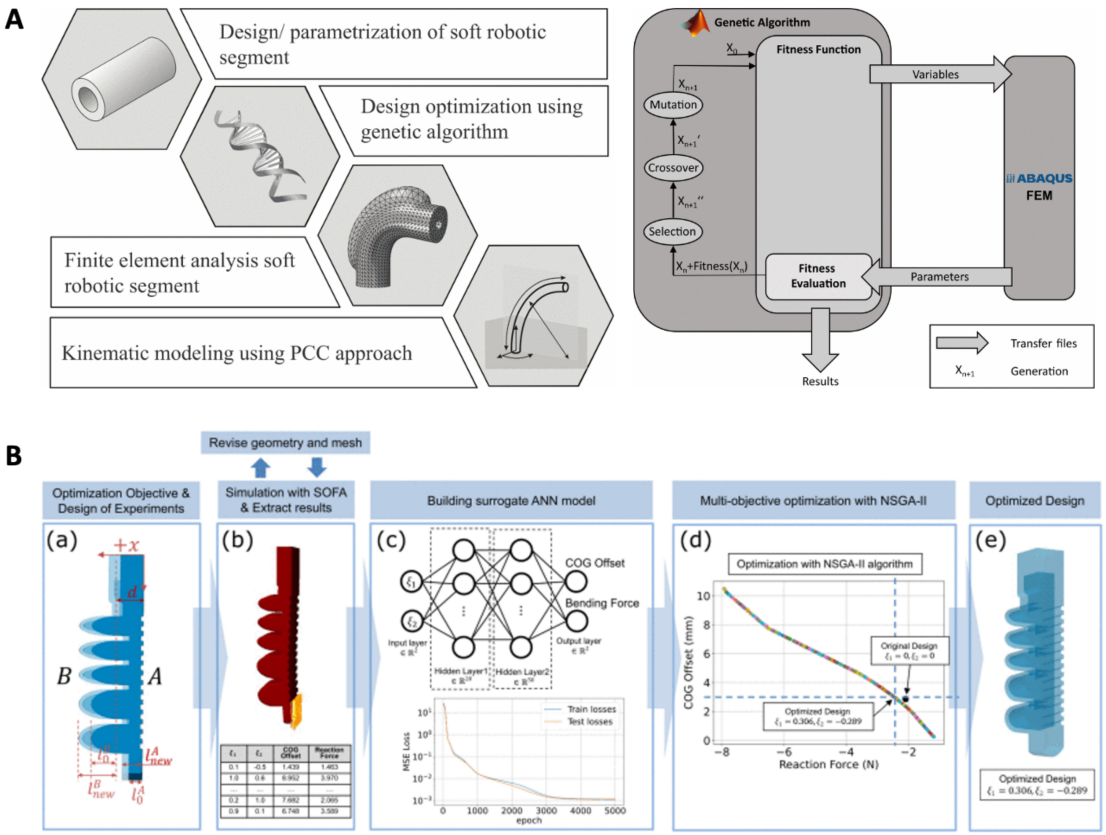


Figure 2.4: Gradient-free design optimisation for soft actuators. **(A)** presents a framework for the design and optimisation of a SPA using aGA coupled with FEM simulation to achieve target deformation responses. The left panel illustrates the framework components, and the right panel outlines the optimisation workflow, reproduced with permission [58]. **(B)** shows a multi-objective design optimisation framework for a soft finger, where a surrogate ANN—trained on simulation data from SOFA—is optimised using a genetic algorithm (NSGA-II) to achieve desired reaction force and centre-of-gravity alignment, reproduced with permission [89].

limiting their effectiveness for realistic 3D designs such as bellow-SPAs.

When explicit sensitivity information is unavailable or impractical to compute, actuator-level design optimisation is commonly carried out using gradient-free algorithms, such as GA, particle swarm optimisation (PSO), and BO [58, 90, 91]. These methods treat the performance evaluation as a black-box process, operating directly on outputs from numerical models—most commonly high-fidelity simulations such as FEM—without requiring differentiability, enabling the optimisation of actuator geometry, material, or actuation parameters for desired deformation or force responses. Illustrative frameworks employing gradient-free optimisation

methods are presented in Fig. 2.4. While capable of handling complex, non-convex design landscapes, their use of repeated FEM evaluations makes the process computationally demanding, with each simulation typically taking from tens of seconds to dozens of minutes depending on model complexity. This computational burden restricts exploration to limited parameter sets and narrowly defined design spaces.

To accelerate the process, surrogate models trained from pre-computed FEM or experimental data have been introduced [89, 92]. These models substantially reduce evaluation time while retaining approximate predictive fidelity. However, as these datasets are typically constructed by varying a limited set of geometric or actuation parameters and are trained to reproduce specific actuator-level responses, the resulting surrogate models are inherently tied to a fixed design parameterisation and task representation.

The preceding review reveals that inverse soft actuator design has been investigated through a range of modelling approaches and optimisation strategies. Rather than providing an exhaustive survey, Table 2.2 highlights representative model–optimisation combinations that have been adopted in the literature.

These representative combinations also illustrate that the computational characteristics of actuator-level optimisation depend jointly on the underlying modelling approach and the optimisation strategy employed. In particular, the overall optimisation efficiency depends strongly on (i) the model choice (direct FEM evaluations versus pre-trained surrogates), (ii) the algorithmic regime (sample-efficient BO versus heuristic-based GA/PSO), and (iii) the dimensionality of the design space. In general, for low-dimensional problems with expensive evaluations, sample-efficient strategies (e.g., BO) are well suited, whereas heuristic-based methods tend to be more effective for higher-dimensional searches when the per-evaluation cost is low, such as when surrogate models are used. However, a systematic understanding of these efficiency regimes remains limited—particularly when considering the total wall-clock budget, including data generation and optimisation runtime, alongside the quality of the final solution.

Table 2.2: Representative model–optimisation combinations for inverse soft actuator design.

Representative work	Model	Optimisation	Typical characteristics
Connolly et al. [50] Yang et al. [87]	Analytical model	Gradient-based	Analytical gradients; fast optimisation
Chen et al. [88] Zhang et al. [86]	FEM	Gradient-based	Sensitivity analysis; topology optimisation
Ghoreishi et al. [93]	Analytical model	BO	Multi-parameter design; sample-efficient search
Wang et al. [91]	FEM	BO	High fidelity; sample-efficient search
Runge et al. [58] Wang et al. [94, 95]	FEM	Heuristic-based e.g., GA, PSO	Expensive evaluations; global search
Liu et al. [89] Atia et al. [90]	Surrogate model	Heuristic-based e.g., GA, PSO	Fast evaluations; global search

Despite their effectiveness, gradient-free actuator-level optimisation relies on manually defined proxy objectives that stand in for task performance, such as deformation targets or force metrics [85, 96, 97]. This reliance on indirect task representations motivates the transition toward task-level optimisation frameworks, in which actuator design is evaluated directly through functional performance within simulated task environments.

2.3.2 Task-level Optimisation

Unlike actuator-level optimisation, task-level optimisation evaluates whether a complete system can accomplish a specified task within a simulated environment. In this setting, task requirements are expressed directly as performance metrics, candidate actuator configurations are instantiated through a chosen parameterisation, and their behaviour is assessed through simulation under actuation and environmental interaction, with optimisation used to iteratively improve task performance (e.g., grasp success [98], locomotion speed [99], or energy efficiency [100]).

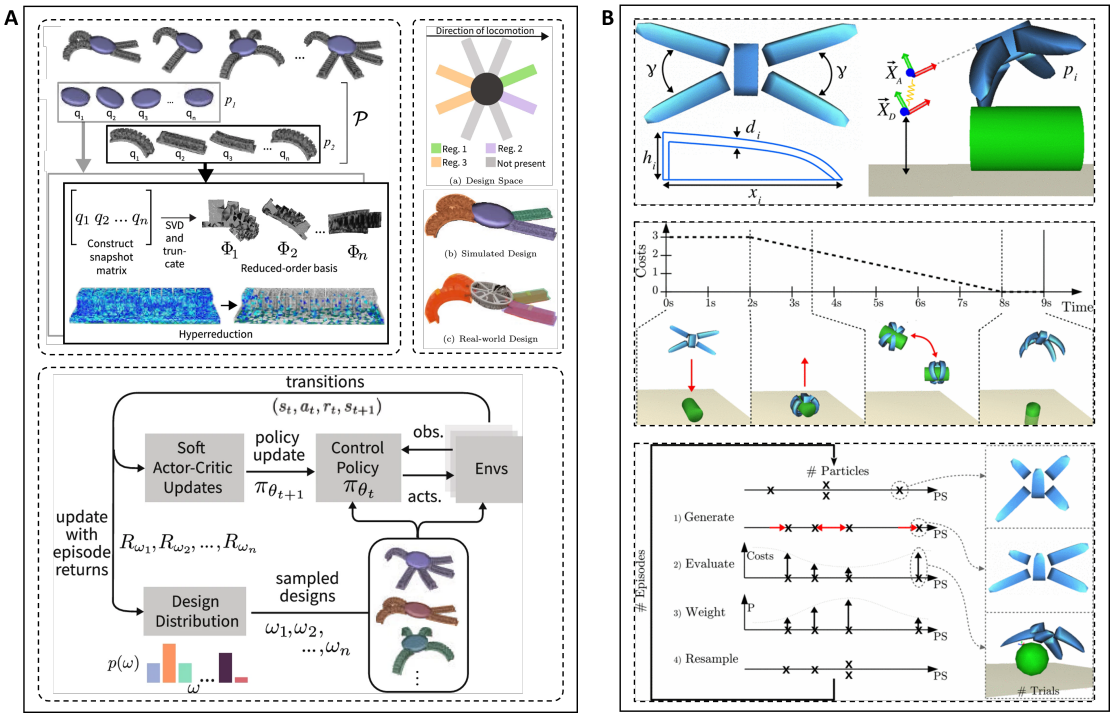


Figure 2.5: Task-level optimisation frameworks built upon high-fidelity simulations. **(A)** shows a RL-based design and control co-optimisation framework for soft robot crawlers, employing POD-reduced FEM model for simulating physically consistent locomotion to optimise travel distance. The upper left panel depicts the use of POD for model reduction, the upper right shows the design space and representative simulated and real-world designs, and the lower panel outlines the joint optimisation process, reproduced with permission [99]. **(B)** presents a co-design method for a soft hand, utilising SOFA-based simulation to jointly optimise the morphology and control for object grasping with the particle filter optimisation method. The upper panel displays the design and control parameter spaces, the middle panel illustrates the simulation of grasping phases, and the lower panel outlines the optimisation workflow, reproduced with permission [98].

Task-level optimisation frameworks built upon high-fidelity simulations such as FEM or its reduced-order variants couple actuator design and control directly within physically grounded environments [98, 99, 101]. Representative examples are shown in Fig. 2.5. Such methods enable realistic evaluation of soft robots under contact, gravity, or environmental interactions, providing physically consistent and fabricable designs. For instance, Schaff et al. [99] employed a POD-reduced FEM model to accelerate the optimisation of soft crawler morphology and actuation pattern, achieving efficient locomotion that could be successfully transferred to physical prototypes. Similarly, Deimel et al. [98] utilised SOFA-based simulation to jointly optimise the morphology and control of a soft hand for grasping.

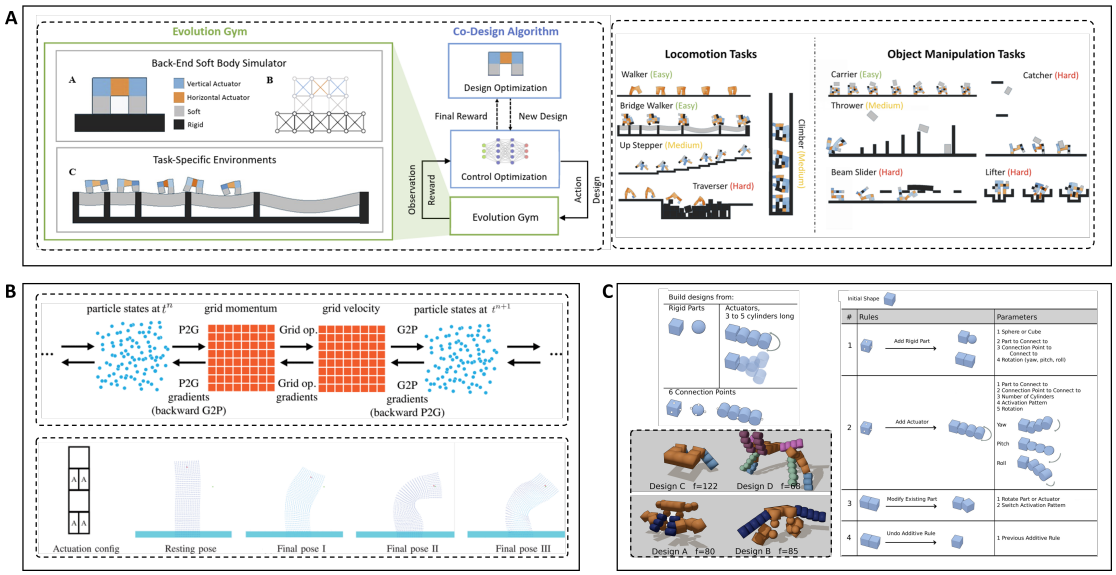


Figure 2.6: Task-level optimisation frameworks using fast proxy simulations. **(A)** presents a benchmark platform for the co-optimisation of voxel-based soft robots modelled as 2D mass–spring systems, employing evolutionary and RL algorithms to optimise both morphology and control across multiple locomotion and manipulation tasks. The left panel provides an overview of the benchmark platform and its integration with co-design algorithms, and the right panel illustrates the diverse optimisation tasks enabled by the framework, reproduced with permission [102]. **(B)** illustrates a modelling and co-optimisation framework based on the MPM for a multi-link soft robotic arm to achieve target poses. The upper panel shows the simulation process using MPM, while the lower panel compares the resulting configurations under different stiffness settings: fixed stiffness at 300% and 100% of the initial Young's modulus (Poses I and II), and a co-optimised stiffness distribution (Pose III). Only the co-optimised design achieves the target configuration, reproduced with permission [43]. **(C)** shows a computational design synthesis approach for automated generation of virtual soft robots, where designs are represented by PRBM combined with RL-based optimisation for locomotion. The upper left panel displays the building blocks of the spatial grammar, the right panel presents the grammar rules used for design generation, and the lower left panel shows representative optimised designs, reproduced with permission [103].

While these approaches provide physically grounded evaluation and fabrication-relevant designs, their simulation pipelines are tightly coupled to specific actuator geometries, actuation models, and task formulations. As a result, extending them to new actuator configurations or task scenarios typically requires substantial redevelopment, limiting their reuse as general task-driven design pipelines.

In contrast, proxy-based optimisation frameworks prioritises compatibility and computational efficiency by adopting abstraction-based representations of soft bodies. Instead of modelling detailed actuator mechanics, these pipelines employ simplified

representations—such as voxel lattices [102, 104], particle-based models [43, 80], or rigid-link approximations [103, 105]—that support fast evaluation across a wide range of morphologies, actuation patterns, and task environments, making large-scale search over coupled design and control spaces computationally feasible. Representative examples are illustrated in Fig. 2.6.

Building on this paradigm, a variety of search and learning strategies have been explored to handle task objectives that are sparse, discontinuous, or difficult to express through smooth actuator-level metrics. Early work employed evolutionary and heuristic-based search to explore morphology spaces by maintaining and mutating large sets of candidate designs [106, 107]. More recent platforms, such as Evolution Gym and SoftZoo, formulate morphology–control co-design as task-level optimisation problems evaluated across diverse environments, using heuristic search or reinforcement learning to explore coupled design and control spaces [80, 102].

Beyond direct search, several studies have introduced learned generative representations to organise and explore morphology spaces more efficiently. Latent-variable models such as variational autoencoders have been used to embed robot morphologies into continuous latent spaces that facilitate sampling and optimisation [108]. Diffusion-based generative models have further been explored to propose candidate morphologies conditioned on task performance signals [109]. These approaches do not remove the need to define a design representation, but instead provide alternative mechanisms for navigating large, high-dimensional design spaces under abstraction-based simulation.

However, the benefits of abstraction-based task-level optimisation arise directly from the simplification of physical and design representations. By unifying geometry, actuation, and interaction into generic modelling primitives, these pipelines omit actuator-specific mechanics and fabrication constraints, making it difficult to ensure consistency with real soft actuator behaviour when transferring designs beyond simulation.

Another direction explores end-to-end differentiable simulation, based on FEM or MPM, for task-driven soft robot design. In these approaches, differentiable

physics models provide sensitivity information that can be used to guide design and control optimisation at the task level. This paradigm has been applied to soft robot co-design using a range of optimisation strategies, including gradient-based topology optimisation for magnetically actuated robots [110] and hybrid pipelines that combine differentiable simulation with generative models, such as diffusion-based morphology generation [109].

While differentiable simulation enables tight coupling between modelling and optimisation, adopting it at the task level typically requires geometry, actuation, deformation, and interaction to be expressed within a differentiable and computationally tractable representation. As a result, practical differentiable pipelines often face a trade-off between simplifying the underlying representation to maintain tractability, or relying on substantial computational acceleration (e.g., GPU-based solvers) to sustain high-fidelity task-level optimisation [79, 83].

Overall, existing task-level optimisation approaches form complementary but disconnected design pipelines. High-fidelity simulations provide physically grounded evaluation but are difficult to reuse across actuator configurations and tasks. Abstraction-based pipelines enable efficient exploration and flexible task specification but sacrifice physical specificity and transferability. Differentiable simulation offers an integrated optimisation formulation but either relies on simplified representations or computational acceleration. As a result, the literature does not yet provide a unified task-driven design pipeline that simultaneously supports physically grounded evaluation, computationally efficient optimisation, and reuse across changing actuator configurations and task contexts.

2.4 Research Gap Analysis

The preceding sections reviewed the SOTA in soft actuator design, modelling approaches, optimisation methods, and task-driven design frameworks. Although these studies have significantly advanced the design and application of soft robotic actuators, several common limitations remain. Rather than being specific to individual methods, these limitations reveal broader research gaps that motivate

the work presented in this thesis. The four research gaps identified from the literature are summarised below.

2.4.1 Research Gap 1

As discussed in Section 2.2, existing modelling approaches range from analytical models and reduced-order formulations to high-fidelity FEM simulations. Analytical models provide rapid evaluation but typically rely on simplified assumptions regarding actuator geometry and deformation [11, 45, 46], while FEM-based approaches offer higher predictive accuracy at substantially greater computational cost [39, 53–55] More recently, several open-source simulation platforms have improved modelling accessibility [43, 44]; however, these tools generally remain either computationally expensive or require significant expertise to use effectively during iterative actuator design. Consequently, despite considerable progress, the literature still lacks modelling and simulation tools that simultaneously provide sufficient accuracy, computational efficiency, and usability for iterative design. This motivates the first research gap: **Limited availability of accurate and computationally efficient modelling and simulation tools for iterative soft actuator design.**

2.4.2 Research Gap 2

The literature reviewed in Section 2.3.1 demonstrates that inverse design of soft actuators for increasingly sophisticated objectives, evolving from actuator-level performance metrics, such as bending angle and blocked force [85, 96, 97], towards prescribed deformation profiles and trajectories[50, 111, 112]. Although these studies demonstrate the feasibility of optimisation-based inverse design, increasing task complexity inevitably leads to larger design spaces and stronger nonlinear relationships between actuator morphology and target behaviour. Consequently, existing approaches struggle to simultaneously maintain modelling fidelity and computational efficiency when solving inverse design problems involving complex behavioural objectives. This motivates the second research gap: **Difficulty of**

inverse design when task complexity leads to large and highly nonlinear design spaces.

2.4.3 Research Gap 3

As summarised in Table 2.2 and discussed in Section 2.3.1, a wide range of model–optimisation combinations have been adopted for inverse soft actuator design, ranging from analytical models with gradient-based optimisation [50, 86, 88] to high-fidelity FEM [56, 57] and surrogate models [89, 92] coupled with Bayesian [91] and heuristic-based optimisation [58]. Among these, high-fidelity FEM and surrogate models combined with gradient-free optimisation have become increasingly adopted for tackling complex nonlinear design problems. Although these combinations have each demonstrated promising performance across different inverse design applications, there has been little systematic investigation into how modelling choices and optimisation strategies jointly influence computational cost, convergence behaviour, and solution quality. This motivates the third research gap: **Lack of systematic understanding of optimisation trade-offs in inverse soft actuator design.**

2.4.4 Research Gap 4

Building upon the task-driven design frameworks reviewed in Section 2.3.2, recent studies have demonstrated the potential of jointly optimising robot morphology and control using high-fidelity simulations [98, 99, 101], abstraction-based representations [43, 102, 103], or differentiable simulation [109, 110]. Although these studies have demonstrated effective task-driven optimisation for specific applications, the resulting design methodologies remain tightly coupled to particular actuator parameterisations, modelling assumptions, and task formulations. Consequently, substantial redevelopment is often required when extending these methodologies to new actuator parameterisations or task scenarios. This motivates the fourth research gap: **Limited generalisation and reuse of task-driven design pipelines across actuator designs and task scenarios.**

3

A Simulation Toolbox for 3D-Printed Bellow SPAs

Contents

3.1 Positioning with respect to the SOTA 33
3.1.1 Problem 33
3.1.2 Objective 34
3.1.3 Contributions 34
3.2 Assumptions, Validity Domain and Limitations 35
3.3 Contributed Paper 36
3.4 Supporting Information 43
3.4.1 Toolbox Repository 43
3.4.2 Supporting Video 43

3.1 Positioning with respect to the SOTA

3.1.1 Problem

The design of bellow SPAs predominantly relies on three types of modelling workflows: analytical models, commercial FEM packages, and, more recently, open-source simulation frameworks. Analytical models, particularly beam-based formulations, provide computationally efficient predictions through simplified geometric and material assumptions [11, 21, 46]. Commercial FEM packages provide high-fidelity

prediction of nonlinear deformation and stress distributions [39, 53], while open-source simulation frameworks improve modelling flexibility and accessibility for soft robotics research [44].

Each of these workflows presents important trade-offs when applied to iterative actuator design. Analytical models are computationally efficient but their predictive capability is limited by simplifying assumptions, particularly for complex geometries and large deformations. High-fidelity FEM provides accurate prediction but requires substantial modelling expertise and computational effort. Open-source simulation frameworks offer greater flexibility and accessibility but generally require considerable programming effort before they can be integrated into practical design workflows. Consequently, designers are often forced to compromise between modelling fidelity, computational efficiency, and usability.

Consequently, there remains a need for modelling tools that enable designers to efficiently translate design intuition into desired actuator behaviour without repeated prototyping and experimental testing.

3.1.2 Objective

The objective of this study is to develop a simulation toolbox for 3D-printed bellow SPAs that supports efficient and predictive actuator design. Specifically, the toolbox aims to provide (i) rapid prediction to facilitate intuition-guided design iteration, (ii) physically grounded evaluation to support reliable engineering design decisions, and (iii) an integrated digital workflow that streamlines the transition from design exploration to manufacturing preparation.

3.1.3 Contributions

The main contributions of this work are summarised as follows:

- A unified simulation toolbox for 3D-printed bellow SPAs, featuring an interactive GUI, that integrates actuator modelling, performance evaluation, and fabrication-oriented outputs within a single digital workflow, supporting efficient iterative actuator design.

- A multi-fidelity modelling strategy that combines rapid theoretical prediction with high-fidelity FEM simulation, enabling efficient design exploration while retaining physically grounded performance evaluation.

- Experimental validation demonstrating that the proposed toolbox can accurately predict actuator behaviour and effectively support the design of 3D-printed bellow SPAs prior to fabrication.

3.2 Assumptions, Validity Domain and Limitations

The proposed toolbox is intended for the design and evaluation of bellow SPAs under **quasi-static loading conditions**. Both the analytical model and the FEM simulations neglect inertial and dynamic effects, assuming equilibrium between internal pressure and structural deformation. This assumption is appropriate for slow pneumatic actuation and design-stage evaluation, where static deformation, workspace, and force characteristics are of primary interest. Consistently, all experimental measurements were recorded only after the actuator reached a stable equilibrium configuration. Therefore, the toolbox is intended for quasi-static design evaluation rather than dynamic motion prediction or controller development.

The **analytical model** is derived from beam theory by representing the actuator as a 2D bending beam, providing rapid prediction of its global deformation behaviour. Rather than modelling the distributed pressure acting normal to the chamber surfaces, the pneumatic loading is represented by an equivalent transverse load that produces an equivalent bending response. This abstraction is appropriate for preliminary estimation of actuator deformation and workspace, but does not capture local deformation, detailed stress distributions, or three-dimensional effects, which are instead evaluated using the accompanying FEM module.

The proposed toolbox adopts a **linear elastic material** description parameterised by Young's modulus and Poisson's ratio rather than a nonlinear hyperelastic constitutive model. Within the intended operating pressure range (0–10 kPa),

deformation is dominated by structural bending, resulting in relatively small material strains. Consequently, material properties characterised within a strain range of 0–1 are sufficient for accurate prediction, as supported by the FEM strain analysis and the good agreement with experimental results. This formulation avoids the need for specialised nonlinear material characterisation and nonlinear material modules. Accordingly, the adopted material model should be regarded as valid within the intended operating range rather than for substantially larger deformations.

The toolbox is developed and **validated for the operating pressure range of 0–10 kPa**, corresponding to the intended operating conditions of the investigated actuators while avoiding structural damage during repeated experiments. Within this range, both the analytical and FEM models show good agreement with experimental measurements. Beyond this range, increasing geometric nonlinearity reduces the accuracy of the analytical approximation and gradually increases discrepancies between simulation and experiment. Accordingly, the toolbox should be regarded as a predictive design tool within its validated operating range rather than a universally applicable model.

Finally, the current implementation is developed specifically for parameterised, pressure-driven bellow SPAs. Extension to other soft actuator architectures would require corresponding modifications to the underlying analytical formulation, parameterisation, and validation procedure.

3.3 Contributed Paper

This work was published as a conference paper in the 2022 IEEE 5th International Conference on Soft Robotics (RoboSoft) and presented orally in Edinburgh, UK. The paper is attached on the following pages.

A Simulation-Based Toolbox to Expedite the Digital Design of Bellow Soft Pneumatic Actuators

Yao Yao[1], Liang He[1*] and Perla Maiolino[1]

***Abstract*— Task-oriented applications of soft pneumatic actuators (SPAs) are growing, as digital design and rapid prototyping allow the actuator to be customised. The bellow SPA is promising for current soft material 3D printing due to small material stretch during actuation, yet the complex geometry causes challenges in physical modelling and design. This article provides the users with an open-source toolbox of simulation for a standardised bellow soft pneumatic actuator that can be 3D printed to expedite the design process. Based on MATLAB and COMSOL Multiphysics, the toolbox allows users to customise design features, automatically generate the CAD and then simulate the actuators' performance. It provides two methods to analyse the design: 1) a theoretical modelling method to quickly generate the actuator's workspace with interactive results, 2) a finite element method to evaluate in detail the actuator's deformation and force performance as well as the mechanical properties. The modelling is introduced and validated by comparing the experimental results of 3D printed bellow actuator prototypes with simulation results for angular deflection and blocked force. The maximum root-mean-square error for the theoretical modelling of deformation, finite element method of deformation and force are 9.00°, 7.40° and 1.66 g, respectively.**

I. INTRODUCTION

Soft Pneumatic Actuator (SPA), made of elastomeric materials and actuated by pressurised air, has received particular interest from the soft robotic community due to its low cost, high compliance, flexibility in design and ease of control. SPAs have been designed in many different methods, targeting a variety of applications, including grasping [1], manipulation [2], and locomotion [3]. In general, SPAs can be categorised into ribbed, cylindrical and pleated types based on channel structure morphology [4]. This study mainly focuses on the bending SPA with the segment of pleated type or bellow type because it can be easily implemented with rapid prototyping. The bellow SPA refers to an actuator consisting of multiple hollow channels connected by a centre channel. When the fluid in the actuator is pressurised, due to the constraint layer at the bottom, each hollow channel would expand and cause the soft actuator to bend.

With the development of soft material 3D printing, the bellow SPAs are increasingly applied in recent studies [5]. Part of the main reason is that the currently available soft materials for 3D printing still do not meet the elasticity and elongation-at-break of conventional silicone rubbers [6], [7].

*This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) Grant EP/V000748/1;

[1] Yao Yao, Liang He and Perla Maiolino are with Oxford Robotics Institute, University of Oxford, Oxford, OX1 2JD, United Kingdom; *Corresponding Author: Liang He yao/liang/perla@oxfordrobotics.institute

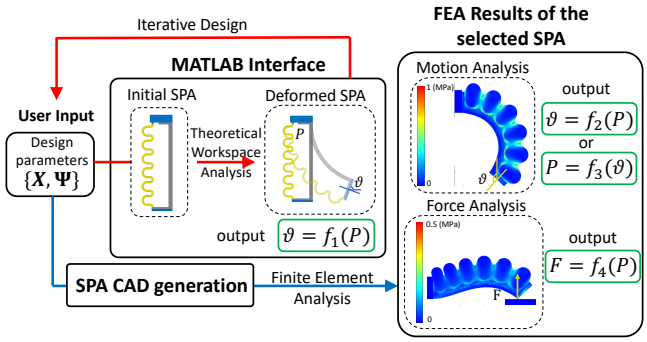


Fig. 1: The framework of the proposed bellow soft pneumatic actuator toolbox. The MATLAB interface takes user input, where $\mathbf{X}$ and $\mathbf{\Psi}$ represent the geometric and soft material parameters. The user-defined design parameters then update the initial bellow soft pneumatic actuator design schematic. The theoretical modelling analyses the SPA workspace for iterative design and outputs a relationship f_1 of the pressure P and angular deflection θ. The finite element modelling method takes the input of design parameters and generates a SPA CAD model. It analyses deformation and force in detail, and output relationship f_2, f_3 of P and θ as well as f_4 of P and blocked force F, respectively.

Thus, to maintain a larger deformation with minimum material stretch, a common solution for a 3D printed pneumatic actuator is to convert the material deformation to structural deformation. Bellow SPA is one of the promising designs for such implementation. The advantage is its capability of generating higher bending curvature and exerting higher maximum forces than ribbed and cylindrical morphology [4] as the bellow structure can provide a linear displacement by unfolding and ignoring the radial expansion [8]. The challenge is its requirements of the complex manufacturing process and higher shore hardness elastomer. However, this can be easily mitigated by 3D printing solutions.

Physical modelling is essential for the design of bellow SPAs. Udupa et al. [9] explored different bellow shapes and geometric parameters using beam theory and finite element analysis (FEA) simulation. Wang et al. [10] designed a soft gripper consisting of bellow SPA and analysed its performance using FEA. Without the physical modelling, the SPA needs to be tested with more in-depth experimental characterisations and developed with many trials of conceptual design iterations. Theoretical modelling is limited by the SPA geometry complexity, and differences in design features will lead to greater changes in the model. While FEA simulation has been widely used in soft robot analysis, it is still a difficult tool to use considering the wide range of parameter settings. All current works develop the simulation model for the actuator design with fixed design features/properties, making it difficult to be generalised. If others want to extend the design to achieve different workspace or force

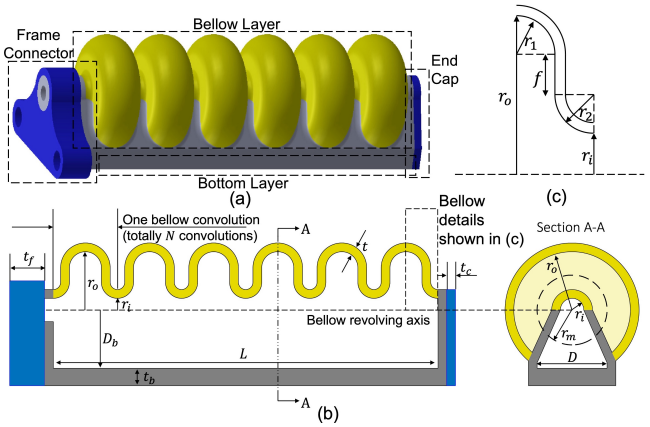


Fig. 2: The bellow SPA design. The 3D dimetric view is shown in (a). The front and sectional schematic view of the parameterised design is shown in (b), in which the colours represent the same parts as in (a), and the shallow yellow in the sectional view A-A is to show the radius of the revolved bellow changing from r_i to r_o. The enlarged half bellow convolution with detailed geometric parameters is shown in (c).

performance, they may need to go through a new process of design, simulation, prototyping, and validation.

Therefore, there is a need for a comprehensive design and analysis solution based on a standardised bellows design that allows the material properties and geometric parameters of the bellows SPA to be customised. A generalised design toolbox is an effective solution to significantly reduce the time the user spends on physical modelling and validation during actuator design. Design tools have already demonstrated their effectiveness for SPAs. Moseley et al. developed an FEA-based tool to facilitate the design of SPAs with ribbed channel structure and cast silicone as the material [11]. However, it is inapplicable for pleated actuators manufactured by soft 3D printing technologies.

This work is going to present a design toolbox for bending bellow SPAs with tunable geometric and material parameters that can analyse the actuator workspace via theoretical modelling for iterative design and further simulate the deformation and force performance by using FEA (Fig.1). In Sec II, the design of the standardised actuator is introduced with key parameters. In Sec III, the toolbox architecture and graphical user interface (GUI) are presented. In Sec IV, the theoretical modelling and FEA modelling are detailed explained. In Sec V, experimental results are shown to validate the modelling methods.

II. Actuators Design and Fabrication

A. Design Parameters

According to Wilson [12], there are mainly ten basic bellow configurations. Among those, the U-shaped bellow is selected in our design as it can compute maximum deflection for under the same input pressure [13].

The standardised bellow SPA design used in the proposed toolbox is shown in Fig.2(a). The bellow SPA is a combination of a revolved bellow layer (in yellow colour) and a flat bottom layer made of soft materials (in grey colour), attached by a frame connector and an end cap made of rigid materials (in blue colour) for characterisation. The design is parameterised according to its geometrical parameters as shown in Fig. 2(b)(c) and Table. I. The bellow layer consists of N of identical axis-symmetric bellow convolution. As shown in Fig. 2(c), half of each convolution can be parameterised by a 1/4 upper arc with radius r_1, a flank with length f, and a 1/4 lower arc with radius r_2. The effective length $L = 2N(r_1 + r_2)$ is defined as the total length of the bellow convolution. Fig. 2(b) indicates that the distance from the revolving axis to the lowest point of the bellow convolution is the inner radius r_i of the bellow, whereas the distance to the highest point is the outer radius r_o, and r_m is defined as the average radius ($r_m = \frac{1}{2}(r_i + r_o)$). The distance from the bellow revolving axis to the bottom layer is defined as $D_b = r_o$ and the width of the bottom layer is defined as $D = 2r_m$. In addition, the bellow wall thickness is defined as t, and the bottom layer thickness is $t_b = 2t$. The frame connector thickness is defined as t_f and the end cap thickness is $t_c = t$.

TABLE I: Design Parameters

Parameter	Symbol	Default Value
Effective Length	L	66 mm
Bellow Number	N	6
Average Radius	r_m	6mm
Inner Radius	r_i	2mm
Outer Radius	r_o	10mm
Upper Arc Radius	r_1	2.75mm
Lower Arc Radius	r_2	2.75mm
Flank Length	f	2.5mm
Distance to the Bottom	$D_b = ro$	10mm
Bottom Width	$D = 2r_m$	12mm
Bellow Wall Thickness	t	1.5mm
Bottom Thickness	$t_b = 2t$	3mm
Frame Connector Thickness	t_f	6mm
End Cap Thickness	$t_c = t$	1.5mm

B. Fabrication of the SPAs

To validate the performance of the toolbox, the SPAs were printed using a commercially available 3D printer (PolyJet J735, Stratasys Inc., Minnesota, United States). This printer can conduct multi-material printing by mixing rigid materials such as the Vero family and soft materials such as Agilus30 to produce digital materials in a variety of flexible Shore A hardness. After printing, the support material SUP706 was removed by chemical bathing.

III. The Toolbox Architecture and Interface

The design toolbox is built using an interface between COMSOL MultiPhysics and MATLAB. Fig. 3 shows the GUI of the toolbox, where users can define critical design features and material properties. Theoretical analysis is used to simulate the 2D deformation of the SPA within a couple of seconds. Users can evaluate the actuator angular deflection in the interactive interface by manually control a slider to adjust the driving pressure. Once the user is satisfied with the SPA 2D workspace, detailed 3D FEA can be carried by click on the "Analyse" button. Subsequently, the design parameters and material properties will be sent to COMSOL with a CAD model generated. Simulation in structural mechanics will be built in the background with the physics, boundary conditions, and mesh created. The GUI will then shows the

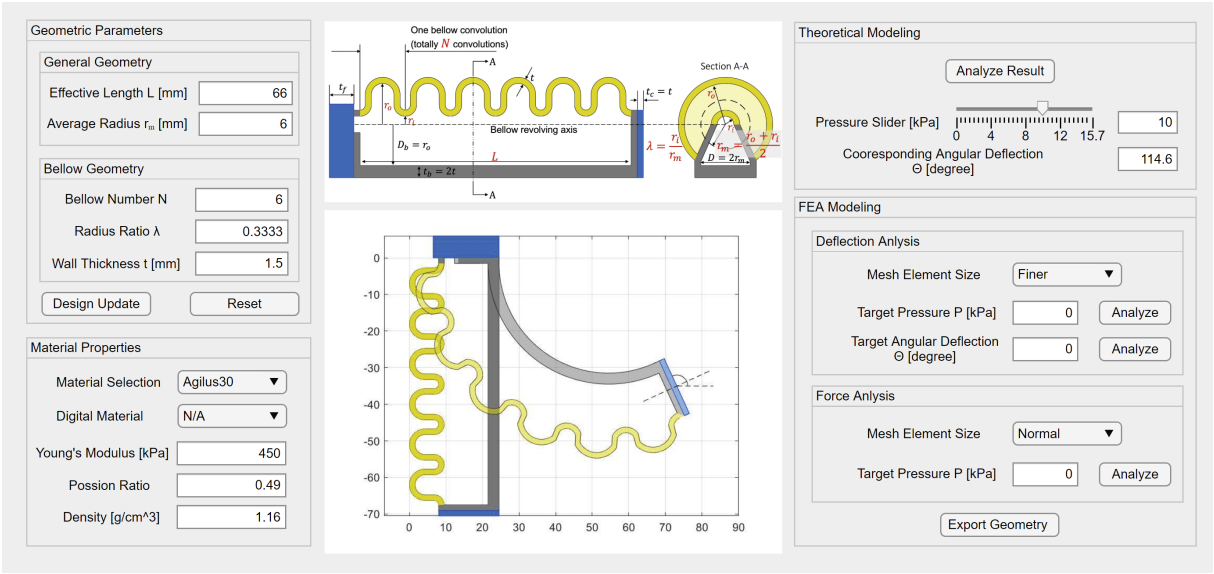


Fig. 3: The interface of the bellow SPA toolbox. The left upper section is the "Geometric Parameters" panel. The left bottom section is the "Material Proprieties" panel. The middle upper section shows the parameterised design of the bellow SPA to provide design guidance. The middle lower section is the design schematic diagram, showing the initial design of the bellow SPA and the deformed bellow SPA after theoretical modeling simulation. The right upper section is the "Theoretical Modeling" panel. The right lower section is the "Finite Element Analysis Modeling" section.

detailed results of displacements, forces, stress and strains acquired via the FEA results (detail will be shown in Sec IV-B and Fig.6). After inspecting the detailed FEA result of the SPA deformation and force output, the user can decide to generate an STL file for 3D printing of the actuator.

A. Geometric Parameters

The toolbox allows for setting critical geometrical parameters based on the design of the standardised bellow SPA, which is categorised in two aspects.

- The general SPA geometry: (a) effective length L. Default: 66mm. (b) average radius of the bellow r_m. Default: 6mm. These two parameters restrain the initial volume of the SPA.
- The bellow geometry: (a) number of the bellow convolution N. Default: 6. (b) the ratio of inner radius over average radius of the bellow $\lambda = r_i/r_m$. Default: 1/3. (c) wall thickness t. Default: 1.5mm. These three parameters would effect bellow's ability of unfolding.

The "Design Update" button checks the geometric constraints of the input and update the design schematic diagram; the "Reset" button would set all geometric parameters to the default value.

Aside from all the geometric parameters should be positive, the geometric constraints are the following: 1) the bellow number N must be a positive integer; 2) the wall thickness t can not exceed the upper or lower arc radius of the bellow, which equals to $L/4N$; 3) the difference between the outer radius r_o and inner radius r_i shouldn't be smaller than the sum of the upper and lower arc radius of the bellow.

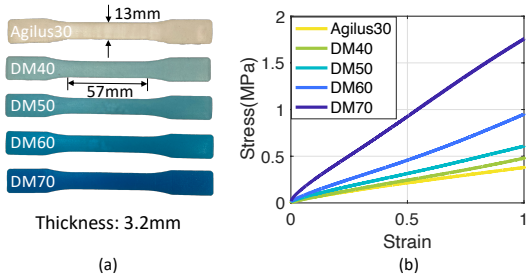


Fig. 4: Material characterization of Agilus30 and its digital materials. (a) Specimens for tensile tests. (b) Stress-Strain curves of the materials.

The equation form is

$$\begin{cases} N \in \mathbb{Z}^+ \\ t < \frac{L}{4N} \\ (1-\lambda) r_m \geq \frac{L}{4N} \end{cases} \tag{1}$$

B. Material Properties

In this paper, the toolbox provides two types of soft materials – Agilus30 and TangoPlus, and their digital materials mixed with the Vero family [14]. Three key material parameters are shown:

- The Young's Modulus $E = \sigma/\epsilon$, where σ and ϵ are the material's mechanical stress and strain. The standardised bellow SPAs undergo relatively small strain as the deformation is mainly due to the geometry unfolding. Thus, the SPA is modelled with a constant E to describe its elasticity where a linear elastic model is assumed for the theoretical model, and a neoHookean model is considered in the FEA. The selection of neoHookean model is based on its few parameters requirements and good agreement for small strain problem [15]. For TangoPlus and its digital materials, the material

properties are extracted from the data found in literature [16]. For Agilus30 and its digital materials, material characterisations were conducted with tensile tests of printed specimens. The specimens are shown in Fig.4(a), and the stress-strain curves are shown in Fig.4(b).

- The Poisson's ratio is a measure of the Poisson effect, the deformation of a material in directions perpendicular to the specific direction of loading. Default: 0.49.
- The density of the material, acquired via data sheet [14].

Material parameters will be updated automatically via drop-down selections in the GUI. Users can also characterise other materials or acquire data from literature/datasheets and input them into the GUI for simulation.

C. Simulation Methods

The toolbox simulation contains theoretical and FEA modelling. The theoretical modelling is based on the Beam Theory and Castigliano's Theorem [17]. Compared with FEA modelling, it can generate the SPA workspace in a short time (less than a minute). FEA modelling is based on COMSOL MultiPhysics, providing pressure-controlled and displacement-controlled simulation. The pressure-controlled simulation computes the angular deflection and blocked force depending on the input pressure, whereas the displacement-controlled simulation calculates the required pressure of the input target angular deflection. In addition, mechanical results such as detailed stress and strain of SPAs are also provided by the FEA modelling. The computation time of the FEA modelling varies depending on the complexity of the model. It takes 1 hour and 14 mins to finish simulating the deformation of SPA with default parameters under 10 kPa pressure. An Intel(R) Core(TM) i7-10875H CPU @ 2.30GHz computer processor with 16.0 GB RAM is used for simulation.

IV. Modelling and Simulation

A. Theoretical Modelling

The theoretical model is built based on the following assumptions: 1) the SPA deforms with approximately constant curvature [18], 2) the bottom layer is considered to be flexible and inextensible [5], 3) the SPA bends mainly due to the bellow unfolding to create a length difference between the bellow and bottom layer instead of radial expansion [8]. Based on these assumptions, the bending bellow SPA can be considered as a linear bellow actuator restrained one side. The ability of the bellow unfolding is related to the actuation pressure, the geometry of the bellow, and material properties. The linear bellow displacement under internal pressure can be modeled by treating each half bellow convolution as a unit-width guided cantilever beam as shown in Fig.5 [17].

The following assumptions are made for analysing the beam: 1) only consider the elastic situation, 2) deformation is only caused by bending strain energy, 3) the shift of the neutral axis of the beam is neglected during deformation, $R_1 = r_1 + \frac{t}{2}$ and $R_2 = r_2 - \frac{t}{2}$, 4) there is no rotation at A and D, 5) the vertical component of the pressure load is

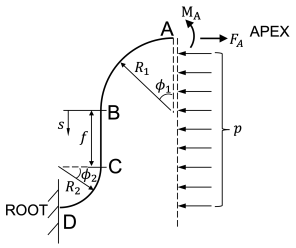


Fig. 5: Unit-width half bellow convolution modelled as guided cantilever beam

eliminated in the analysis, since the radial expansion of the bellow is neglected [19]. Therefore, the bending moment for each region of the beam is:

$$M_1 = M_A + p\frac{R_1^2(1-\cos\phi_1)^2}{2} - F_A R_1(1-\cos\phi_1) \quad (2)$$

$$M_2 = M_A + p\frac{(s+R_1)^2}{2} - F_A(s+R_1) \quad (3)$$

$$\begin{aligned} M_3 =& M_A + p\frac{(R_1+f+R_2\sin\phi_2)^2}{2} \\ &- F_A(R_1+f+R_2\sin\phi_2) \end{aligned} \quad (4)$$

where the subscript 1,2,3 stand for arc AB, flank BC and arc CD, respectively. The bending strain energy is

$$U = \frac{\int_0^{\frac{\pi}{2}} M_1^2 R_1 \mathrm{d}\phi_1 + \int_0^f M_2^2 \mathrm{d}s + \int_0^{\frac{\pi}{2}} M_3^2 R_2 \mathrm{d}\phi_2}{2EI} \quad (5)$$

where $I = \frac{1}{12}t^3$ is the unit-width second moment of inertia. According to Castigliano's theorem, the rotation φ and axial displacement δ_x at A are defined as

$$\varphi = \frac{\partial U}{\partial M_A} = 0 \quad (6)$$

$$\delta_x = \frac{\partial U}{\partial F_A} \quad (7)$$

From the symmetry of the bellow convolution and the pressure load on the end cap of the linear bellow

$$F_A = \frac{p(R_1+f+R_2)}{2} + \frac{p\pi r_i^2}{2\pi r_i} \quad (8)$$

The length difference between the bellow layer and constrained layer is $\Delta L = N\delta_x/b$, where b is a free parameter that allows adjustment of the length difference so that it can better match with experimental results. Then the angular deflection Θ of the bending bellow SPA is the length difference ΔL divided by the distance from the bellow highest point to the bottom of the actuator.

$$\Theta = \frac{\Delta L}{2r_o + t_b} \quad (9)$$

B. FEA Modelling on COMSOL

The FEA Modelling with COMSOL MultiPhysics contains SPA's deformation and force performance evaluation.

The simulation of deformation is built in the following steps: 1) Create a 3D solid mechanics physics model of stationary study type. 2) Define geometric and material parameters in Global Definitions. 3) Build the 3D geometry

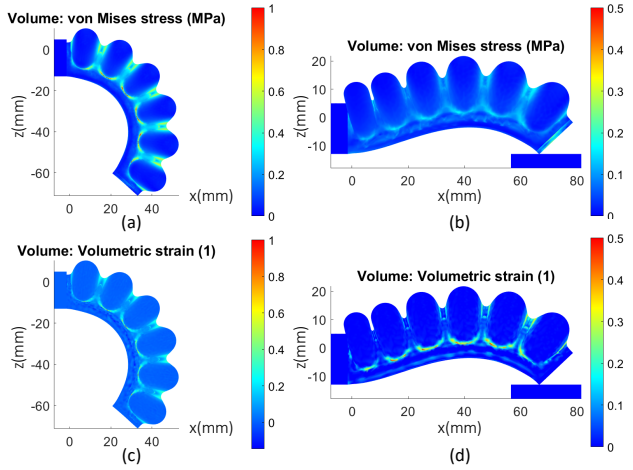


Fig. 6: The FEA results of the example bellow SPA at 10 KPa, including the stress distribution of the deformation simulation (a) and the force simulation (b), together with the strain distribution of the deformation simulation (c) and force simulation (d).

model. 4) Define materials and assign them to the corresponding domains. Use mixed "Pressure formulation" in Solid Mechanics → Linear Elastic Material. Apply "Neo-Hookean" model on the soft domain. 5) Add boundary conditions. The pressure load "P" is defined in Global Definitions. 6) Mesh SPA with default element size "Finer" and build. In the toolbox, users can also change mesh size to get a more accurate result or less computational time. 7) Create stationary Study 1 for pressure control simulation: use range(0, P_s, P_f) to ramp "P", where "P_f" and "P_s" are defined as the target pressure and simulation interval; Use "Suggested Direct Solver (solid)" as the linear solver and select "MUMPS" solver; Set "Constant Newton" and "Anderson acceleration" as the nonlinear method. 8) Create stationary Study 2 for displacement control simulation: use range(0, d_s, d_f) to ramp the edge displacement "d", where "d_f" and "d_s" are defined as the target displacement and simulation interval. Before that, create an Average operator in Component → Definitions and select the end bottom edge as an entity. Next, in the Solid Mechanics → Global Equation, define "aveop(-u)-d" as the equation and "P" as the dependent variable.

The simulation of blocked force is modelled similarly except: 3) Add a cylinder adhered to the end of the bottom layer. Form the whole geometry as an assembly with a contact pair. 5) Define contact pairs in Component → Definitions. Add contact pair condition under Solid Mechanics. 6) Select mesh size "Normal". 7) In this case, we only use the pressure control simulation.

Fig.6 shows the FEA results of the example bellow SPA using the default geometry and material settings (see Sec III-A and III-B) acquired via the toolbox. Both the deformation and stress/strain can be observed. Compared with the tensile strength of 2.4 - 3.1 MPa and elongation-at-break of 220 - 270% of Agilus30 [20], the resultant maximum von mises stress and maximum volumetric strain shown in Fig. 6 indicates that the example bellow SPA should be safe under the pressure of 10 kPa.

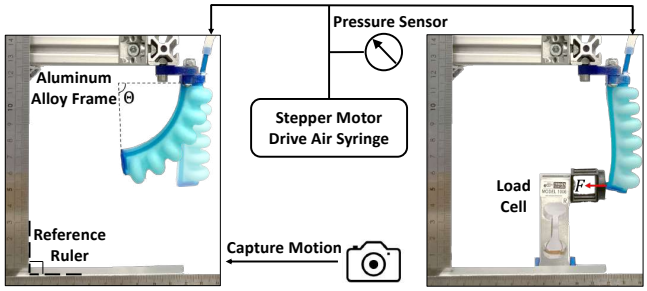


Fig. 7: The experimental set up to verify the simulation results of deformation (left) and force (right) analysis.

V. Experiments Validation and Discussion

To validate the analysis results provided by the toolbox, we designed and conducted experimental characterisation. In the experiment, the driving pressure of the SPA is selected as from 0 to 10kPa. Different design parameters are tested to verify the model accuracy, while experiments of both deformation and force are designed in three groups: changing the bellow number N, radius ratio $\lambda = r_i/r_m$ and the material stiffness, with respect to the default design parameters. The effective length and average radius of all prototypes are kept as the same: $L = 66$mm, $r_m = 6$mm. For each prototype, the experiment was conducted 5 times repeatedly.

Fig. 7 shows the experiment setup, where the bellow SPA was fixed by connecting the frame connector to an Aluminum alloy frame. A pneumatic system consisting of a 12V DC stepper motor driving a 60mL air syringe, a microcontroller (Arduino Mega), a pressure sensor (ADP51A11, pressure range 0-40 kPa, Panasonic, Japan), and a PI controller is used to control the driving pressure of the SPA. In each test, the pressure sensor is used as feedback to conduct close loop control of the output pressure and data is collected after pressure stabilising for 30 sec to minimise material hysteresis. In the experiments of SPA deformation, the angular deflection Θ is defined as the angle between the frame connector and the end cap. A RealSense camera (Intel, USA) is used to capture the deflection angle, which is calculated by MATLAB image processing toolbox using the frame connector end cap's pixel coordinates. A right-angle reference ruler was attached to the Aluminum alloy frame to provide calibration of the camera image. In the blocked force experiments, a load cell (Model 1006, Tedea-Huntleigh) with amplifier (HX711, SparkFun, United States) is used to collect the force data via the microcontroller. We took average measurement of 200 readings after pressure stabilised for 30 seconds, which lasted about 20 seconds.

To match the data provided by the theoretical modelling with the experimental results, we find the free parameter b defined in Sec IV-A works well when $b \sim \frac{\lambda^{0.3}}{N^{0.9}.E^{0.6}}$.

Fig.8 shows the comparison between the toolbox simulation and the experimental result. In the results of deformation, both modelling methods fit the tendency of experimental data. Due to gravity, the experimental and FEA results have an offset at 0 kPa, and the theoretical model does not take this into account, so it starts from 0°. Compared to the experimental result, the root mean square error (RMSE) for

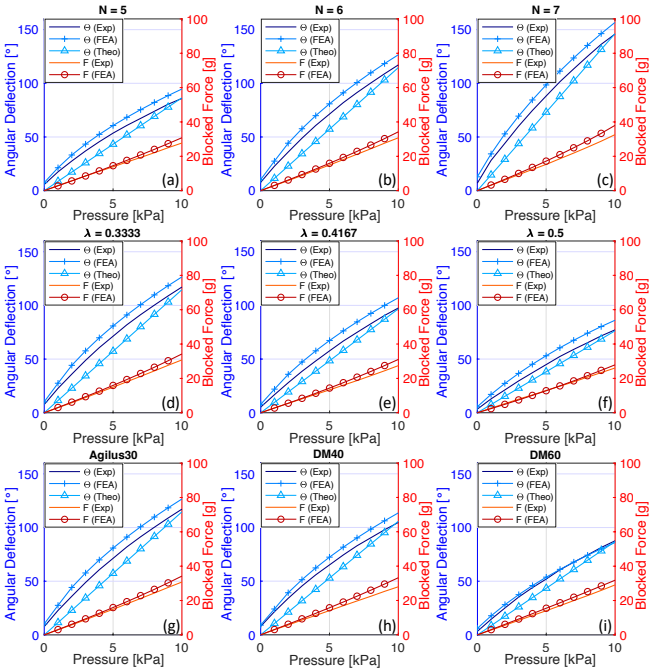


Fig. 8: The comparison of results from experiment and modelling of both deformation and force: the group of changing bellow number N, radius ratio λ and soft material are shown in (a-c), (d-f) and (g-i), respectively.

the FEA simulation and theoretical modelling are 9.00° and 7.40°, respectively.

As for the blocked force comparison, the FEA results also show a similar trend to the experimental ones. In all cases, the deviation of the FEA results from the experimental data increases as the pressure increases, which leads to an increasing error. The RMSE of the FEA simulation is 1.66 g. In general, the modelling results of deformation and force are in good agreement with the experimental results, which indicates the toolbox is reliable.

Furthermore, the toolbox has some limitations. First of all, it analyses the SPA performance based on the quasi-static equilibrium assumption. Secondly, it only considers newly manufactured prototypes, although SPA performance will vary with age. Besides, it doesn't provide information about the durability of the design actuators. However, the paper of Dämmer et al. [21], provides insights about the relationship between load cycles to failure and finite element simulated maximum principal strain on the PolyJet bellows actuators. Suppose users are interested in the durability of the bellow actuator. In that case, they can look at the strain results from the finite element simulation in the toolbox and compare them with the work in [21].

VI. Conclusion

This paper describes a toolbox for the design and analysis of the bellow soft pneumatic actuator and discusses its limitations. The toolbox aims to expedite the digital design process of bellow SPAs with a package containing the customisation of geometric/material parameters, fast theoretical modelling (based on Beam Theory and Castigliano's Theorem), detailed FEA modelling (based on COMSOL Multiphysics), and the SPA CAD for direct 3D printing. Experimental results of the angular deflection and blocked force of the selected SPAs in three dimensions of design parameters is conducted, validating the toolbox with maximum RMSE of 9.00°, 7.40° and 1.66 g for the theoretical modelling of deformation, FEA modelling of deformation and force, respectively. Future work of the design toolbox would focus on introducing multi-objective design optimisation to user-defined tasks.

References

[1] 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.

[2] L. He, X. Tan, K. Suzumori, and T. Nanayakkara, "A method to 3d print a programmable continuum actuator with single material using internal constraint," *Sensors and Actuators A: Physical*, vol. 324, p. 112674, 2021.

[3] A. D. Marchese, C. D. Onal, and D. Rus, "Autonomous soft robotic fish capable of escape maneuvers using fluidic elastomer actuators," *Soft Robotics*, vol. 1, no. 1, pp. 75–87, 2014. PMID: 27625912.

[4] A. D. Marchese, R. K. Katzschmann, and D. Rus, "A recipe for soft fluidic elastomer robots," *Soft Robotics*, vol. 2, no. 1, pp. 7–25, 2015. PMID: 27625913.

[5] A. Zolfagharian, M. A. P. Mahmud, S. Gharaie, M. Bodaghi, A. Z. Kouzani, and A. Kaynak, "3d/4d-printed bending-type soft pneumatic actuators: fabrication, modelling, and control," *Virtual and Physical Prototyping*, vol. 15, no. 4, pp. 373–402, 2020.

[6] J. Herzberger, J. M. Sirrine, C. B. Williams, and T. E. Long, "Polymer design for 3d printing elastomers: Recent advances in structure, properties, and printing," *Progress in Polymer Science*, vol. 97, p. 101144, 2019.

[7] O. Shorthose, L. He, A. Albini, and P. Maiolino, "Design of a multi-material 3d-printed soft actuator with bi-directional variable stiffness," in *Annual Conference Towards Autonomous Robotic Systems*, pp. 238–248, Springer, 2021.

[8] R. Hashem, M. Stommel, L. Cheng, and W. Xu, "Design and characterization of a bellows-driven soft pneumatic actuator," *IEEE/ASME Transactions on Mechatronics*, pp. 1–1, 2020.

[9] G. Udupa, P. Sreedharan, P. Sai Dinesh, and D. Kim, "Asymmetric bellow flexible pneumatic actuator for miniature robotic soft gripper," *Journal of Robotics*, vol. 2014, no. 902625, 2014.

[10] Z. Wang and S. Hirai, "Chamber dimension optimization of a bellow-type soft actuator for food material handling," in *2018 IEEE International Conference on Soft Robotics (RoboSoft)*, pp. 382–387, 2018.

[11] P. Moseley, J. M. Florez, H. A. Sonar, G. Agarwal, W. Curtin, and J. Paik, "Modeling, design, and development of soft pneumatic actuators with finite element method," *Advanced Engineering Materials*, vol. 18, no. 6, pp. 978–988, 2016.

[12] J. Wilson, "Mechanics of bellows: A critical survey," *International Journal of Mechanical Sciences*, vol. 26, no. 11, pp. 593–605, 1984.

[13] H. F. Lau, "3D-Printed Inflatable Actuators:Design and Development of Soft Actuators for a Pneumatically-Actuated Soft Robotic Arm," Master's thesis, ETH Zurich, Zurich, Switzerland, 2019.

[14] Stratasys, *Digital Materials Data Sheet*, 2020.

[15] M. S. Xavier, A. J. Fleming, and Y. K. Yong, "Finite element modeling of soft fluidic actuators: Overview and recent developments," *Advanced Intelligent Systems*, vol. 3, no. 2, p. 2000187, 2021.

[16] V. Slesarenko and S. Rudykh, "Towards mechanical characterization of soft digital materials for multimaterial 3d-printing," *International Journal of Engineering Science*, vol. 123, pp. 62–72, 2018.

[17] R. Chipman, R. Galantine, J. Susnir, U. A. E. Commission, and N. A. A. A. I. Division, *Stress Analysis of Bellows for the HNPF Intermediate Heat Exchanger*. AEC research and development report, Atomics International, 1962.

[18] G. Zhong, Y. Hou, and W. Dou, "A soft pneumatic dexterous gripper with convertible grasping modes," *International Journal of Mechanical Sciences*, vol. 153-154, pp. 445–456, 2019.

[19] N. Murray, "Stress analyses of wrinkle bends in pipelines," *Thin-Walled Structures*, vol. 17, no. 1, pp. 65–80, 1993.

[20] Stratasys, *Agilus30 Materials Data Sheet*, 2020.

[21] G. Dammer, S. Gablenz, A. Hildebrandt, and Z. Major, "Polyjet-printed bellows actuators: Design, structural optimization, and experimental investigation," *Frontiers in Robotics and AI*, vol. 6, p. 34, 2019.

3.4 Supporting Information

3.4.1 Toolbox Repository

The open-source toolbox, along with its installation and setup instructions, can be found here: `https://github.com/ORI-SRL/Bellow_SPA_Toolbox.git`

3.4.2 Supporting Video

A video demonstrating how to use the toolbox is provided here: `https://github.com/ORI-SRL/Bellow_SPA_Toolbox/blob/main/SupplementaryVideo.mp4`

4

An Optimisation Toolbox for Bellow SPAs in Shape Matching

Contents

4.1 Positioning with respect to the SOTA 44
4.1.1 Problem 44
4.1.2 Objective 45
4.1.3 Contributions 45
4.2 Assumptions, Validity Domain and Limitations 46
4.3 Contributed Paper 46
4.4 Supporting Information 62
4.4.1 Toolbox Repository 62
4.4.2 Supporting Video 62
4.4.3 Supplementary Data 62

4.1 Positioning with respect to the SOTA

4.1.1 Problem

Inverse design has become an increasingly important paradigm for soft actuator development, enabling actuator geometries to be automatically generated from desired performance objectives rather than through manual trial-and-error. Existing approaches generally rely on analytical models [50, 111], high-fidelity FEM-based

optimisation [56, 58, 94], or data-driven surrogate models [89, 90, 92].

As soft actuators are increasingly required to realise complex behaviours, inverse design objectives have evolved from simple actuator performance metrics, such as bending angle or blocked force, to prescribed continuous deformation profiles. These increasingly complex inverse design problems expose the limitations of existing modelling and optimisation workflows. Analytical models remain computationally efficient but lack sufficient fidelity. Direct optimisation based on FEM offers high fidelity but becomes computationally prohibitive due to repeated nonlinear simulations. Although data-driven surrogate models substantially improve computational efficiency, existing approaches are primarily developed to predict individual actuator performance responses rather than the complex deformation behaviour required for shape-matching design optimisation.

Consequently, there remains a need for inverse design methodologies that maintain predictive accuracy while remaining computationally efficient for soft actuator shape matching.

4.1.2 Objective

The objective of this work is to extend inverse design of bellow SPAs beyond simple actuator performance objectives towards shape-matching objectives. To achieve this, the chapter aims to develop a modelling and optimisation approach that supports efficient and accurate evaluation within iterative optimisation, enabling actuator geometries to be generated automatically from prescribed deformation profiles.

4.1.3 Contributions

The main contributions of this work are summarised as follows:

- A modular actuator representation for bellow SPAs that enables flexible composition of actuator modules towards shape-matching objectives.
- A surrogate modelling strategy in which a NN is trained on modular FEM simulation data and integrated with PCC-based kinematic formulations,

enabling efficient evaluation of complex shape-matching objectives during inverse design optimisation.

- An end-to-end inverse design framework, implemented as the SPADA toolbox, that automatically generates manufacturable bellow SPA designs from prescribed shape-matching objectives.

4.2 Assumptions, Validity Domain and Limitations

The proposed surrogate model is trained on deformation data generated from modular FEM simulations, with a NN learning the behaviour of individual actuator modules. The predicted module-level responses are subsequently combined through the adopted kinematic formulation to reconstruct the overall actuator shape. This approach assumes that the training dataset provides sufficient coverage of the design space, allowing the surrogate to accurately interpolate between sampled designs. Consequently, the surrogate is intended for interpolation within the predefined geometric and pressure ranges represented in the training data, whereas prediction outside this domain constitutes extrapolation and cannot be expected to maintain the same level of accuracy.

The overall actuator shape is reconstructed using the adopted piecewise constant-curvature kinematic formulation, which assumes that the deformation of each actuator module can be represented by constant curvature and that the global actuator shape is determined primarily by pressure-induced deformation. Consequently, the current shape model is not intended for applications in which actuator deformation is significantly influenced by gravity, external loading, or contact interaction, for which these effects would need to be explicitly incorporated into the kinematic formulation.

4.3 Contributed Paper

The work has been published as a journal paper in Robotics Reports. The paper is attached on the following pages.

SPADA: A Toolbox of Designing Soft Pneumatic Actuators for Shape Matching based on Surrogate Modeling

Yao Yao[1], Liang He[2] and Perla Maiolino[1,3] *

Abstract

Soft pneumatic actuators (SPAs) produce motions for soft robots with simple pressure input, however, they require to be appropriately designed to fit the target application. Available design methods employ kinematic models and optimization to estimate the actuator response and the optimal design parameters to achieve a target actuator's shape. Within SPAs, bellow SPAs excel in rapid prototyping and large deformation, yet their kinematic models often lack accuracy due to the geometry complexity and the material nonlinearity. Furthermore, existing shape-matching algorithms are not providing an end-to-end solution from the desired shape to the actuator. In addition, despite the availability of computational design pipelines, an accessible and user-friendly toolbox for direct application remains elusive. This paper addresses these challenges, offering an end-to-end shape-matching design framework for bellow SPAs to streamline the design process, and the open-source toolbox SPADA (**S**oft **P**neumatic **A**ctuator **D**esign fr**A**mework) implementing the framework with a graphic user interface for easy access. It provides a kinematic model grounded on a modular design to improve accuracy, finite element method (FEM) simulations, and piecewise constant curvature (PCC) approximation. An artificial neural network-trained surrogate model, based on FEM simulation data, is trained for fast computation in optimization. A shape-matching algorithm, merging three-dimensional (3D) PCC segmentation and a surrogate model-based genetic algorithm, identifies optimal actuator design parameters for desired shapes. The toolbox, implementing the proposed design framework, has proven its end-to-end capability in designing actuators to precisely match two-dimensional shapes with root-mean-square errors of 4.16, 2.70, and 2.51 mm, and demonstrating its potential by designing a 3D deformable actuator.

Keywords: soft actuators, modelling, optimization, toolbox

1. Introduction

Soft pneumatic actuators (SPAs) have a significant impact on the soft robotics field by enabling a variety of applications of soft robotics, such as grasping,[1,2] locomotion,[3,4] and manipulation.[5,6] These versatile functionalities come from not only the inherent flexibility and adaptability of soft materials but also the capacity of SPAs to produce diverse two-dimensional (2D) and three-dimensional (3D) motion patterns.[7–9] This is achieved by preprogramming the morphology of the actuator to achieve complex motions from a single pressure input, thus reducing the need of complex control.[10–13]

This trait has been widely employed in designing soft robots, especially those that draw inspiration from biological entities, as the desired actuator shape can be derived from creatures' movement. For instance, the locomotion pattern of inchworms has been replicated by designing actuators to recursively crawl,[4] while the grasping profile of an elephant trunk has been emulated by actuators with helical configurations to achieve versatile and secure holds.[6]

*[1] Yao Yao and Perla Maiolino are with Oxford Robotics Institute, University of Oxford, Oxford, OX1 2JD, United Kingdom; [2] Liang He is with Institute of Biomedical Engineering, University of Oxford, Oxford, OX1 2JD, United Kingdom; [3] Perla Maiolino is also with the University of Genoa, Liguria, IT. Corresponding Author: Yao Yao. yao.yao/liang.he/perla.maiolino@eng.ox.ac.uk

Despite considerable advancements in discovering suitable soft actuator structures and materials to mimic biological motions, the design process is impeded by complexities tied to the dynamic, contact, and multimodal modeling of these actuators.[8,9] Addressing these challenges remains difficult, given the limitation of current modeling and computational approaches in managing large geometric deformation and material nonlinearity.[14] As a result, the research community's current primary focus lies in designing soft actuators for specific profiles or trajectories with kinematic models and optimization methods[15] by establishing systematic methodologies and strategies to enhance efficiency.[12,15–17]

Fiber-reinforced SPAs, recognized for their versatile applications,[4,18,19] leverage fiber arrangements for varied 2D and 3D motion.[10] Connolly et al.[16] presented an analytical model for these SPAs, comprising distinct modules for bending, twisting, elongating, and expanding. They took a predefined kinematic trajectory for each module and applied optimization to identify optimal actuator design parameters for matching shapes along the trajectory.

Singh and Krishnan,[17] concentrating on bending modules, proposed a method that segments 2D curves into piecewise constant curvature (PCC) sections, with design parameters derived from their analytical model. However, the complex manufacturing of fiber-reinforced SPAs, especially the fiber routing process, faces challenges such as extended production time, inconsistency issues, and design limitations.

PneuNets actuators, formed of interconnected, pleated channels within an elastomer, deform when pressurized.[20] Their behavior is defined by altering channel geometry or material distribution.[11] Advances in 3D printing allow these actuators to be crafted either via 3D-printed molds and silicon casting[15]or direct 3D printing with flexible materials.[21] Following Connolly et al.'s strategy,[16] Jiang et al.[15] modeled PneuNets, which include bending, twisting, and helical modules. Their design process starts with manual 3D curve segmentation, followed by their analytical model and optimization to determine the actuator design for the target shape.

Transitioning to bellow SPAs, replacing PneuNets' sharp channels with bellow-shaped convolutions, leads to an unfolding structure under pressure.[22,23] This feature enables transferring material strain into structural deformation, making bellow SPAs particularly suitable for 3D printing materials like Agilus30™ due to their lower elongation-at-break compared to silicon elastomers ,[24] ensuring precision in production. Kan et al.[12] proposed an analytical model for modularized bellow SPA designs, using interconnected channels to achieve varied deformation curves. They employed a sampling-based optimization to design channels for a desired end-tip trajectory.

Analytical models, although widely used for forward kinematics, often struggle to accurately predict substantial deformations due to their reliance on geometric simplifications coupled with material nonlinearity. The finite element method (FEM) is known for its accurate predictions[25] but is computationally expensive, making it unsuited for repeated optimization tasks.[26,27] Recent advances use machine learning, particularly supervised artificial neural networks (ANNs) trained with FEM data, to create efficient surrogate models for bellow SPAs.[28] However, these models have covered limited actuator design spaces.[29]

Furthermore, current shape-matching optimization methods for SPAs are limited by the need for human intervention in segmenting 3D shapes and in converting optimal parameters into a manufacturable design. This results in a lack of an end-to-end solution for seamlessly connecting the desired shape to a ready-to-print actuator design file. Furthermore, the inherent complexities of implementing design frameworks emphasize the need for an open-source user-friendly design toolbox to enhance accessibility and efficiency.[30] Presently, the available toolboxes for bellow SPAs exhibit some limitations, either focusing solely on simulation or covering a limited design space.[31–33] In contrast, some permit optimization but lack a specific focus on shape-matching.[33,34]

Therefore, to address the above challenges, this article provides an end-to-end shape-matching design framework for bellow SPAs, and an open-source toolbox named SPADA (**S**oft **P**neumatic **A**ctuator **D**esign fr**A**mework), which implements this framework through a user-

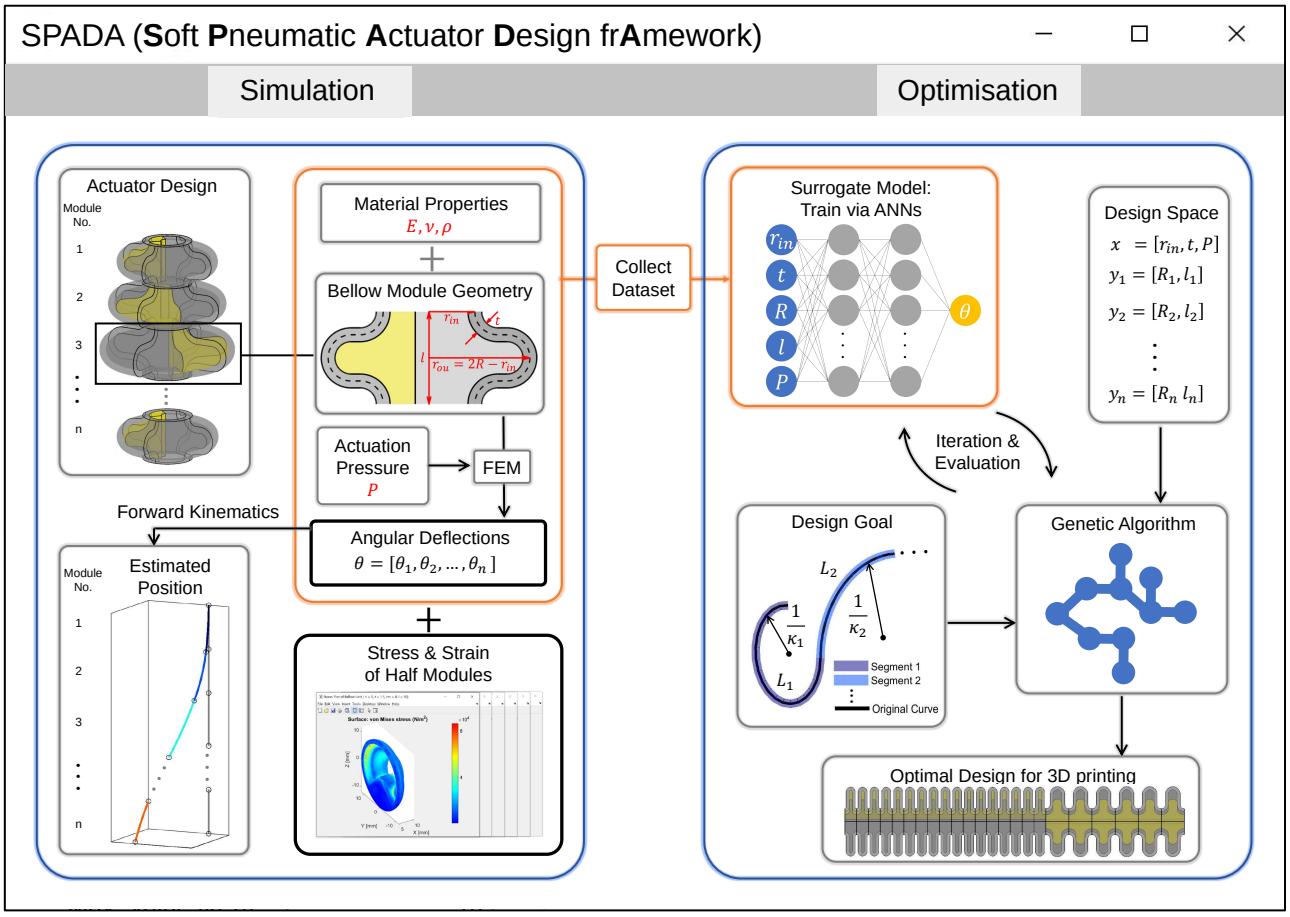


Figure 1. An overview of the bellow soft pneumatic actuators (SPAs) design toolbox, consisting of two main components: the simulation and the optimization components. The simulation component (left) facilitates the design of bellow SPAs by allowing modification of material properties, such as Young's Modulus, Poisson's ratio and density, adjusting the modular geometric parameters, and customizing module assemblies. It utilizes the FEM to simulate each module's angular deflection, mechanical stress, and strain based on the defined actuation pressure and predicts the actuator's configuration through forward kinematics. It also collects a data set of the FEM model. The optimization component (right) uses an ANN to train on the data set to provide a surrogate model for optimization. It takes an input of a target shape and divides it into constant curvature segments with length and curvature parameters using a 3D PCC segmentation. Then, a genetic algorithm is employed to optimize the parameters in the design space of a bellow SPA. Finally, a CAD file of the bellow SPA based on the optimal design parameter for matching the target shape can be generated.

friendly graphic user interface (GUI). Fig. 1 shows the schematic of the functions included in this design toolbox, consisting of simulation and optimization functionalities. The simulation component permits modular design customization, defining material properties, and predicting actuator kinematics using FEM combined with PCC approximation.

It also allows for collecting a FEM dataset based on varied geometric parameters and actuation pressures. The optimization component can generate a surrogate model, trained on FEM or self-characterized data, to expedite computations. It segments input 2D/3D curves using PCC segmentation and uses the surrogate model to optimize the bellow SPA's design parameters for shape-matching.

The rest of this article is organized as follows: Design and Methods section details the shape-matching design method, covering the bellow SPA's kinematics, FEM simulation, surrogate modeling, and the shape-matching optimization for determining design parameters. The Toolbox Implementation section introduces the SPADA toolbox, implementing the discussed framework with a user-friendly GUI for kinematic analysis and shape-matching optimization. Results section applies the toolbox for precise 2D shape-matching designs, also exploring 3D actuator design potential.

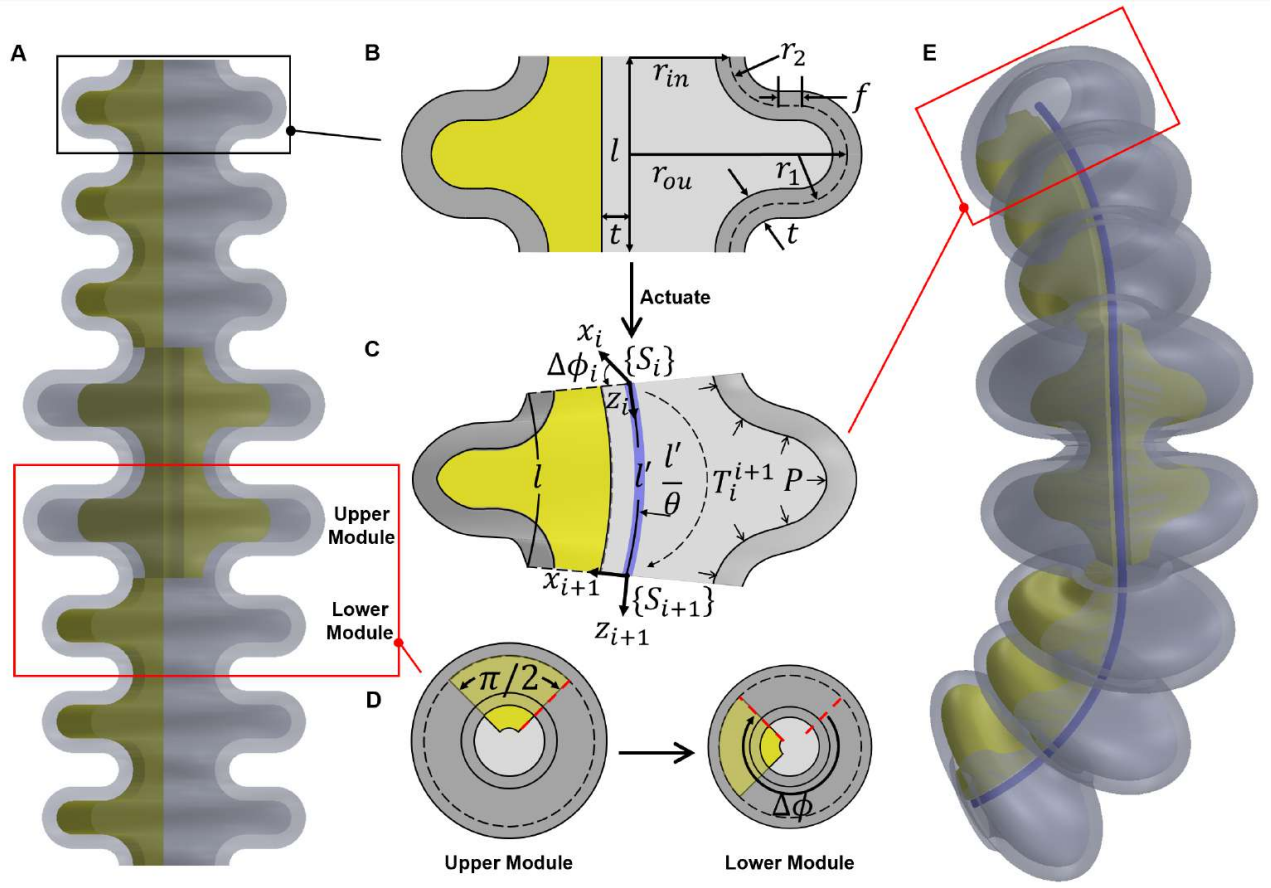


Figure 2. The design and kinematics of the bellow SPAs. A) A bellow SPA consists of modules. B) The parameterized 2D geometry of the module. C) The deformed module and its kinematics model represented by an arc with length and curvature. D) Rotation angle between adjacent modules. E) The deformed bellow SPA upon pressurization with an inner line representing its deformed shape.

2. DESIGN AND METHOD

This section describes the methodologies employed for designing, analyzing, and optimizing bellow SPAs for shape-matching. We present a modular design, describe its kinematics, discuss FEM simulations and ANN training, and introduce a shape-matching optimization algorithm for desired actuator shapes.

A. bellow SPAs Design and Kinematics

Modularity has demonstrated its effectiveness in exploring large design space of SPAs for achieving desired behaviors (Fig. 2).[10–13] This approach has been uniformly adopted in prior design methodologies for shape-matching, owing to its capability to accelerate kinematics prediction and optimization.

As shown in Fig. 2, a bellow SPA (Fig. 2A) can be built by stacking modules, whose deformation is determined by the deformation of each module along with the rotation angle between two adjacent modules. Each module consists of a U-shaped bellow ring and a $\frac{\pi}{2}$ fan-shaped constraint (Fig. 2B), parameterized by the geometric parameters given in Table 1.

Table 1. Geometric Parameters of the bellow SPA Module.

Parameters	Symbol/ Equation	Default Value (mm)
Module Length	l	10
Inner Radius	r_{in}	5
Average Radius	R	8
Outer Radius	$r_{ou} = 2R - r_{in}$	11
Upper Radius	$r_1 = l/4$	2.5
Lower Radius	$r_2 = l/4$	2.5
Wall Thickness	t	1.5
Flank Length	$f = r_{ou} - r_{in} - r_1 - r_2$	1

Essentially, the geometry of the bellow SPA module is governed by four parameters r_{in}, t, R and l, bounded by their non-negativity,

$$\begin{cases} l > 2t \\ R - r_{in} \geq \frac{l}{4} \end{cases} \tag{1}$$

The constraint restricts the expansion of the module, allowing it to unfold only in opposite directions during pressurization, resulting in a bending deformation of angle θ under the applied pressure P. Assuming the deformed module remains its original length along an axis on the side of constraint, offset by a distance of $r_{in} + \frac{t}{2}$ from the central axis, and the deformation follows the constant curvature (CC) approximation, it can be represented by an associate arc (indicated by a line) with the length l' and curvature $\frac{\theta}{l'}$ (Fig. 2C), where $l' = \theta(\frac{l}{\theta} + r_{in} + \frac{t}{2})$.

As shown in Fig. 2D, by having a clockwise rotation angle $\Delta\phi$ between two adjacent modules (from upper to lower), the designed actuator can achieve spatial deformation, where the inner line indicates its deformed shape (Fig. 2E). To further simplify the design, all stacking modules share the same inner radius r_{in}, wall thickness t, actuation pressure P and the manufacturing material. To this end, the kinematics model of bellow SPAs can be derived using the transformation matrix T_i^{i+1} mapping from the arc base frame $\{S_i\}$ to the tip frame $\{S_{i+1}\}$ of the i-th module, where

$$T_i^{i+1} = \begin{bmatrix} c\Delta\phi_i c\theta_i & -s\Delta\phi_i & c\Delta\phi_i s\theta_i & \frac{l'_i}{\theta_i} c\Delta\phi_i (1 - c\theta_i) \\ s\Delta\phi_i c\theta_i & c\Delta\phi_i & s\Delta\phi_i s\theta_i & \frac{l'_i}{\theta_i} s\Delta\phi_i (1 - c\theta_i) \\ -s\theta_i & 0 & c\theta_i & \frac{l'_i}{\theta_i} s\theta_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2}$$

in which c and s represent $\cos$ and $\sin$.

B. *Surrogate Model based on the FEM data set and ANN training*

The above kinematics model for bellow SPAs maps arc parameters to actuator position but doesn't define the transformation from design space to arc parameters (Fig. 3). Hence, a model connecting the design parameters (geometry and material properties) of a module to its deformation is needed. Fig. 3 shows the surrogate modeling process using a FEM model for data set generation, subsequently training an ANN to expedite computation for optimization.

I FEM Simulation and Dataset Collection

To generate a viable surrogate model of the bellow SPA, a data set should be obtained by sampling its feasible feature space. Four attribute geometry parameters and the actuation pressure are chosen as input to obtain a data set (Fig. 3A). The ranges and intervals of the selected input are set as given Table 2.

Table 2. Range and interval of FEM simulation input data for dataset collection.

Parameters	Minimum	Maximum	Interval
r_{in} (mm)	2	10	2
t (mm)	$r_{in}/4$	$r_{in}/3$	$r_{in}/24$
R (mm)	$r_{in} + t$	$2r_{in}$	$(r_{in} - t)/4$
l (mm)	$4t$	$4(R - r_{in})$	$R - r_{in} - t$
P (kPa)	0	10	0.5

Considering the limitation of the material customizability, the data set is material specified. Therefore, a new set of collection is required for a new material. The incompressible Neo-Hookean hyperelastic material model is applied to the material (Fig. 3B).

$$W = C_1(I_1 - 3) = \frac{\mu}{2}(I_1 - 3) \tag{3}$$

where $\mu = E/2(1 + \nu)$ is the shear modulus, I_1 is the first invariant of the right Cauchy-Green deformation tensor, E is Young's modulus and ν is the Poisson's ratio. Agilus30™ is used as the default material as it is a commonly used material for designing soft robots.[35] Material properties are obtained from a characterization work[36] and an official datasheet.[24]

A MATLAB® script is written to automatically create FEM simulations in COMSOL MultiPhysics® based on the input and extract simulation results as output. Thanks to the symmetry of the bellow geometry, the FEM simulation is performed on the half bellow module as a stationary study of the 3D solid mechanics model (top of Fig. 3C). The top end is fixed, and the other is connected to a cylinder to smooth the element deformation. A uniformly distributed load is applied to the inner surface colored in blue.

From the nonlinear simulation, deformation of the half bellow module and its von Mises stress distribution can be obtained (bottom of Fig. 3C). By using the displacement of a group of points at the end of the cylinder, the angular deflection θ of the bellow module can be calculated (see details

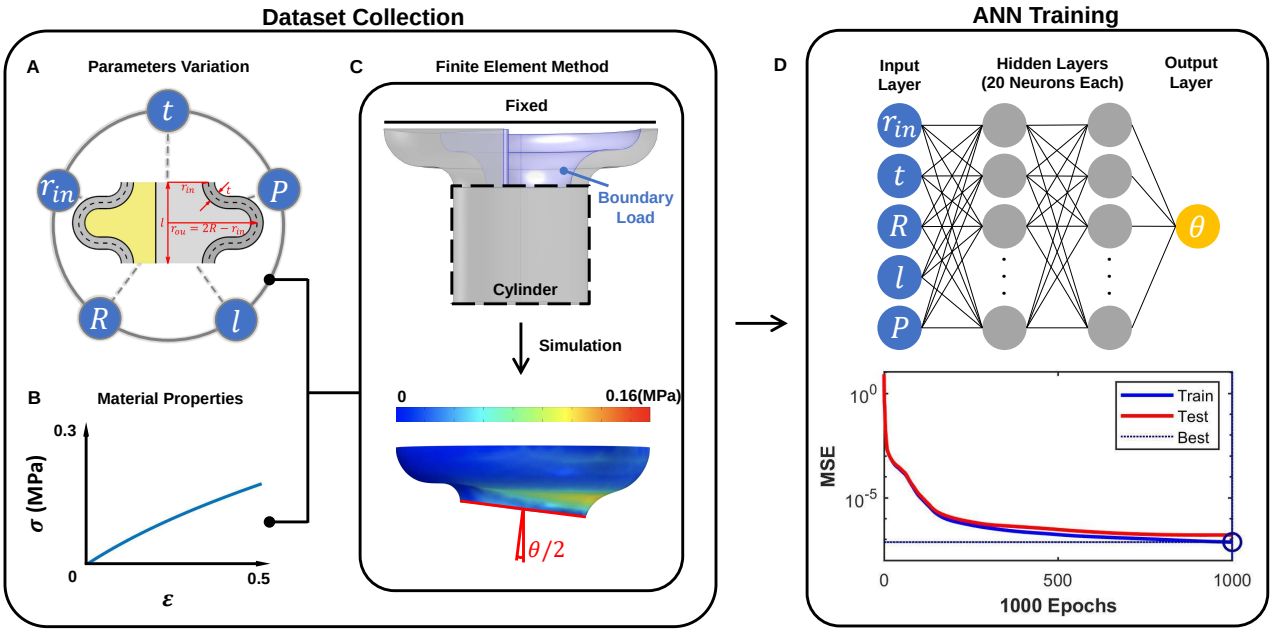


Figure 3. The process of generating the surrogate model. A) The design parameters to be varied in data set collection. B) The elastic properties of the material. C) Boundary conditions of the half bellow module in the FEM simulation from the cross-section perspective (top), and the deformed stress plot after simulation (bottom). D) The ANN used to train the surrogate model based on the data set (top), and the mean square error (MSE) of both training and test data in relation to the change of epochs (bottom).

of the FEM and material model in Section A in Supplementary Data).

II Surrogate Model Trained via an ANN

To generate the surrogate model, a feedforward ANN (top of Fig. 3D) was constructed with an input layer of 5 neurons for the input data (consisting of 4 geometric parameters and the actuation pressure based on Table 2, 2 hidden layers of 20 neurons each and an output layer of 1 neuron for the corresponding output data (angular deflection). The mean squared error (MSE) served as the performance function. The Bayesian Regularization backpropagation is used as the training function.

The pre-collected FEM simulation data set is randomly divided into training and test sets (80% and 20%, respectively) and trained until 1000 epochs have been reached. After training, the MSE reaches $< 10^{-6}$ for both training and test data (bottom of Fig. 3D). This pretrained network will be then used as the surrogate model in the form of $\theta = f(r_{in}, t, R, l, P)$ to quickly predict the angular deflection of the bellow SPA module for a given set of design parameters, allowing for efficient design exploration and optimization.

C. *Shape Matching Optimization*

Given a curve represented by a set of 3D ordered points $\{\boldsymbol{p}_k\}_{k=0}^{m}$, we aim to design a bellow SPA to deform from an unactuated straight line to a desired shape upon pressurization. Fig. 4 outlines our shape-matching process. Initially, A 3D PCC segmentation algorithm discretizes the curve into CC segments (Fig. 4A-C). Next, the actuator's design involves determining the optimal module parameters to match each CC segment (Fig. 4D, E). Notably, "segment" pertains to the desired shape's divided part, whereas "module" refers to the bellow SPA's building block. Each segment's shape is approximated using one or multiple identical modules.

PCC requires the curve to follow G^1 continuity, meaning two adjacent CC segments share an endpoint with aligned tangent vectors. The biarc method, frequently used for this,[37] requires two endpoints $\boldsymbol{P}_i$, $\boldsymbol{P}_e$ and their tangents $\boldsymbol{T}_i$, $\boldsymbol{T}_e$ (Fig. 4C). The challenge is determining the connection point $\boldsymbol{P}_c$ between two CC segments.

$$\boldsymbol{P}_c = \frac{d_2(\boldsymbol{P}_i + d_1\boldsymbol{T}_i) + d_1(\boldsymbol{P}_e - d_2\boldsymbol{T}_e)}{d_1 + d_2} \tag{4}$$

where $d_1 = |\boldsymbol{Q}_i - \boldsymbol{P}_i| = |\boldsymbol{Q}_i - \boldsymbol{P}_c|$ and $d_2 = |\boldsymbol{Q}_e -$

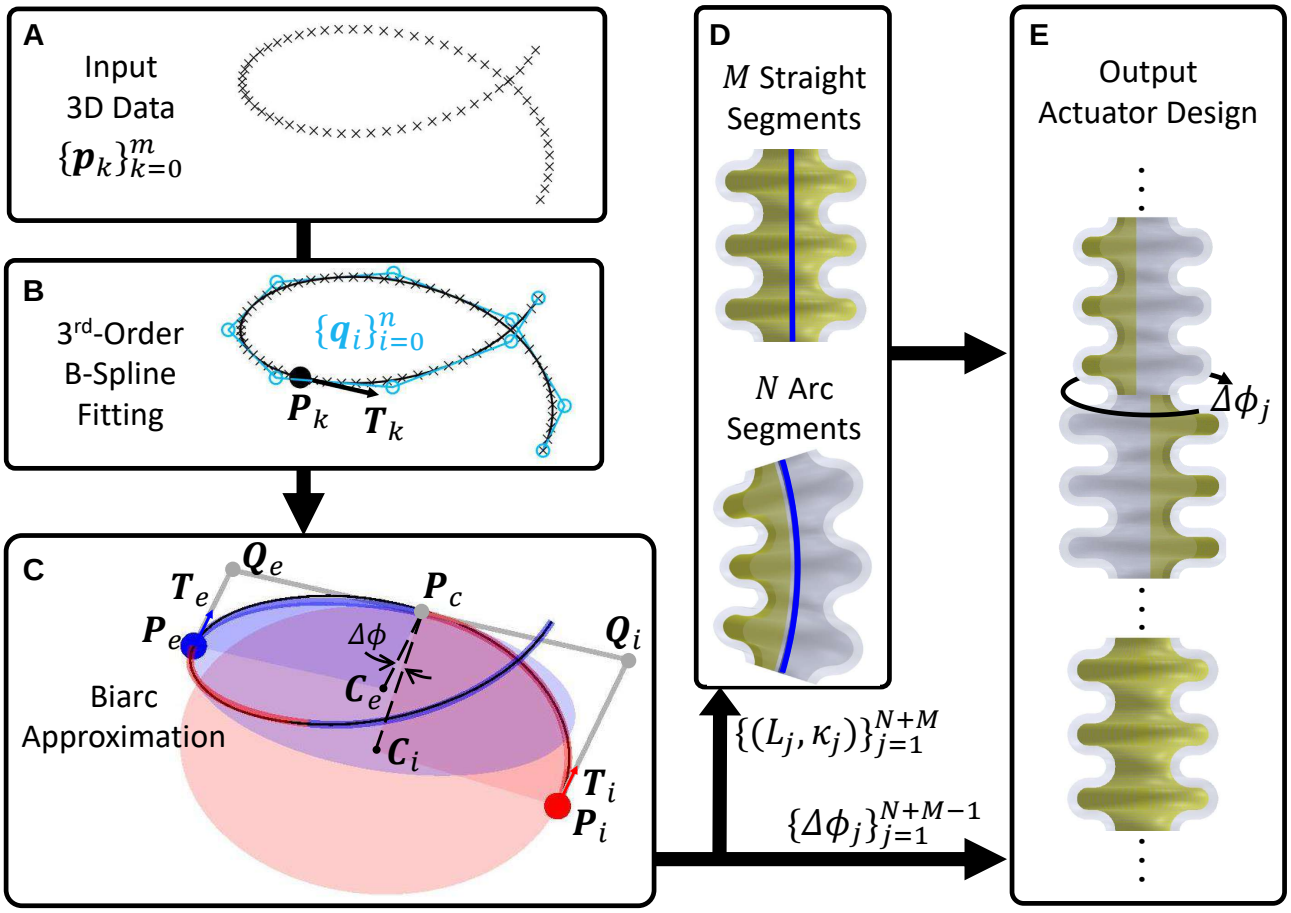


Figure 4. The shape-matching optimization process, explained in Shape Matching Optimization section in detail. A)-C) Shows the PCC segmentation algorithm, in which the input data $\{\boldsymbol{p}_k\}_{k=0}^m$ is first fitted by a third-order B-spline curve controlled by light blue control points $\{\boldsymbol{q}_i\}_{i=0}^n$, with projected point $\boldsymbol{P}_k$ and its tangent vector $\boldsymbol{T}_k$; then discretized into CC segments using the biarc approximation by interpolating between two end points $\boldsymbol{P}_i$, $\boldsymbol{P}_e$ and their tangent vectors $\boldsymbol{T}_i$, $\boldsymbol{T}_e$, based on the control points $\boldsymbol{Q}_i$, $\boldsymbol{Q}_e$ and the connection point $\boldsymbol{P}_c$; the rotation angle $\Delta\phi$ between two segments is $\angle \boldsymbol{C}_i\boldsymbol{P}_c\boldsymbol{C}_e$. D) Shows that for each CC segment represented by shape parameters (L_j, κ_j), a module will be designed to match its shape by stacking one or more identical modules. E) Shows the output actuator is designed by stacking the previous modules in order with rotation angles $\Delta\phi_j$ between segments.

$\boldsymbol{P}_e| = |\boldsymbol{Q}_e - \boldsymbol{P}_c|$, with $\boldsymbol{Q}_i$ and $\boldsymbol{Q}_e$ as biarc control points. The relationship $|\boldsymbol{Q}_e - \boldsymbol{Q}_i| = d_1 + d_2$ allows solving the biarc with $d_1 = d_2$. Furthermore, the rotation angle $\Delta\phi$ between two CC segments is determined by the angle between $\boldsymbol{C}_i\boldsymbol{P}_c$ and $\boldsymbol{C}_e\boldsymbol{P}_c$, where $\boldsymbol{C}_i$ and $\boldsymbol{C}_e$ are the respective circle centers.

The biarc approximation samples data points, starting from the initial point, seeking the longest biarc that approximates data within a preset tolerance. This process is iteratively repeated until all data are approximated.[38] Tangent vectors are derived using a third-order B-spline curve fitted to the original 3D data (Fig. 4B), enabling the calculation of projection points and their tangent vectors. This B-spline curve is defined by $n+1$ control points $\{\boldsymbol{q}_i\}_{i=0}^n$,

$$\boldsymbol{S}(t) = \sum_{i=0}^{n} B_{i,d}(t)\boldsymbol{q}_i \tag{5}$$

where $d = 3-1$ is the degree of curve, 3 is the order of curve, t is a sequence of non-decreasing uniform knots that $t_i = \frac{i-d}{n+1-d}$ for $d+1 \le i \le n$, and there is $d+1$ number of 0 and 1 at the beginning and end of the knot sequence, respectively. B is the basis function that is defined recursively,

$$B_{i,0}(t) = \begin{cases} 1, & t_i \le t < t_{i+1} \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

$$\begin{aligned} B_{i,j}(t) = &\frac{t - t_i}{t_{i+j} - t_i} B_{i,j-1}(t) \\ &+ \frac{t_{i+j+1} - t}{t_{i+j+1} - t_{i+1}} B_{i+1,j-1}(t) \end{aligned} \tag{7}$$

Assuming that the sample sequence of the input data points is $t_k = \frac{k}{m}$, the matrix $\hat{q}$, composed of the control points of the B-spline that fits the data can be obtained using the least-squares fitting,

$$\hat{\mathbf{q}} = ((\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T)\hat{\mathbf{p}} \tag{8}$$

where $\mathbf{A} = [B_{i,d}(t_k)]$ and $\hat{\mathbf{p}}$ is the matrix composed of all data points, and the number of control points $n+1$ can be determined by minimizing the error from input data points to the B-spline curve.[39,40] Then, all data points are projected to the B-spline curve to get the projected points $\{\boldsymbol{P}_k\}_{k=0}^m$ and their tangent vectors $\{\boldsymbol{T}_k\}_{k=0}^m$.

A number of CC segments are obtained after the PCC segmentation (Fig. 4D). If a segment curvature is $< 10^{-3}$, it is deemed straight. These segments are designed by stacking modules filled with internal constraints. For arc segments, a genetic algorithm determines the bellow SPA module design parameters for matching shapes by stacking identical modules. The selection of the genetic algorithm is informed by our previous work,[41] thanks to its compatibility with surrogate modeling and its capacity to effectively handle a large number of design variables.

The objective is to minimize the difference in arc parameters (arc length and curvature) between the deformed segment shape (composed of designed modules and predicted by a surrogate model) and the desired segment shape (see details in Section B in Supplementary Data). The final actuator is constructed by orderly stacking these modules, considering rotation angles between segments (Fig. 4E).

3. The Toolbox Implementation

To amplify the efficiency and user accessibility of our design framework, we have developed SPADA — an open-source design toolbox with a user-friendly GUI (Fig.5) — built on MATLAB® and COMSOL MultiPhysics®. Compared with our previously published bellow SPA design toolbox — designed solely for simulating bending actuator behaviors and taking approximately 74 minutes per simulation[31] — SPADA stands out by facilitating simulations for diverse deformations in just a few minutes and offering efficient end-to-end shape-matching optimization. Here, "end-to-end" refers to the process from the desired shape to the stereolithography files of the designed actuator for 3D printing.

The simulation component allows for the design of bellow SPAs by adjusting the modular geometric parameters, customizing assemblies, and modifying material properties. It utilizes background FEM simulation to analyze the behavior of modules and predict the actuator's configuration through forward kinematics. Additionally, it can generate material-specified data sets.

The optimization component can train a selected data set into an ANN to serve as a surrogate model. It takes as input a file containing the de-

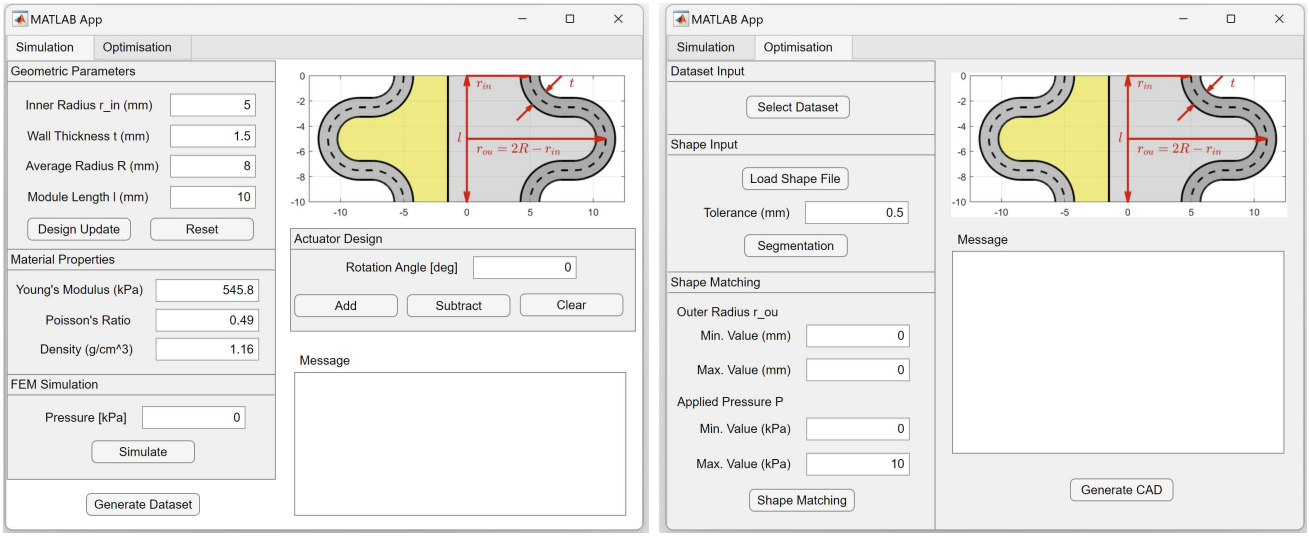


Figure 5. The graphic user interfaces of the toolbox SPADA, including the simulation (left) and optimization (right) components.

sired shape of the actuator, represented as a series of ordered 3D points. The 3D PCC segmentation algorithm segments the desired shape into CC segments, and an optimization algorithm based on a genetic algorithm and the surrogate model is used to find the optimal actuator design parameters that approximate the desired shape. It can also generate computer-aided design files of the designed actuator, ready for direct 3D printing.

See Section C in Supplementary Data for detailed instructions on how to use SPADA. The source code is available on a GitHub repository,[42] and a demonstration of the toolbox is provided in the Supplementary Video.

4. Results

In this section, SPADA was used to design three actuators that accurately match the predefined 2D shapes with root-mean-squared-errors (RMSEs) of 4.16, 2.70, and 2.51mm, respectively, hence validating the accuracy of the kinematics model, illustrating the efficacy of the shape-matching algorithm, and demonstrating its ability in achieving end-to-end from desired shapes to designed actuators for direct 3D printing. Furthermore, we harnessed the toolbox's potential in the design of 3D deformable actuators, specifically by designing an actuator according to an elephant trunk-inspired helical shape.

A. 2D Shape Matching: Letter Writing

In the 2D case of validating the shape-matching algorithm along with the kinematics model, and also demonstrating the ability of end-to-end from the desired shapes to 3D printed actuators, SPADA was used to design three actuators that match the shape of the letters "S", "R", and "L", which is the acronym of Soft Robotics Lab (Fig. 6).

The shape for each letter is represented by a series of ordered 3D coordinates, imported into SPADA as a ".mat" file. The segmentation results for each letter are highlighted in Fig. 6A-C. The letter "S" is segmented into two arc segments with the same length and curvature. The letter "R" is segmented into two straight segments and three different arc segments. The letter "L" is segmented into two straight segments and one arc segment. All the lower bound values of the outer radius were set to be 9mm for three actuator optimization for ease of manufacturing.

After optimization, the optimal design parameters are labeled on the schematics of actuators designed by those parameters in Fig. 6D and E. The surrogate model used was provided with the toolbox, that was trained by a provided FEM data set of the default material Agilus30™ (see information of the FEM and material parameters in Section A in Supplementary Data). The prototypes were 3D printed on a Stratasys J735™ using the CAD files generated by SPADA.

The shape of the actuated prototypes was compared to the initial desired shape in Fig. 6G by overlapping them in the x-y plane. The respective RMSEs between the desired shapes and actuators' real shapes are 4.16, 2.70, and 2.51mm, validating and demonstrating our shape-matching algorithm (refer to Section D.I in Supplementary Data for details of the experiments, and see the prototypes taking the defined shapes in the Supplementary Video available on the project's GitHub page[42]).

B. 3D Shape Matching: Elephant Trunk-Inspired Shape

The previous 2D case of shape-matching validates the efficacy of our end-to-end design framework, given its independence from gravitational effects (Fig. 7). Yet, in the 3D scenario, gravity is unavoidable. Furthermore, gravity acts unidirectionally while the deformation of actuators significantly varies in space, making it exceedingly challenging to incorporate gravity's impact into the modeling process, particularly in modular approaches. This complexity explains why nearly all shape-matching design frameworks sidestep considering gravitational effects.[12,16,17] While Jiang et al.[15] did incorporate gravity into their analytical model, they were only able to roughly approximate its influence for two types of segments and did not evaluate it in the real world.

We recognize that shape-matching, on its own, can not offer a comprehensive solution for tasks involving gravity and contact. Yet, our design framework and toolbox still provide valuable insights for actuator design, even without considering these factors, thanks to the adaptability of

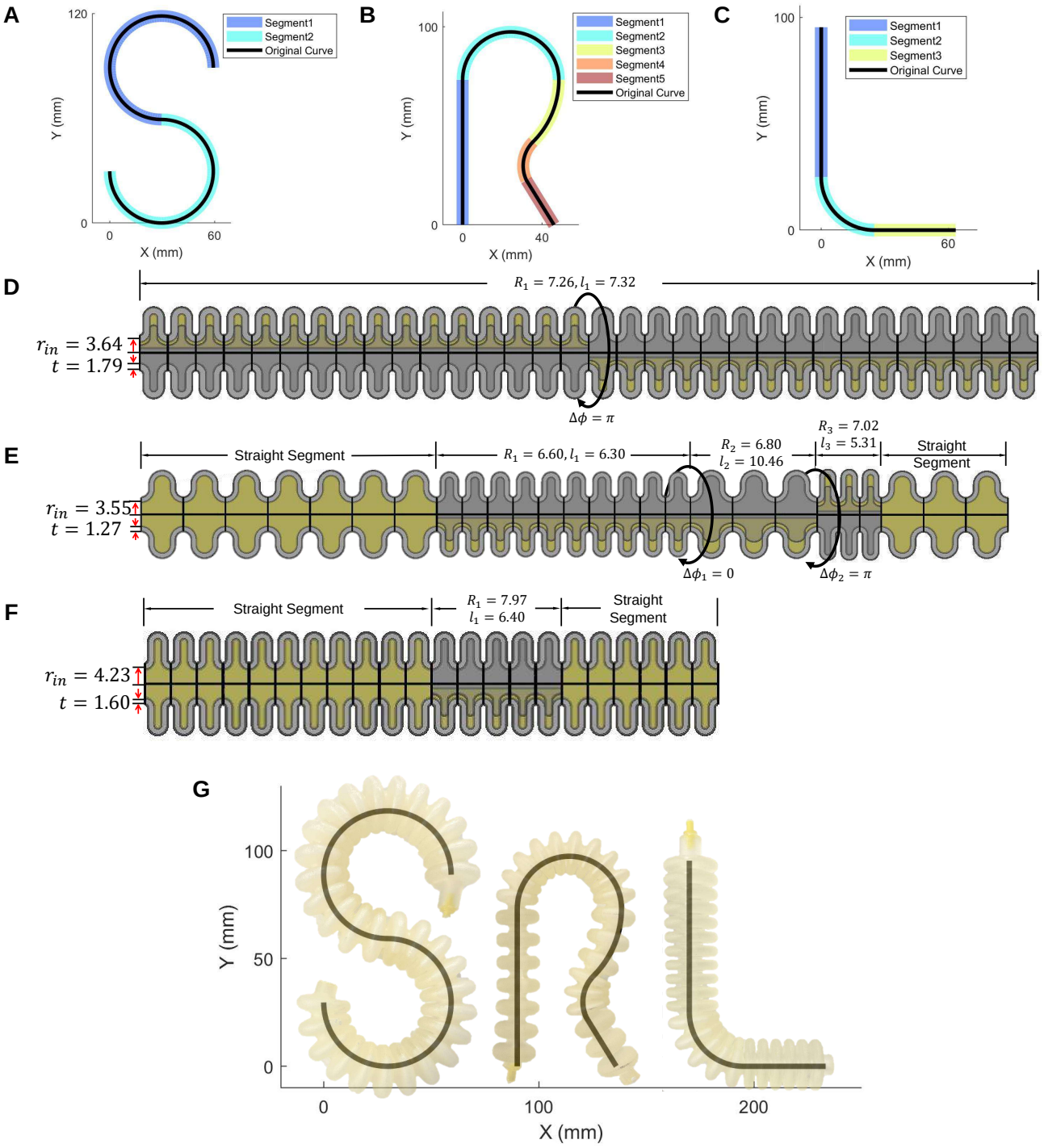


Figure 6. The 2D shape-matching case: the input shapes of letters "SRL" are segmented in A)-C). D)-F) Are the actuator schematics that match the input shapes and are designed by optimal parameters. G) Shows the deformed prototypes compared with the input shapes.

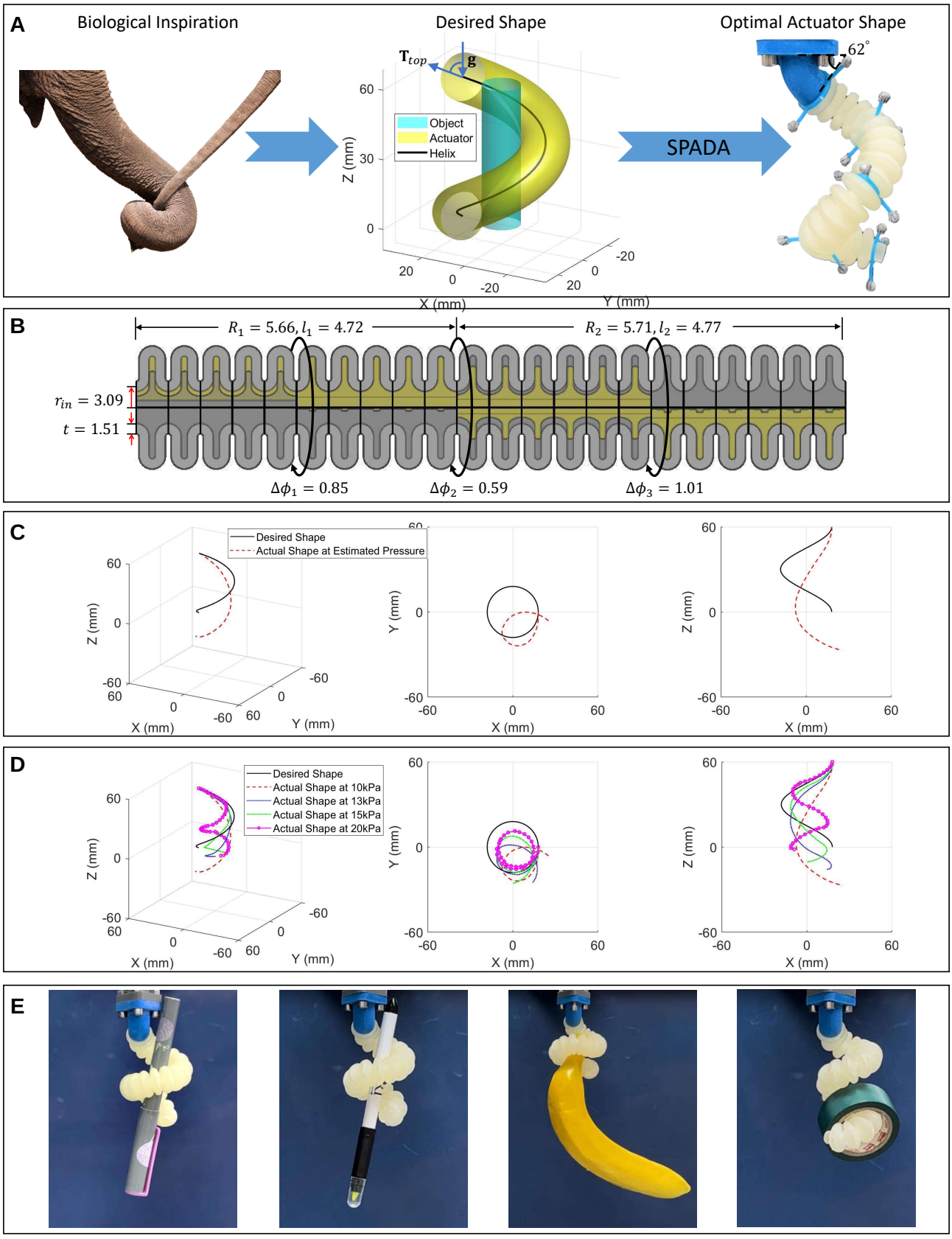


Figure 7. The 3D shape-matching case: A) Depicts the process of inspiration drawing from an elephant trunk (edited image © Shutterstock, accessed on August 1, 2023), acquiring the helical shape, and subsequent design of the actuator using our toolbox SPADA. B) Showcases a schematic of the actuator with optimal design parameters. C) Provides a comparison between the desired helical shape and the actual shape of the actuator under the pressure estimated by the toolbox. D) Compares the desired shape and the actual ones of the actuator under escalated pressure. E) Demonstrates the actuator's versatility in handling and grasping various objects.

soft actuators. The design of actuators to assume a helical shape, emulating the versatile and secure grasping behavior of an elephant's trunk, is a prime example among the research community.[6,21,43]

Hence, both to demonstrate the potential of the toolbox for employing 3D shape-matching in guiding actuator design and to quantitatively assess gravity's impact on the 3D shape-matching scenario, we delve into an end-to-end example of designing a helical actuator, inspired by the structure of an elephant's trunk, for the purpose of grasping irregular objects, as shown in Fig. 7.

A right-handed helix, with x, y, z coordinates represented by the mathematical expression $(a\cos t,\ a\sin t,\ ht)$ where t is evenly spaced between 0 and 2π, is utilized to embody the helical configuration inspired by the grasping movement of an elephant's trunk. By presuming that the object targeted for grasping fits within a cylindrical dimension of 16mm diameter and 120mm height, and further constraining the actuator's outer radius to be approximately 10mm, the helix parameters $a = 18$mm and $h = 60$mm can be derived for a power grasp.

The helical shape, consisting of a sequential of ordered 3D points, is imported into SPADA. An actuator is then designed to match this shape using the toolbox framework while disregarding gravitational effects. Fig. 7B showcases the design parameters of the actuator, and the toolbox estimates the actuation pressure to be 9.85kPa (which is approximated to 10kPa in experiments due to the hardware limitations). To assess the discrepancy between the desired shape and the actual shape of the pressurized actuator under the effect of gravity, we executed a series of experiments. The top end of the actuator is fixed to a connector with an angle of 62° (Fig. 7A). This angle corresponds to the divergence between the tangent vector at the helix's top end and the gravity axis.

We first pressurized the actuator to 10kPa, a value estimated by the toolbox as optimal for achieving the helical shape. Fig. 7C illustrates the deviation between the desired helical shape and the actual configuration of the actuator at this pressure. The RMSE is calculated to be 25.64mm.

Subsequently, we escalated the actuation pressure to 13kPa, 15kPa, and 20kPa. Fig. 7D compares the desired shape and the actuator's actual shapes under these increased pressure values. The calculated root-mean-square errors for these pressures were 19.08mm, 18.69mm, and 23.77mm, respectively. Moreover, as observed from the XY and XZ perspectives in Fig. 7C and D, it's evident that as pressure increases, the central point of the actual shape (XY view) approaches that of the targeted shape. Concurrently, the number of convolutions in the actual shape (XZ view) increases, while the height of the actual shape along the z-axis decreases. This phenomenon can be attributed to the escalating pressure that not only counteracts gravitational effects but also enhances the curvature perpendicular to the direction of gravity. This, in turn, induces an increase in the number of helical turns.

However, despite the error caused by gravity, it is still possible to exploit behaviors arising from the design and interaction with the object to achieve a successful grasp. The grasping experiments were performed to explore the potential of the designed actuator for handling various objects, as shown in Fig.7E. Refer to Section D.II in Supplementary Data for details of the grasping experiments and watch the prototype grasping objects in the Supplementary Video.

5. Conclusion

In this work, an end-to-end design framework for the shape-matching of bellow SPAs is proposed and implemented as an open-source design toolbox, SPADA. A model leverages FEM-simulated module deformation and PCC approximation to predict the kinematics of actuators. Surrogate modeling involving ANN training on a FEM data set is applied to speed up computation. A 3D PCC segmentation algorithm approximates the desired curve by dividing it into CC segments.

An optimization algorithm, grounded on a genetic algorithm and a surrogate model, determines the optimal design parameters of the actuator to align with the shape of these segments. The toolbox, based on the proposed design framework, has proven its end-to-end capability in designing actuators to accurately match 2D shapes, while also demonstrating its potential to design deformable actuators in 3D space.

Overall, this design framework can be general-

ized to other soft actuators, such as tendon-driven soft actuators. Users can also specify more material properties for the actuator and use the toolbox FEM simulation function to create a data set for training. A GitHub repository has been created with the toolbox and locations for users to upload self-acquired data sets for co-design. In the future, the authors will maintain and update the toolbox with consideration of the gravity effect and environmental parameters (such as underwater conditions). Add-ons for protocols to design soft actuators for specific tasks will also be investigated, such as soft grippers, soft actuators for surgical operations, and soft crawlers for locomotion.

Acknowledgements

The authors thank Dr Alessandro Albini (University of Oxford) for helping with the experimental setup and Mr Yuwen Chen (University of Oxford) for advice on optimization algorithms.

Authorship Confirmation Statement

Y.Y., L.H. and P.M. conceived the concept and wrote the article. Y.Y. performed the simulations and optimization, carried out experiments and interpreted the results. L.H. and P.M. directed the project. All authors commented on the article.

Author Disclosure Statement

No competing financial interests exist.

Funding Statement

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) Grant EP/V000748/1.

References

[1] Ilievski F, Mazzeo AD, Shepherd RF, Chen X, Whitesides GM. Soft Robotics for Chemists. Angewandte Chemie International Edition. 2011;50(8):1890-5. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/anie.201006464.

[2] Shintake J, Cacucciolo V, Floreano D, Shea H. Soft Robotic Grippers. Advanced Materials. 2018;30(29):1707035. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/adma.201707035.

[3] Drotman D, Ishida M, Jadhav S, Tolley MT. Application-Driven Design of Soft, 3-D Printed, Pneumatic Actuators With Bellows. IEEE/ASME Transactions on Mechatronics. 2019;24(1):78-87.

[4] Zhang Y, Yang D, Yan P, Zhou P, Zou J, Gu G. Inchworm Inspired Multimodal Soft Robots With Crawling, Climbing, and Transitioning Locomotion. IEEE Transactions on Robotics. 2022;38(3):1806-19.

[5] Marchese AD, Rus D. Design, kinematics, and control of a soft spatial fluidic elastomer manipulator. The International Journal of Robotics Research. 2016;35(7):840-69. Available from: https://doi.org/10.1177/0278364915587925.

[6] Guan Q, Sun J, Liu Y, Wereley NM, Leng J. Novel Bending and Helical Extensile/Contractile Pneumatic Artificial Muscles Inspired by Elephant Trunk. Soft Robotics. 2020;7(5):597-614. PMID: 32130078. Available from: https://doi.org/10.1089/soro.2019.0079.

[7] Zolfagharian A, Mahmud MAP, Gharaie S, Bodaghi M, Kouzani AZ, Kaynak A. 3D/4D-printed bending-type soft pneumatic actuators: fabrication, modelling, and control. Virtual and Physical Prototyping. 2020;15(4):373-402. Available from: https://doi.org/10.1080/17452759.2020.1795209.

[8] Tawk C, Alici G. A Review of 3D-Printable Soft Pneumatic Actuators and Sensors: Research Challenges and Opportunities. Advanced Intelligent Systems. 2021;3(6):2000223. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/aisy.202000223.

[9] Xavier MS, Tawk CD, Zolfagharian A, Pinskier J, Howard D, Young T, et al. Soft Pneumatic Actuators: A Review of Design, Fabrication, Modeling, Sensing, Control and Applications. IEEE Access. 2022;10:59442-85.

[10] Connolly F, Polygerinos P, Walsh CJ, Bertoldi K. Mechanical Programming of Soft Actuators by Varying Fiber Angle. Soft Robotics. 2015;2(1):26-32. Available from: https://doi.org/10.1089/soro.2015.0001.

[11] Wang T, Ge L, Gu G. Programmable design of soft pneu-net actuators with oblique chambers can generate coupled bending and twisting motions. Sensors and Actuators A: Physical. 2018;271:131-8. Available from: https://www.sciencedirect.com/science/article/pii/S0924424717316977.

[12] Kan Z, Pang C, Zhang Y, Yang Y, Wang MY. Soft Actuator with Programmable Design: Modeling, Prototyping, and Applications. Soft Robotics. 2022;9(5):907-25. PMID: 35005997. Available from: https://doi.org/10.1089/soro.2020.0148.

[13] Ellis DR, Venter MP, Venter G. Generative Design Procedure for Embedding Specified Planar Behavior in Modular Soft Pneumatic Actuators. Soft Robotics. 2022;9(3):552-61. PMID: 34550802. Available from: https://doi.org/10.1089/soro.2020.0013.

[14] Armanini C, Boyer F, Mathew AT, Duriez C, Renda F. Soft Robots Modeling: A Structured Overview. IEEE Transactions on Robotics. 2023;39(3):1728-48.

[15] Jiang C, Wang D, Zhao B, Liao Z, Gu G. Modeling and inverse design of bio-inspired multi-segment pneu-net soft manipulators for 3D trajectory motion. Applied Physics Reviews. 2021;8(4):041416. Available from: https://doi.org/10.1063/5.0054468.

[16] Connolly F, Walsh CJ, Bertoldi K. Automatic design of fiber-reinforced soft actuators for trajectory matching. Proceedings of the National Academy of Sciences. 2017;114(1):51-6. Available from: https://www.pnas.org/doi/abs/10.1073/pnas.1615140114.

[17] Singh G, Krishnan G. Designing Fiber-Reinforced Soft Actuators for Planar Curvilinear Shape Matching. Soft Robotics. 2020;7(1):109-21.

[18] Liu L, Zhang J, Liu G, Zhu Z, Hu Q, Li P. Three-Fingered Soft Pneumatic Gripper Integrating Joint-Tuning Capability. Soft Robotics. 2022;9(5):948-59. PMID: 34705563. Available from: https://doi.org/10.1089/soro.2021.0082.

[19] Uppalapati NK, Krishnan G. Design and Modeling of Soft Continuum Manipulators Using Parallel Asymmetric Combination of Fiber-Reinforced Elastomers. Journal of Mechanisms and Robotics. 2020 10;13(1):011010. Available from: https://doi.org/10.1115/1.4048223.

[20] Marchese AD, Onal CD, Rus D. Autonomous Soft Robotic Fish Capable of Escape Maneuvers Using Fluidic Elastomer Actuators. Soft Robotics. 2014;1(1):75-87. PMID: 27625912. Available from: https://doi.org/10.1089/soro.2013.0009.

[21] Hu W, Alici G. Bioinspired Three-Dimensional-Printed Helical Soft Pneumatic Actuators and Their Characterization. Soft Robotics. 2020;7(3):267-82. PMID: 31687877. Available from: https://doi.org/10.1089/soro.2019.0015.

[22] Shepherd RF, Stokes AA, Nunes RMD, Whitesides GM. Soft Machines That are Resistant to Puncture and That Self Seal. Advanced Materials. 2013;25(46):6709-13. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/adma.201303175.

[23] Udupa G, Sreedharan P, Sai Dinesh P, Kim D. Asymmetric Bellow Flexible Pneumatic Actuator for Miniature Robotic Soft Gripper. Journal of Robotics. 2014;2014(902625).

[24] Agilus30 Materials Data Sheet; 2020. [Last accessed: July 2, 2023]. Available from: https://www.stratasys.com/-/media/files/material-spec-sheets/mds_pj_agilus30_1020a.pdf.

[25] Xavier MS, Fleming AJ, Yong YK. Finite Element Modeling of Soft Fluidic Actuators: Overview and Recent Developments. Advanced Intelligent Systems. 2021;3(2):2000187. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/aisy.202000187.

[26] Chen F, Wang MY. Design Optimization of Soft Robots: A Review of the State of the Art. IEEE Robotics & Automation Magazine. 2020;27(4):27-43.

[27] Pinskier J, Howard D. From Bioinspiration to Computer Generation: Developments in Autonomous Soft Robot Design. Advanced Intelligent Systems. 2022;4(1):2100086. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/aisy.202100086.

[28] Zolfagharian A, Durran L, Gharaie S, Rolfe B, Kaynak A, Bodaghi M. 4D printing soft robots guided by machine learning and finite element models. Sensors and Actuators A: Physical. 2021;328:112774. Available from: https://www.sciencedirect.com/science/article/pii/S0924424721002375.

[29] Liu J, Low JH, Han QQ, Lim M, Lu D, Yeow CH, et al. Simulation Data Driven Design Optimization for Reconfigurable Soft Gripper System. IEEE Robotics and Automation Letters. 2022;7(2):5803-10.

[30] Paik J. Soft robot design methodology for 'push-button' manufacturing. Nature Reviews Materials. 2018;3:81-3.

[31] Yao Y, He L, Maiolino P. A Simulation-Based Toolbox to Expedite the Digital Design of Bellow Soft Pneumatic Actuators. In: 2022 IEEE 5th International Conference on Soft Robotics (RoboSoft); 2022. p. 29-34.

[32] Smith L, MacCurdy R. SoRoForge: End-to-End Soft Actuator Design. IEEE Transactions on Automation Science and Engineering. 2023;20(3):1475-86.

[33] Moseley P, Florez JM, Sonar HA, Agarwal G, Curtin W, Paik J. Modeling, Design, and Development of Soft Pneumatic Actuators with Finite Element Method. Advanced Engineering Materials. 2016;18(6):978-88. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/adem.201500503.

[34] Navarro SE, Navez T, Goury O, Molina L, Duriez C. An Open Source Design Optimization Toolbox Evaluated on a Soft Finger; 2023.

[35] Stano G, Percoco G. Additive manufacturing aimed to soft robots fabrication: A review. Extreme Mechanics Letters. 2021;42:101079. Available from: https://www.sciencedirect.com/science/article/pii/S2352431620302662.

[36] Abayazid FF, Ghajari M. Material characterisation of additively manufactured elastomers at different strain rates and build orientations. Additive Manufacturing. 2020;33:101160. Available from: https://www.sciencedirect.com/science/article/pii/S2214860420305327.

[37] Juckett R. Biarc Interpolation;. [Last accessed: July 2, 2023]. Available from: https://www.ryanjuckett.com/biarc-interpolation/.

[38] Lin J. Biarc approximation of space data. Journal of Fujian University of Technology. 2008;6(6). Available from: http://www.cqvip.com/qk/83823a/200806/29158784.htm.

[39] Eberly D. Least-Squares Fitting of Data with B-Spline Curves; 2005. [Last accessed: July 2, 2023]. Available from: https://www.geometrictools.com/Documentation/BSplineCurveLeastSquaresFit.pdf.

[40] Hunyadi L. B-splines; 2010. [Last accessed: July 2, 2023]. Available from: https://uk.mathworks.com/matlabcentral/fileexchange/27374-b-splines.

[41] Yao Y, Chen Y, He L, Maiolino P. Design Optimization for Bellow Soft Pneumatic Actuators in Shape-Matching. In: 2023 IEEE International Conference on Soft Robotics (RoboSoft); 2023. p. 1-7.

[42] Yao Y. SPADA; 2023. [Last accessed: July 2, 2023]. https://github.com/ORI-SRL/SPADA.git.

[43] Martinez RV, Branch JL, Fish CR, Jin L, Shepherd RF, Nunes RMD, et al. Robotic Tentacles with Three-Dimensional Mobility Based on Flexible Elastomers. Advanced Materials. 2013;25(2):205-12. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/adma.201203002.

4.4 Supporting Information

4.4.1 Toolbox Repository

The open-source toolbox, along with its installation and setup instructions, can be found here: `https://github.com/ORI-SRL/SPADA.git`

4.4.2 Supporting Video

A video demonstrating how to use the toolbox is provided here: `https://github.com/ORI-SRL/SPADA/blob/main/SupplementaryVideo.mp4`

4.4.3 Supplementary Data

The supplementary data of this paper is attached on the following pages.

SUPPLEMENTARY DATA

October 14, 2025

A The FEM Simulation and Material Model

The FEM simulation is performed as a stationary study of the 3D solid mechanics model in COMSOL MultiPhysics® 5.6a. The half bellow module is modeled with the same parameters in Table. 1, with the incompressible Neo-Hookean hyperelastic material model in Eq. 3, factoring in uniaxial deformation.

The rationale behind the material model selection is twofold:

(1) It is a simple material model that only requires a shear modulus, which can be approximated by Young's modulus and Poisson's ratio, making it easy for users to find parameters for their own materials.

(2) This model offers a satisfactory agreement for lower strain values, typically below 50% as reported [25]. Furthermore, based on our previous investigations [31], the mechanical strain for bellow-SPAs primarily remains under 50% for an actuation pressure range of 0-10kPa. Consequently, the neo-Hookean model has been deemed an appropriate choice.

Regarding material properties, we've set Poisson's ratio to 0.49, and the material density ρ = 1.16 g/cm^3 is obtained from the official data sheet [24]. Unable to conduct direct experimental characterization of the material, we derived Young's modulus by linear fitting the Ogden model data of Agilus30™ , as presented in the work of Abayazid and Ghajari [36], within a strain range of 0-1. This range was chosen as the data exhibited a consistent slope up to this point, with a noticeable change in slope for strains exceeding 1. As we couldn't access the original experimental data from their work, our comparison between the neo-Hookean model and their Ogden model for the given strain range yielded a Root Mean Square Error (RMSE) of 3.39% as indicated in Fig.S1 below.

Furthermore, the accuracy of the kinematic shape-matching is affected by the accuracy of the material model. From the experimental results of the 2D shape-matching example, it is possible to notice that despite the discrepancy between the two material models, we achieved a very good match.

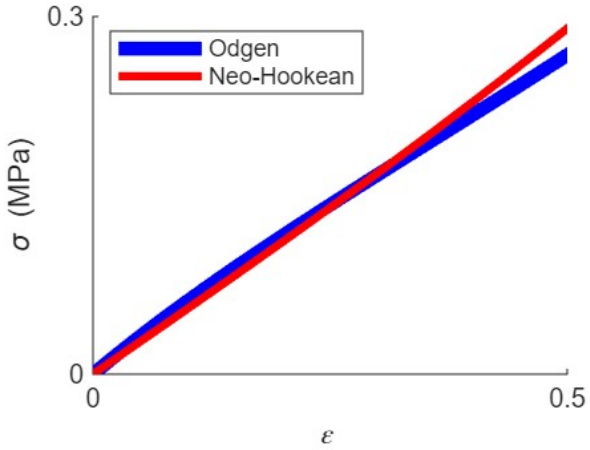


Figure S1: The comparison between the Ogden model data (from the work of Abayazid and Ghajari) and the Neo-Hookean model (using the linearly fitted Young's Modulus) for Agilus30™ in the strain range of 0-0.5.

In the FEM model, the cylinder used to smooth its element deformation has a radius of $r_{in} + \frac{t}{2}$ and a length of l connected to the smaller end of the half bellow using the same material model. The pressure is modeled as the uniformly distributed load on the inner surface of the half bellow, ramped by intervals of 0.5 KPa. The mesh is set as the default element size of "Finer". To allow for large deformation of elements, the geometric nonlinearity is included, with "Constant Newton" and "Anderson acceleration" set as the nonlinear simulation method.

Assuming the plane where the cylinder is connected to the bellow is parallel to the plane at the other end of the cylinder, the angular deflection of the half bellow is the angle between this plane and the plane at the fixed end of the half bellow. Therefore, the angle can be calculated by the displacement of the top point E_1 and the bottom point E_2 in the plane at the free end of the cylinder (as shown in Fig. S2(A). If the coordinate of E_1, E_2 are (x_1, y_1, z_1) and (x_2, y_2, z_2), respectively. Then the angular deflection of the module can be calculated by

$$\theta = \arccos \frac{z_1 - z_2}{\sqrt{(z_1 - z_2)^2 + (x_1 - x_2)^2}} \tag{S1}$$

Additionally, the FEM simulated stress distribution, angular deflection and the strain distribution of the bellow SPA model with default design parameters in relation to the change of pressure are shown in Fig. S2.

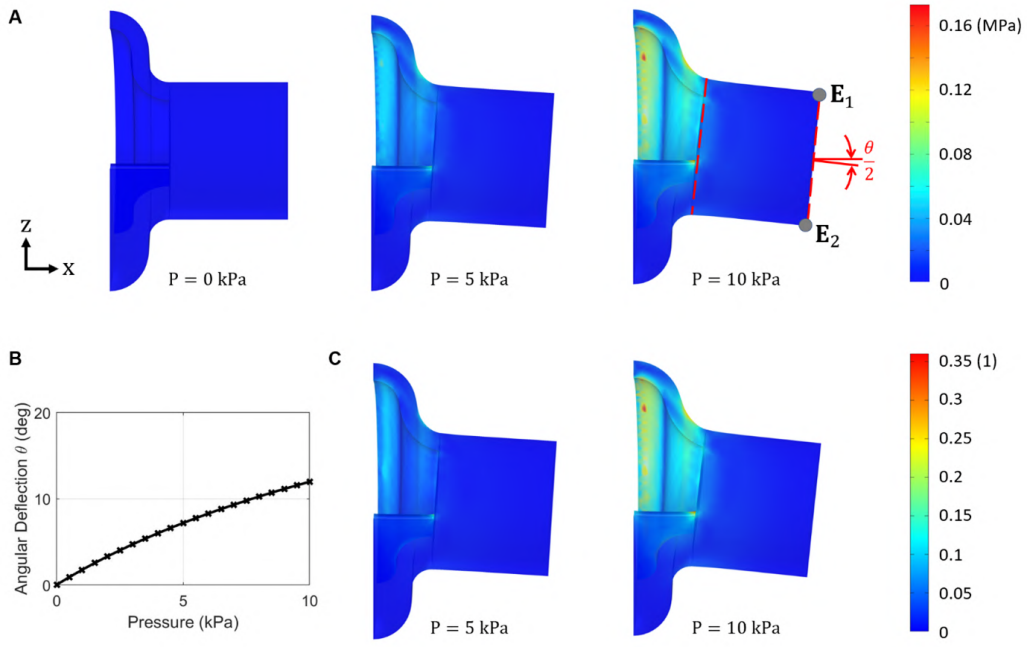


Figure S2: The detailed FEM simulation results of the bellow SPA module with the default geometric parameters (Table. 1) and the default material Agilus30™ . (A) The von Mises stress distribution of half FEM model changes with increased pressure. (B) The module's angular deflection θ predicted by the FEM model changes with increased pressure. (C) The first principal strain distribution of half FEM model changes with increased pressure.

B The Optimization Algorithm

Numerous optimization algorithms are available, as detailed in the review [26]. Based on our prior investigation into optimization algorithms suitable for bellow-SPAs shape-matching design [41], we identified the genetic algorithm as a prime choice for this work. This choice was influenced by its compatibility with surrogate modeling and its efficiency in handling large design variable dimensions.

Despite being provided in our previous work, we revisit the optimization algorithm here by integrating a genetic algorithm, a surrogate model, and the symbols introduced in this study. Assuming there are in total N unique arc segments with length and curvature pairs $[(L_1, \kappa_1), (L_2, \kappa_2), \ldots, (L_N, \kappa_N)]$ different from each other, there will be N different bellow SPA modules, each approximating one segments.

We wish to determine their shared parameters $\mathbf{x} = [r_{in}, t, P] \in \mathbb{R}^3_+$ and a set of respective geometric parameters $\mathbf{y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N] = [(R_1, l_1), (R_2, l_2),$

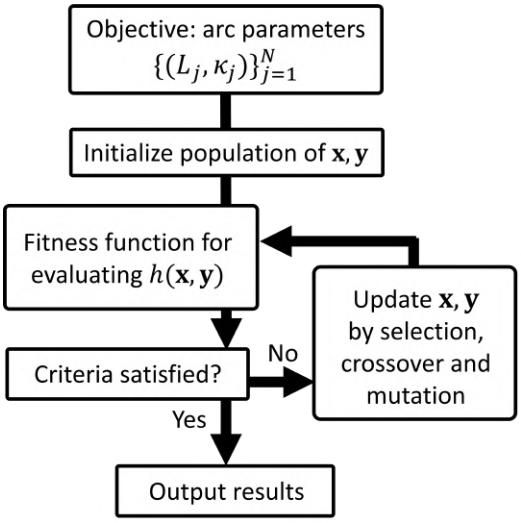


Figure S3: The optimization algorithm of designing bellow SPAs for shape-matching.

$\ldots, (R_N, l_N)] \in \mathbb{R}_+^{2\times N}$. The optimization objective is to minimize discrepancies in arc parameters (arc length and curvature) between the desired shape $\{(L_j, \kappa_j)\}_{j=1}^{N}$ and the deformed shape of the actuator designed by $\mathbf{x}$, $\mathbf{y}$, which is constructed from designed modules and predicted by a surrogate model. We have a total of $2N + 3$ optimization variables for one actuator, as shown in Fig. S3.

The constraints are specified by the maximum actuation pressure $P_{\max}$, the maximum curvature of segments $\kappa_{\max}$, the segment length and the inherent design constraints of the bellow SPA module. The problem is formulated as follows:

$$\min h(\mathbf{x}, \mathbf{y}) := \frac{1}{N}\sum_{j=1}^{N} d_j(\mathbf{x}, \mathbf{y}_j) \tag{S2}$$

$$\text{s.t.} \begin{cases} 0 \le P \le P_{\max} \\ 0 < r_{in} \le \frac{1}{\kappa_{\max}} \\ \frac{R_j}{4} \le t \le \frac{r_{in}}{2} \\ r_{in} + t \le R_j \le \min(\frac{1}{2\kappa_j} + \frac{r_{in}}{2}, 2r_{in}) \\ 2t < l_j < \min(L_j, 4(R_j - r_{in})), \quad \forall j \in [N] \end{cases} \tag{S3}$$

where

$$d_j(\mathbf{x}, \mathbf{y}_j) = w_1|L_j - n_j l'_j| + w_2\left|\kappa_j - \frac{\theta_j}{l'_j}\right| \tag{S4}$$

in which $l'_j = \theta_j(\frac{l_j}{\theta_j} + r_{in} + \frac{t}{2})$ is module's deformed length, $n_j = \left[\frac{L_j}{l'_j}\right]$ is the

number of identical modules in the j-th segment, and $w_1 = \frac{1}{L_j}$, $w_2 = \frac{1}{\kappa_j}$ are the weighting factors used to balance the difference of units. $\theta_j = f(r_{in}, t, P, R_j, l_j)$ is obtained from the surrogate model trained before.

Beyond the default settings of genetic algorithms in MATLAB®, we established stopping criteria for either reaching 500 iterations or the objective function dropping below 0.02. If a feasible point is found, the optimization outputs the shared parameters $[r_{in}, t, P]$ and the respective parameters $[R, l]$ for each segment.

C SPADA User Guide

I The Simulation Component

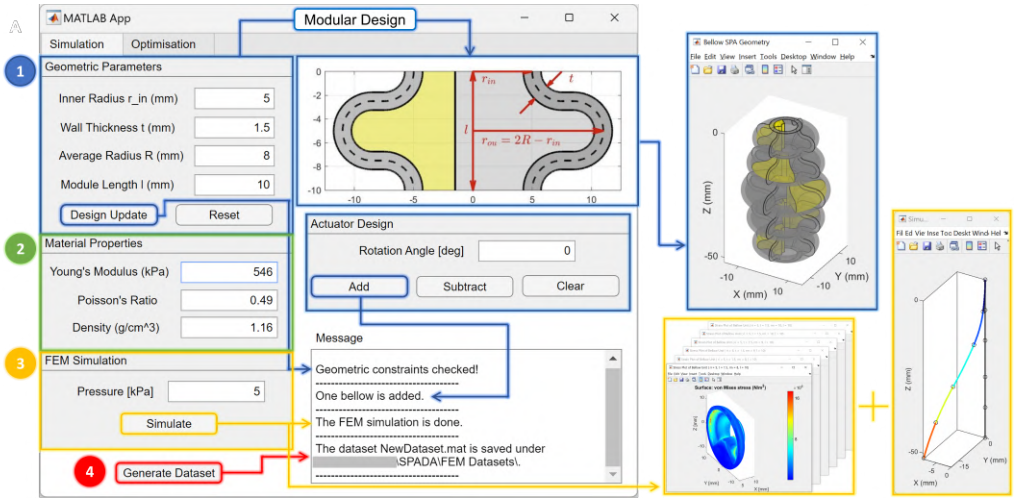


Figure S4: The GUI and usage instructions of the simulation component in SPADA (information for each step is highlighted in the same color as the step number icon)

The interface of the simulation component is shown in Fig. S4.

(1) The "Geometric Parameters" section allows entering four geometric parameters of the bellow SPA module. The "Design Update" button checks geometric constraints and updates the 2D schematic based on the input, while the "Reset" button resets them all to the default values. The "Actuator Design" section constructs the actuator in a modular approach (an example is shown in a blue box on the right side). The "Add" button stacks modules based on the current geometric parameters and orientation angles; "Subtract" removes the last added module, while "clear" removes all.

(2) The ”Material Properties” section allows for defining the material of the actuator by its Young’s modulus, Poisson’s ratio and density.

(3) The ”FEM Simulation” section takes the input of the actuation pressure; the ”Simulation” button launches COMSOL MultiPhysics® in the background and analyzes the designed actuator using FEM. It simulates the behavior of each unique module and predicts the configuration of the whole actuator with forward kinematics (examples are shown in the first and second yellow boxes on the right side, respectively).

(4) The ”Generate Dataset” button uses FEM to collect a dataset as explained in the paper’s Section 2.B.I based on the input material properties. A dataset of Agilus30™ is already collected and provided to users.

II The Optimization Component

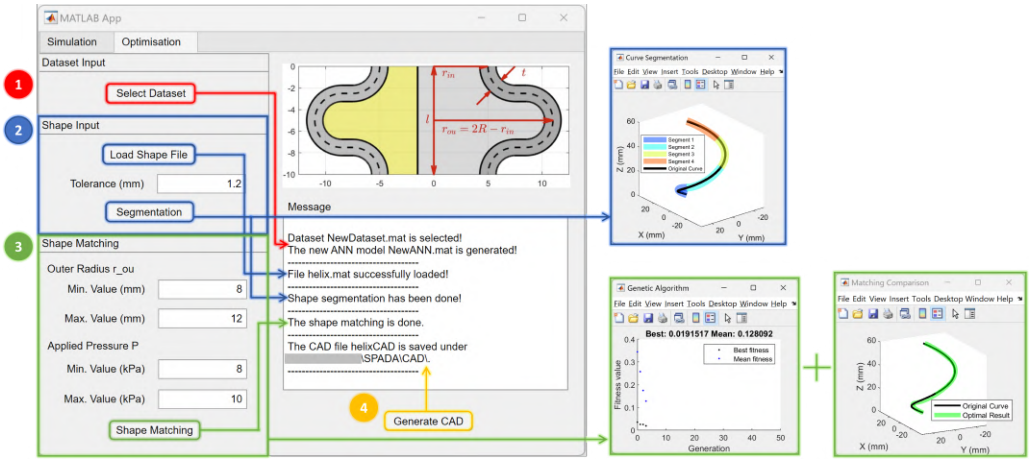


Figure S5: The GUI and usage instructions of the optimization component in SPADA (information for each step is highlighted in the same color as the step number icon)

The interface of the optimization component is shown in Fig. S5.

(1) In the ”Dataset Input” section, the ”Select Dataset” button trains a selected dataset into a surrogate model via an ANN as described in the paper’s Section 2.B.II. The dataset can be obtained by executing step 4 in the toolbox’s simulation component, with user-defined material properties. Alternatively, users can create their own datasets, ensuring matching the format of our provided dataset for Agilus30™ , available on our toolbox’s GitHub repository [42].

(2) In the ”Shape Input” section, the desired curve file, consisting of sequential 3D coordinates of points, can be input with the ”Load Shape File” button. The ”Segmentation” button helps divide the 3D shape into straight and circular arc segments by the 3D PCC segmentation algorithm according to the input tolerance value.

(3) The ”Shape Matching” section uses a genetic algorithm to find and output the optimal design parameters of the actuator that matches the shape upon pressurization based on the input upper and lower bounds of the actuator's outer radius and the applied pressure.

(4) If the algorithm can find a set of feasible design variables, the ”Generate CAD” button can generate a .stl format file of the designed actuator for direct 3D printing.

D Experimental Setup

I The 2D Shape-Matching

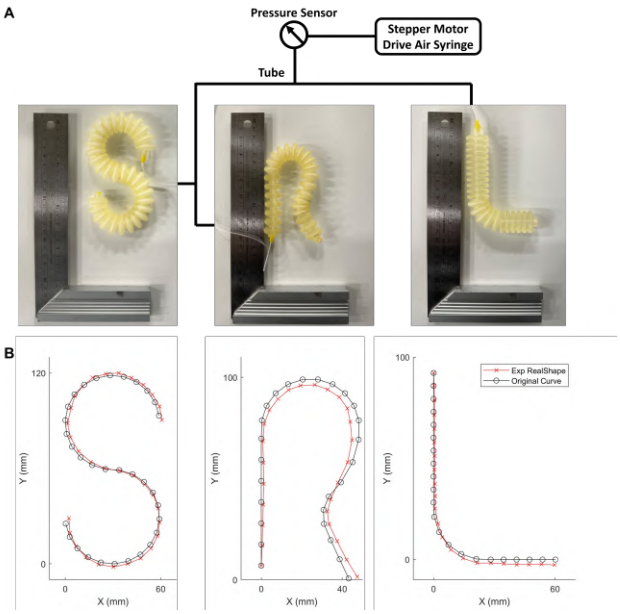


Figure S6: A) The setup for the 2D shape-matching experiment (top). B) The comparison between the original letter shapes and actuators' real shapes obtained through experiments (bottom).

The setup of the experiment in Sec 4.1 is shown in Fig.S6. Three actuators, "S", "R", "L" were driven with respective optimal pressures 8.11, 6.97 and 6.94kPa (which were approximated to 8, 7 and 7kPa in experiments due to the hardware limitations), determined along with their design parameters by the shape-matching design framework implemented on SPADA. A pneumatic system consisting of a 12V DC stepper motor driving a 60mL air syringe, a microcontroller (Arduino Uno), a pressure sensor (ADP51A11, pressure range 0-40 kPa, Panasonic, Japan) and a PI controller was used to control the input pressure. Lubricant was applied beneath the actuators to remove frictional effects. After taking photos from actuators' upward direction, the MATLAB Image Processing Toolbox™ was used to identify their real shapes by getting the central pixel coordinates of each module and using the steel square as reference. The original letter curves were sampled with the same intervals to compare with the real shapes. The root-mean-squared-errors for three actuators are 4.16, 2.70, and 2.51mm, respectively.

II The 3D Shape-Matching

The setup of the experiment in the paper's Section 4.B is shown in Fig. S7. The top end of the actuator is fixed to a connector (as shown in Fig.7A), which is connected to the top center of a 780×700×563 mm aluminum alloy frame. In order to get the deformed shape of the actuator, we 3D printed five rings, each featuring three evenly spaced rods connected to small reflective tape-coated spheres acting as markers for the OptiTrack™ system. Apart from the actuator's two ends, these rings are situated between every pair of adjacent segments. By identifying the position and orientation of these rings, and presuming a constant curvature between each pair, the real shape of the pressurized actuator can be obtained. The space coordinates of the markers were captured by a motion tracking system (OptiTrack™ with 4 Flex 3 cameras) and can be used to create a rigid body to represent the real-time positions of the free end.

A pneumatic system consisting of a 12V DC stepper motor driving a 60mL air syringe, a micro-controller (Arduino Uno), a pressure sensor (ADP51A11, pressure range 0-40 kPa, Panasonic, Japan) and a PI controller was also used to control the input pressure. We repeated the experiment 3 times for each trial. To analyze the data, we (1) calculated the average positions and orientations of all rings, which were treated as rigid bodies in the OptiTrack™ system, (2) determined arc planes and centers adjacent rings using their positions and tangent vectors, based on the constant curvature assumption, (3) aligned the position and orientation of the top ring with the uppermost point of the desired helical shape, (4) resampled the desired helical shape at the same interval as the sampled arcs' coordinates to derive RMSEs, comparing the desired and real shapes.

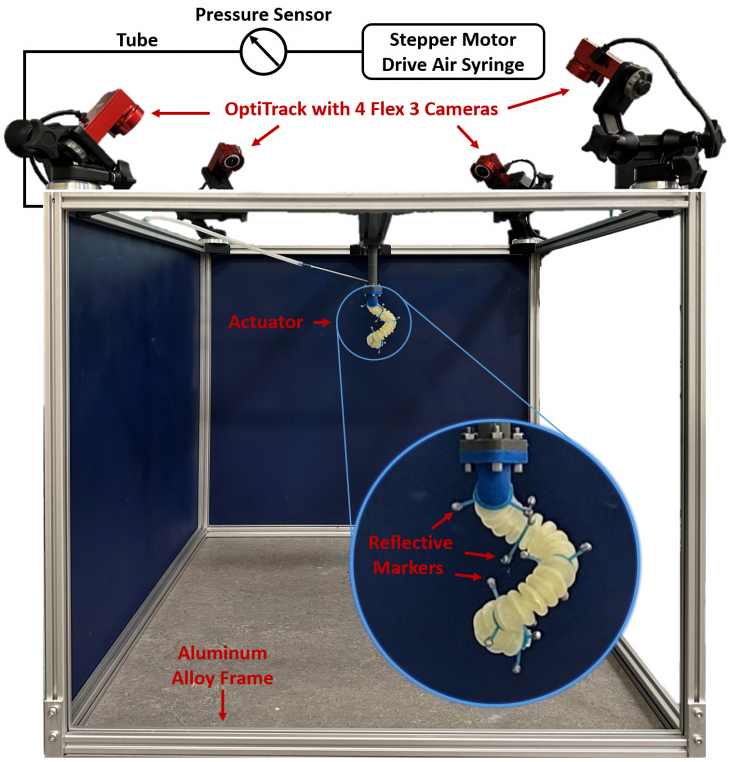


Figure S7: The experiment setup for the elephant-trunk-inspired, helical-shaped 3D deformable actuator.

To exploit grasping behaviors arising from the actuator design and interaction with the object, we performed grasping experiments on a series of objects. The object 1 in Fig.7E — a pen with a diameter of 16mm and length of 120mm, was the motivation for determining the helical shape's radius $a = \frac{16}{2} + r_{ou} = 8 + 10 = 18$mm, $h = 60$mm ensuring a power grasp. The experiment shows that the designed actuator can achieve a successful grasp for this object, which demonstrates the efficiency of our design pipeline. Subsequently, as shown in Fig. S8(A), we attached a 60mL syringe beneath the object and progressively increased its weight by filling it with water to explore the actuator's maximum grasping capacity. We discovered that when the syringe draws in water exceeding 30mL, equivalent to 73g (including the object), the grasp becomes unsuccessful. This is because grasping a heavier object requires higher pressure, which the designed actuator cannot handle for weights above 73g. Next, we evaluated the actuator's capability using another pen with smaller diameter shown in Fig.7E to assess its ability in grasping objects of reduced diameter. Following a successful grasp, we further tested its capability with even finer objects, confirming its ability to securely grasp items as slender as a zip tie shown in Fig. S8(B). Aside from power grasping, we also

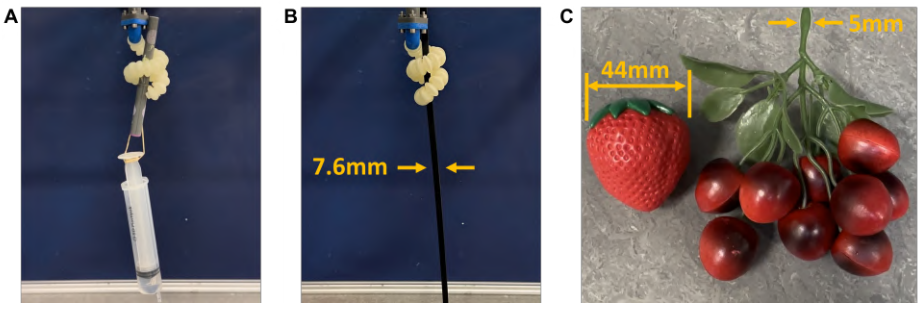


Figure S8: The supplemental objects for the grasping experiments: A) The object 1 in Fig. 7 carrying a syringe with water. B) A zip tie with a width of 7.6mm. C) Plastic strawberry with a diameter of 44mm and cherry with a handle of 5mm width.

explored the actuator's other grasping modes with object 3 and 4, as well as those shown in Fig. S8(C). Object 3 – a plastic banana represents a grasping mode where only a small part of the object, having a diameter within the preset range, is grasped. Despite the successful grasp of object 3, our attempt with a plastic cherry that has a slimmer handle, as seen in Fig. S8(C), was unsuccessful. This suggests that this grasping mode of the designed actuator has a diameter requirement for the object's contact region. Object 4 — a tape, represents another grasping mode where the object's geometry is not in the preset range as outlined in Fig.7A. However, the actuator, drawing inspiration from an elephant's trunk, coupled with the tape's mere 8mm thickness, could navigate through the tape's central hole and pinch it effectively. We attempted to use the pinch mode for grasping a plastic strawberry with diameter of 44mm but failed, indicating this grasping model of the designed actuator also has a requirement for objects' size.

In general, an actuator's grasping capability is influenced by its shape, material, and actuation pressure. Utilizing our toolbox necessitates a user-guided method to determine these aspects. By conducting grasping experiments and progressive refinements, users can achieve specific grasping performances. This iterative approach is also adaptable to various other applications.

5

An Optimisation Investigation for Bellow SPAs in Shape-Matching

Contents

5.1 Positioning with respect to the SOTA 73
5.1.1 Problem 73
5.1.2 Objective 74
5.1.3 Contributions 75
5.2 Assumptions, Validity Domain and Limitations 75
5.3 Contributed Paper 76
5.4 Supporting Information 84
5.4.1 Additional Experimental Results 84
5.4.2 Shape-Matching Optimisation: Low-Dimensional 85
5.4.3 Shape-Matching Optimisation: High-Dimensional 87

5.1 Positioning with respect to the SOTA

5.1.1 Problem

Optimisation has become a key component of inverse soft actuator design, enabling actuator geometries to be automatically generated by searching the design space with respect to prescribed design objectives. As summarised in Table 2.2, existing inverse design studies have adopted a variety of model–optimisation combinations,

coupling analytical models [38, 50, 86, 88], high-fidelity FEM simulations [56, 57], and surrogate models [89, 92] with optimisation strategies including gradient-based optimisation [50, 88], BO [91, 93], GA [58, 113], and PSO [90].

The suitability of these optimisation strategies depends strongly on the characteristics of the underlying modelling approach. Gradient-based optimisation is highly effective when accurate sensitivities are readily available, such as for analytical models [38, 50] or topology optimisation frameworks [86, 88]. By contrast, this chapter focuses on optimisation workflows based on high-fidelity FEM and surrogate models, as these modelling approaches generally provide improved predictive accuracy while being commonly employed as black-box evaluators in inverse design. Although sensitivities can in principle be obtained, exploiting gradient information would require substantial implementation efforts and expertise. Accordingly, this chapter focuses on gradient-free algorithms to maintain a generic optimisation framework for highly nonlinear and non-convex inverse design problems using accurate black-box computational models.

Among gradient-free optimisation methods, BO and GA are two of the most widely adopted approaches in soft actuator inverse design, representing model-based global optimisation and population-based heuristic search, respectively. However, as summarised in Table 2.2, previous studies have typically investigated only a single model–optimisation combination, with computationally demanding combination coupling high-fidelity FEM directly with population-based optimisation algorithms such as GA [58], reflecting the limited understanding of how optimisation performance depends jointly on the underlying modelling approach and optimisation strategy.

5.1.2 Objective

The objective of this work is to systematically investigate how optimisation performance depends on different combinations of modelling approaches and optimisation strategies for shape-matching inverse design of bellow SPAs. Specifically, high-fidelity FEM and surrogate models are coupled with BO and GA under varying design dimensionalities.

5.1.3 Contributions

The contributions of this work are as follows:

- A systematic investigation of representative model–optimisation combinations for shape-matching inverse design of bellow SPAs, examining BO and GA in combination with high-fidelity FEM and NN-based surrogate models.
- A quantitative comparison of representative model–optimisation combinations in terms of computational cost and achieved design quality across different design-space dimensionalities.
- Practical guidance for selecting appropriate model–optimisation combinations for optimisation-based shape-matching design of bellow SPAs under different computational and design-space constraints.

5.2 Assumptions, Validity Domain and Limitations

The optimisation study adopts the same shape-matching objective defined in Chapter 4. Throughout the comparison, both the high-fidelity FEM model and the NN-based surrogate model are assumed to provide sufficiently accurate evaluation of candidate designs within their respective validity domains, such that the comparison isolates the influence of the optimisation workflow rather than modelling accuracy.

The conclusions of this study should not be interpreted as universally applicable to all optimisation-based soft actuator design problems. They are derived from a specific shape-matching inverse design problem involving parameterised bellow SPAs, representative modelling approaches and gradient-free optimisation algorithms. Consequently, although the identified model–optimisation combinations provide an effective optimisation workflow for the SPADA framework developed in Chapter 4, the resulting guidance for other optimisation problems should be regarded as qualitative rather than quantitative. Different optimisation objectives, modelling

approaches, design constraints, or actuator architectures may require a separate systematic characterisation to establish problem-specific quantitative conclusions.

5.3 Contributed Paper

This work was published as a conference paper in the 2023 IEEE 6th International Conference on Soft Robotics (RoboSoft) and presented orally in Singapore. The paper is attached on the following pages.

Design Optimization for Bellow Soft Pneumatic Actuators in Shape-Matching*

Yao Yao[1], Yuwen Chen[2], Liang He[1] and Perla Maiolino[1]

***Abstract*— Design optimization of soft actuators is essential for task-oriented applications. Models derived from analytical solutions, the Finite Element Method (FEM), or empirical characterized datasets are widely used to estimate the response of the actuators during actuation, acting as the backbone for design optimization. Faced with the trade-off between speed and accuracy, substantial challenges occur when moving from simulation to optimization due to the compliant, high degree of freedom, and high-dimensional design space of the soft-bodied robot. FEM becomes increasingly computationally expensive with increased design complexity during optimization iterations, while the data-driven modeling approach (e.g., Artificial Neural Network) consumes significant resources prior to optimization. To address the challenge of highly nonlinear and non-convex design optimization in soft robots using the black box modeling, this paper compares of Bayesian optimization (BO) algorithm and genetic algorithm (GA) with FEM and Artificial Neural Network (ANN) models. The shape-matching of a multi-legged robot (a starfish) is demonstrated as an example of a task-oriented design scenario that presents design optimization challenges of the design space scalability. The experimental results show that the bi-level BO outperforms BO with FEM by achieving 2.8 to 9.8 times smaller objective values within a certain time for low-dimensional design problems; GA with the ANN model can achieve lower objective values 3 to 18 times faster in high-dimensional design problems than bi-level BO with FEM in low-dimensional design problems.**

I. INTRODUCTION

The field of soft robotics is rapidly developing rapidly, extending the paradigm of traditional rigid robotics because (1) lightweight and low-cost soft materials provide safe interaction, compliance, and mobility during tasks; (2) design flexibility leads to different deformations, which allows morphological adaptability; (3) the commonly used pressurized air driving method reduces the complexity of control [1]. Pinskier et al. decomposed the design architecture of soft robots into subspaces [2], mapping high-level tasks to robot morphology to detailed design optimization problems. For example, fish-like underwater robots are optimized to maximize swimming frequency and velocity [3], while finger-based gripping robots are optimized in their workspace and force for safe grasping [4]. Optimizing the geometry of a complex soft body to perform a specific action (task) through actuation is crucial for the design of soft robots.

*This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) Grant EP/V000748/1;

[1]Yao Yao, Liang He and Perla Maiolino are with Oxford Robotics Institute, University of Oxford, Oxford, OX1 2JD, United Kingdom yao.yao/liang.he/perla.maiolino@eng.ox.ac.uk

[2]Yuwen Chen is with the Department of Engineering Science, University of Oxford, Oxford, OX1 3PJ, United Kingdom yuwen.chen@eng.ox.ac.uk

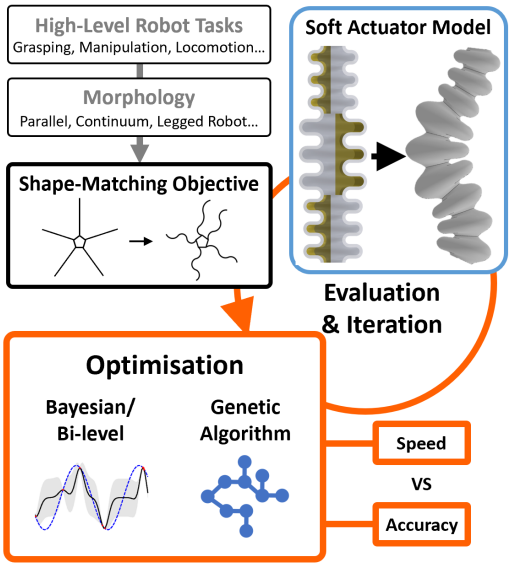


Fig. 1. Schematic diagram of the soft robot design process, from high-level robot tasks to specific morphology to detailed design objectives such as shape matching. The soft robot design is iteratively optimized using optimization algorithms (Bayesian optimization and Genetic Algorithm are considered in this paper), with models that predict the configuration of soft robots. The ultimate challenge of the design optimization of the soft bodied robot is the trade-off between speed and accuracy.

Design optimization builds on a reliable model to predict the configuration and response of the soft robot, bridging the gap between the design space and the objectives of the task. Analytical models are commonly used for fast evaluation in the optimization process [5]. However, theoretical models for continuum robots lack generality and are not suitable for design problems of complex and diverse applications [2]. In contrast, the Finite Element Method (FEM) is generally adaptability and easy to implement [6]. It discretizes the structure into small elements and solves the physics problem in an element-wise manner, accurately predicting the performance of soft robots with different features [7][8][9]. Nevertheless, FEM can be computationally expensive and have convergence issues due to the large deformation and nonlinear material properties of soft robots, posing efficiency challenges in design optimizations[10].

Data-driven methods using neural networks to learn and generate models from reliable datasets have been proposed for efficient design optimization of soft robots [2]. Supervised Artificial Neural Networks (ANNs) are capable of accurately predicting the performance of parameterized soft actuators by learning labeled training data [11], but this approach requires large-scale data collection in simu-

lations/experiments to generate an accurate model, moving most of the resource cost ahead of the optimization.

Both FEM and ANN model lack analytical sensitivity, treated as black boxes in the optimization process. In addition, the large deformation, material nonlinearity, and complex boundary conditions of soft robots present a highly nonlinear and non-convex design space, making gradient-based optimization algorithms prone to falling into local minima and highly dependent on the selection of initial points. In this situation, global optimization approaches, such as model-based and heuristic search algorithms, have been adopted as alternative approaches [12]. Model-based global optimization algorithms create models to learn the objective function's behavior and determine the next evaluation point, keeping the iteration number small [13]. Heuristic search algorithms use strategies inspired by biological behaviors to evaluate and update a group of possible candidates for better search in the space [14].

Despite the wide use of these global optimization algorithms in soft robot design problems, selecting an optimization algorithm is still challenging due to the diversity of available algorithms and the lack of guidance on their suitability for different design spaces and simulation methods [5]. Moreover, the trade-off between computational resources and performance in optimization algorithms selection affects the usability of the methods with adaptable tasks that update with environmental changes.

Therefore, this paper investigates optimization algorithms targeting FEM and ANN models for the shape-matching design optimization of bellow soft pneumatic actuators (bellow-SPAs), see Fig. 1. We specialize in bellow soft pneumatic actuators (SPAs) because their modular design enables various deformation behaviors, which have been used for wide applications in soft robotics [15]. Shape-matching is chosen as a case study for the optimization objective because many tasks require actuators to achieve specific shapes/geometries through actuation [16], [17]. The widely implemented scenario can be simplified to match the desired shapes at critical points along the trajectory. After formulating the design optimization problem, Bayesian optimization (BO) and genetic algorithm (GA) are selected as representative algorithms among model-based and heuristic search algorithms, combined with FEM and ANN models, respectively. Their performance is evaluated through the shape-matching of a starfish design (a simple multi-legged soft robot design) with up to five legs, each consisting of multiple segments of constant curvature. The dimension/number of design variables is varied to compare the converge performance of the objective in terms of iteration numbers of different optimization algorithms, providing a selection strategy for shape-matching design optimization problems.

The rest of this article is organized in Section (II) the design of bellow SPAs and models to predict their performance, (III) the formulated design optimization problem and algorithms, (IV) the results of comparing algorithms for designing starfish-inspired soft robots, (V) the discussion and conclusion.

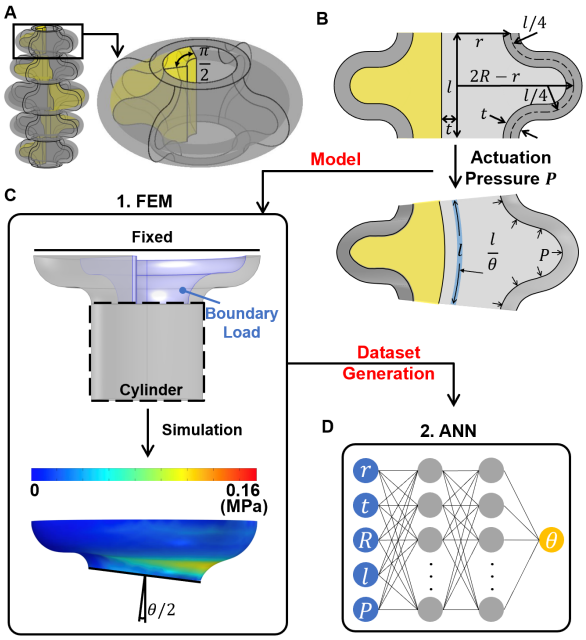


Fig. 2. The design and models of bellow SPAs. (**A**) A bellow SPA consists of different modules. (**B**) The parameterized module and its kinematics model after actuation. (**C**) The FEM model and stress distribution of half module. (**D**) The ANN model trained from FEM dataset.

II. Design and Models

The design of the target bellow SPAs is shown in Fig. 2. The actuator consists of parameterized individual modules (Fig. 2A and B). By tuning the inner radius r, wall thickness t, average radius R and module length l, and the orientation of the filling (a quarter fan-shaped filling shown in yellow), the actuator can achieve different bending radius and direction. Angular deflection θ is defined in Fig. 2B following the constant curvature assumption with an associate arc represented by arc length l and arc curvature $\frac{\theta}{l}$ to describe the actuator motion with driving pressure P.

To simulate the actuator's deformation using FEM, a stationary study in 3D solid mechanics is performed in COMSOL MultiPhysics® on the half bellow module thanks to the geometrical symmetry. As shown in Fig. 2C, the wider end is set as a fixed boundary, while the other end is connected to a solid cylinder to smooth element deformation. Air pressure actuation is modeled as a uniformly distributed load on the internal surface of the half module. Besides, an incompressible Neo-Hookean model is applied to the material [6]. After simulation, the angular deflection of half module is calculated based on the displacement of points at the end of the cylinder.

The FEM model establishes a relationship between the design parameters of the bellow SPA module and its angular deflection with respect to the actuation pressure, which can be written as $\theta = f_1(r, t, P, R, l)$. It can also be used to generate a dataset for data-driven modeling method. A supervised ANN is used to learn the relationship between input (design parameters and actuation pressure) and output

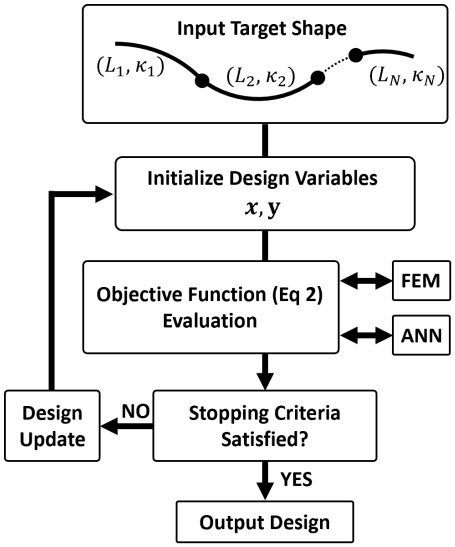


Fig. 3. The design optimization process of bellow SPAs for shape-matching objectives.

angular deflection data to generate an ANN model that can be written as $\theta = f_2(r, t, P, R, l)$. The accuracy of this ANN model is largely dependent on the dataset scale and collection method, resulting in different levels of resource costs.

Both models can be used to predict the response of soft robots in design optimization, where the FEM consumes computational resources during optimization and the ANN model moves most of it to the front.

III. Design Optimization

Given the target shape of the actuator, our goal is to find the optimal design parameters for the bellows SPA such that the deformed actuator approximates the shape when pressurized and satisfies all hardware constraints. The target shape is usually approximated by arcs and line segments using a piecewise constant curvature segmentation algorithm that follows the G^1 continuity [17]. In this work, we prescribe the number and parameters of each arc segment and assume that the actuator is in a linear shape before actuation.

As the bellow SPA is parameterized on its modular design, each arc segment can be designed by stacking modules with the same geometric parameters and fan-shape filling position. For stacking convenience, all modules share the same inner radius r and wall thickness t but may have different average radius R and module length l. Thus, the deformed actuator shape is defined by the shared and respective geometric parameters of each segment, in addition to its actuation pressure.

A. optimization Problem Formulation

The process of the shape-matching design optimization of bellow SPAs is shown in Fig. 3. Assuming the actuator has N arc segments with arc length and curvature pairs $[(L_1, \kappa_1), (L_2, \kappa_2), \ldots, (L_N, \kappa_N)]$ different from each other, we wish to determine the shared parameters $\mathbf{x} = [r, t, P] \in \mathbb{R}^3_+$ and a set of respective geometric parameters $\mathbf{y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N] = [(R_1, l_1), (R_2, l_2), \ldots, (R_N, l_N)] \in \mathbb{R}^{2N}_+$ of the actuator that can approximate the configuration of each segment as closely as possible.

Therefore, the objective is to minimize the difference between designed segment parameters and prescribed target ones. The hardware constraints are specified by (1) the maximum actuation pressure $P_{\max}$, which depends on the degree of SPA deformation, the selection of SPA material and air supply capability, (2) the maximum module radius, limited by the minimum arc radius of segments $\frac{1}{\kappa_{\max}}$, where $\kappa_{\max}$ is the maximum curvature of segments, (3) the minimum wall thickness $t_{\min}$, which should be set to prevent leakage; (4) the maximum module length, restricted by the segment length, and (5) inherent design constraints of bellow SPA module, namely

$$\begin{cases} \frac{r}{4} \le t \le \frac{r}{2} \\ r + t \le R_i \le 2r, \quad \forall i \in [N] \\ 4t \le l_i \le 4(R_i - r) \end{cases}, \tag{1}$$

where $[N]$ is the indexing set of N segments. Therefore, the shape-matching design optimization of bellow SPAs can be formulated as

$$\begin{aligned} &\min\ h(\mathbf{x}, \mathbf{y}) := \frac{1}{N} \sum_{j=1}^{N} d_i(\mathbf{x}, \mathbf{y}_i) \\ &\text{s.t.} \begin{cases} 0 \le P \le P_{\max} \\ 0 < r \le \frac{1}{\kappa_{\max}} \\ \max(t_{\min}, \frac{r}{4}) \le t \le \frac{r}{2} \\ r + t \le R_i \le \min(\frac{1}{2\kappa_i} + \frac{r}{2}, 2r) \\ 4t \le l_i \le \min(L_i, 4(R_i - r)), \quad \forall i \in [N] \end{cases} \end{aligned}, \tag{2}$$

where

$$d_i(\mathbf{x}, \mathbf{y}_i) = w_1 |L_i - m_i l_i| + w_2 \left| \kappa_i - \frac{\theta_i}{l_i} \right|. \tag{3}$$

$m_i = \left[\frac{L_i}{l_i}\right]$ is the number of bellow SPA modules in the i-th segment, and $w_1 = 1/L_i$, $w_2 = 1/\kappa_i$ are the weighting factors used to balance the difference of units. The value of θ_i relies on $\mathbf{x}, \mathbf{y}_i$ and is obtained from either FEM $f_1(\mathbf{x}, \mathbf{y}_i)$ or the ANN model $f_2(\mathbf{x}, \mathbf{y}_i)$.

B. optimization Algorithms

The objective function is highly nonlinear and non-convex due to nonlinear materials, large deformations and distributed actuation loads. Besides, the FEM from commercial software and the ANN model trained from the dataset are both treated as black boxes to the objective function. To this end, we propose two method to deal with these challenges. One is to run FEM within Bayesian optimization, doing an online sequential inference. The other is to first collect a dataset from FEM and generate an ANN model (as explained in Sec.II), then use GA to perform optimization using the ANN model to predict the actuator configuration.

Algorithm 1 BO with FEM: BO($h(\mathbf{x}, \mathbf{y}), K, \epsilon$)

Initialization:
Dataset $\mathbb{D} = \emptyset$. no prior of the response surface $\hat{h}(\cdot) : (\mathbf{x}, \mathbf{y}) \rightarrow h(\mathbf{x}, \mathbf{y})$. Accuracy ϵ and maximum number of iteration K.
while $h(\mathbf{x}^k, \mathbf{y}^k) > \epsilon$ **and** $k < K$ **do**
  $k \leftarrow k + 1$
  Update a response surface $\hat{h}^k(\mathbf{x}, \mathbf{y})$ from $\mathbb{D}$;
  $(\mathbf{x}^k, \mathbf{y}^k) \leftarrow \text{argmax}_{(\mathbf{x},\mathbf{y})} \alpha^k(\mathbf{x}, \mathbf{y})$;
  Obtain $\theta^k = f_1(\mathbf{x}^k, \mathbf{y}^k)$ from FEM and then $h(\mathbf{x}^k, \mathbf{y}^k)$;
  Add $\{(\mathbf{x}^k, \mathbf{y}^k), h(\mathbf{x}^k, \mathbf{y}^k)\}$ to $\mathbb{D}$, i.e. $\mathbb{D} \leftarrow \mathbb{D} \cup \{(\mathbf{x}^k, \mathbf{y}^k), h(\mathbf{x}^k, \mathbf{y}^k)\}$.
end while
Output: $(\mathbf{x}^*, \mathbf{y}^*) = \text{argmin}_{j \in \{1,\ldots,k\}} h(\mathbf{x}^j, \mathbf{y}^j)$

Algorithm 2 Bi-level BO with FEM

Initialization:
Dataset $\mathbb{D} = \emptyset$ and no prior of the response surface $\hat{h}(\cdot) : \mathbf{x} \rightarrow \min_{\mathbf{y} \in \mathcal{Y}_i} h(\mathbf{x}, \mathbf{y})$. Accuracy ϵ and maximum number of iteration for outer loops K_{ou} and inner loops K_{in}.
while $h(\mathbf{x}^k, \mathbf{y}^k) > \epsilon$ **and** $k < K_{ou}$ **do**
  $k \leftarrow k + 1$
  Update a response surface $\hat{h}^k(\mathbf{x})$ from $\mathbb{D}$;
  $\mathbf{x}^k \leftarrow \text{argmax}_{\mathbf{x}} \alpha^k(\mathbf{x})$;
  for $i = 1, \ldots, N$ **do**
    $\mathbf{y}_i^k \leftarrow \text{argmin}_{\mathbf{y}_i} d_i(\mathbf{x}^k, \mathbf{y}_i)$ via BO($d_i(\mathbf{x}^k, \cdot), K_{in}, \epsilon$);
  end for
  Sum up $d_i(\mathbf{x}^k, \mathbf{y}_i^k)$ to obtain $h(\mathbf{x}^k, \mathbf{y}^k)$;
  Update dataset $\mathbb{D}$, i.e. $\mathbb{D} \leftarrow \mathbb{D} \cup \{\mathbf{x}^k, \mathbf{y}^k, h(\mathbf{x}^k, \mathbf{y}^k)\}$.
end while
Output: $(\mathbf{x}^*, \mathbf{y}^*) = \text{argmin}_{j \in \{1,\ldots,k\}} h(\mathbf{x}^j, \mathbf{y}^j)$

1) BO with FEM: BO is an efficient method for solving black-box and time-consuming optimization problems [18]. It iteratively samples data of variables $\mathbf{z}$ and the corresponding objective function evaluations $g(\mathbf{z})$. The dataset $\mathbb{D} = \{\mathbf{z}, g(\mathbf{z})\}$ is used to create a probabilistic model, also known as the response surface $\hat{g}(\cdot) : \mathbf{z} \rightarrow g(\mathbf{z})$, that learns and mimics the behavior of the objective function. The learned model is used to determine the next set of variables to evaluate and add the new data to update the model.

The most commonly used model is the Gaussian process (GP), which places a probability distribution over the objective function with mean and variance describing the model uncertainty. The acquisition function $\alpha(\mathbf{z})$ uses information from GP to determine the next set of variables by maximizing itself

$$\mathbf{z}^* \in \text{argmax}_{\mathbf{z}} \ \alpha(\mathbf{z}) \tag{4}$$

and updating the posterior distribution over the objective functions based on the newly added data. This method of choosing the following evaluation points based on previous results keeps the number of evaluations small, making it suitable for evaluations using FEM. Algorithm 1 shows how to combine Bayesian optimization with FEM for our design optimization problem where $\mathbf{z} := (\mathbf{x}, \mathbf{y})$ and $g(\mathbf{z}) := h(\mathbf{x}, \mathbf{y})$. The iteration process continues until finding a set of variables where the objective value is below ϵ or the number of iterations reaches its maximum value K.

Algorithm 1 has been proven to work well for many optimization problems [19], [20]. However, with the increasing number of segments (design variables), it gets harder to build GP and search for the next points with the acquisition function. It requires the scale of $\mathbb{D}$ (equivalent to the number of iterations) to grow exponentially to keep the same quality of model approximation [21], making it computationally infeasible.

To alleviate computational burden, a bi-level variant is introduced, Algorithm 2, to improve the performance of Bayesian optimization [22] considering our optimization problem is decomposed w.r.t. variable $\mathbf{y}$. The optimization problem is constructed into the upper and lower level, optimizing over $\mathbf{x}$ and $\mathbf{y}$, respectively. During each iteration of the upper level, $\mathbf{x}$ is obtained from the upper-level acquisition function. Then, it is treated as constant in the lower level optimization, which is decomposed into N independent BO over $\mathbf{z} := \mathbf{y}_i$ and $g(\mathbf{z}) := d_i(\mathbf{x}, \mathbf{y}_i)$, where d_i is defined in formula (3). Then, the optimal value of d_i is summed up to update the objective value of the upper level. This process continues until the upper-level cost h is below the desired value ϵ or the number of iterations reaches K_{ou}.

Although bi-level Bayesian optimization can solve Problem (2) with a desired low cost, the use of FEM prohibits the bi-level BO for a design problem with many segments. To tackle the curse of dimensionality, a method combining the ANN model with GA is introduced in the next part for solving high-dimensional design problems without calling FEM too many times.

2) GA with ANN model: GA is a well known evolutionary algorithm based on the concept of biological evolution [23]. It evaluates a large number of design points per iteration, which is suitable to be combined with the ANN model since the ANN model conducts evaluation efficiently.

As explained in Algorithm 3, the ANN model based on the dataset of M feasible points with their θ values obtained from FEM is first generated. This step takes most of the time and it depends on the computational hardware, the complexity of FEM simulation and the dataset scale.

Once the ANN model is obtained, the algorithm proceeds as standard GA. A feasible generation $\mathcal{S}$ is initialized with a fixed size of population. For each individual n inside population $\mathcal{S}$, the corresponding cost $h(\mathbf{x}^n, \mathbf{y}^n)$ is obtained based on the ANN model. If the cost of the best performed point $(\mathbf{x}^*, \mathbf{y}^*)$ of the current population members is within accuracy ϵ or the weighted change Δh^k over the last several iterations is less than τ, the algorithm terminates and returns $(\mathbf{x}^*, \mathbf{y}^*)$ as the optimal design parameter. Otherwise, the next generation is created by preserving elites and selecting parents to produce their crossover and mutation children.

Algorithm 3 GA with ANN model

Initialization:
Generate M feasible points to train a model using ANN: $\theta = f_2(r, t, P, R, l)$;
Set accuracy ϵ, function tolerance τ and the maximum generation number K.
while $k < K$ or $h(\mathbf{x}^*, \mathbf{y}^*) < \epsilon$ or $\Delta h^k < \tau$ **do**
  $k \leftarrow k + 1$
  Create a new generation $\mathcal{S}$;
  for $n \in \mathcal{S}$ **do**
    for $i = 1, \ldots, N$ **do**
      Obtain $\theta_i^n = f_2(\mathbf{x}^n, \mathbf{y}_i^n)$ and then $d_i(\mathbf{x}^n, \mathbf{y}_i^n)$;
    end for
    Sum up $d_i(\mathbf{x}^n, \mathbf{y}_i^n)$ to obtain $h(\mathbf{x}^n, \mathbf{y}^n)$;
  end for
  $(\mathbf{x}^*, \mathbf{y}^*) = \text{argmin}_{(\mathbf{x}^n, \mathbf{y}^n) \in \mathcal{S}} h(\mathbf{x}, \mathbf{y})$;
  Compute a weighted change Δh^k
end while
Output: Best solution (x^*, y^*).

IV. Experimental Results

The above optimization algorithms are analyzed through the design of starfish-inspired soft robots, which presents challenges in terms of scalability related to the number of segments and parameter value ranges, and these challenges can also be found in other applications such as soft grippers and multi-segment soft arms [2][5]. The starfish design space is defined by setting the leg length between 100-150 mm and the range of motion within 2π. Assuming that each leg consists of constant curvature arc segments, a script is used to randomly generate the length and curvature of each segment within the specified range.

To evaluate the performance of optimization algorithms for the shape-matching design optimization of bellow-SPAs, we vary the dimension of the design variables and examine the convergence performance in terms of the trade-off between speed and accuracy. Specifically, we compare the minimum objective value and running time for each algorithm as a function of the number of iterations.

The design problems with different dimensions of parameters are defined as follow: for low-dimensional designs, a single leg is considered, consisting of $N = 1$ or 2 or 3 arc segments; while for high-dimensional designs, there are five legs all different from each other, each consisting of 1 or 2 or 3 arc segments, resulting in a total of $N = 5$ or 10 or 15 arc segments. As FEM simulation needs to be run for each evaluation of the deformation for one arc segment, we decide to perform (bi-level) BO with FEM on low-dimensional design optimizations, and GA with ANN model on high-dimensional ones (we use BO-FEM, BiBO-FEM and GA-ANN to represent each algorithm for simplicity in the following content). We choose Agilus30 as the actuator material in design tasks [24]. Based on our previous characterization [8], the boundary of the actuation pressure and wall thickness are set to be $P_{\max} = 10$ kPa and $t_{\min} = 1$ mm.

The algorithms are implemented in MATALB and executed on a laptop with an Intel(R) Core(TM) i7-10875H CPU @ 2.30GHz and 16.0 GB RAM. To generate an ANN model based on a FEM dataset, it takes approximately 6 hours (the table of parameter ranges and intervals for FEM dataset collection is presented in supplementary video). For the stopping criteria, we specify the maximum number of iterations for (bi-level) BO and the function tolerance for GA. The K of BO-FEM are set to be 300, 500, and 1000 for the corresponding low dimensional design problems. For all low dimensional design problems of BiBO-FEM, the K_{ou} and K_{in} are set to be 30 and 15 for the upper and lower levels, respectively. As for GA-ANN, the function tolerance τ is set to be 1×10^{-6}, which halts the algorithm if the average change of the objective value in the last 50 generations is less than it.

The evaluation experiment is performed 3 times for each design problem, one of which is shown in Fig. 4(A) to (F), where the minimum objective values and corresponding computational times are plotted against the number of iterations (the remaining results are included in the supplementary video). We also calculate the mean and standard deviation of the final minimum objective value, its respective iteration number, and running times for each design problem, as shown in Table I. We focus on the minimum target value range between 0 and 0.1, as it is where the design parameters lead to viable solutions for shape matching. The shape of the design parameters based on objective values at different levels in this range is compared with the target shape in Fig. 4(G).

The results of low-dimensional design problems using BO-FEM and BiBO-FEM are shown in Fig. 4(A) to (D) and Table I. Firstly, all experiments reach their final minimum objective values (or a value very close to it with a difference smaller than 0.05) before half of the maximum number of iterations, validating BO's property of keeping the iteration number small. Second, the results using BO-FEM show that only the final minimum objective value of 0.0307 for $N = 1$ is below 0.1, which is 2.8 times the corresponding result using BiBO-FEM, 0.0117. While the final minimum objective values for $N = 2$ and 3 are 0.1382 and 0.2453, respectively, which are 5 and 9.8 times the corresponding results of 0.0276 and 0.0251 using BiBO-FEM. This demonstrates that BO-FEM cannot handle the scalability of the number of segments, and is only suitable for the design problem of $N = 1$. Third, in the design problems using BiBO-FEM, the running time required to achieve the final minimum objective value for $N = 2$ is longer than $N = 3$, which is 25252 s and 23031 s, respectively. This is because there is one experiment for $N = 2$ achieving its final minimum 0.0414 at iteration 20 after reaching 0.0473 at iteration 7. Apart from this exception, in general, the running time of BiBO-FEM increases with the increase of the number of segments. However, due to the high dependence the FEM model complexity, the running time of BiBO-FEM in relation to the number of iteration is very unpredictable.

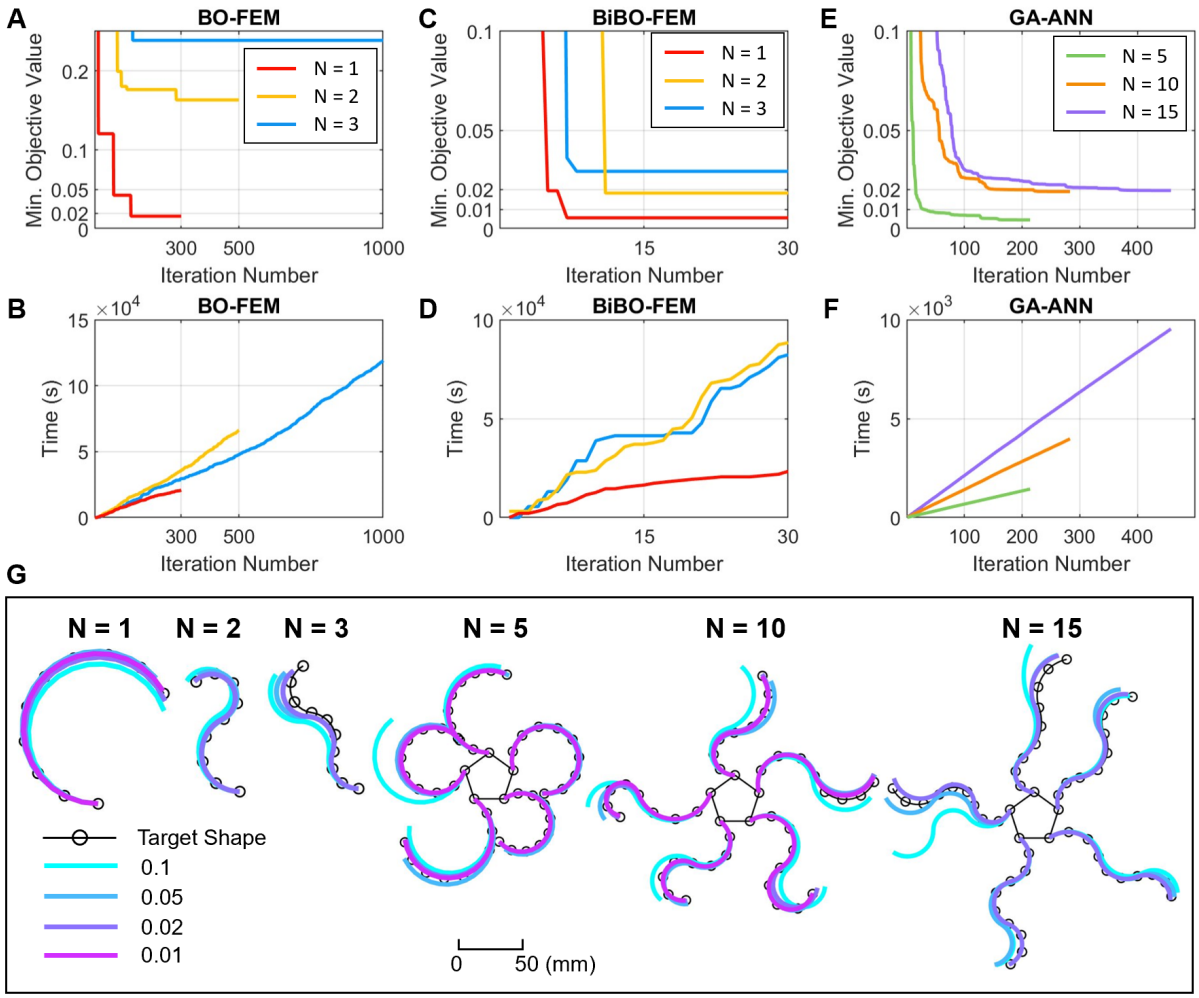


Fig. 4. The evaluation results of optimization algorithms. The iteration number vs min. the objective value of (**A**) BO-FEM, (**C**) BiBO-FEM, and (**E**) GA-ANN for their different design problems. (**B**) (**D**) (**F**) The iteration number vs running time of the corresponding algorithms. (**G**) The comparison of the designed shapes of objective values at different levels with the target shapes, where the design parameters of the low-and high-dimensional problems are from BiBO-FEM and GA-ANN, respectively.

Additionally, for the design problem of $N = 3$ using BiBO-FEM, the average running time of 23031 s required to reach the final minimum objective value is very close to our time for collecting the FEM dataset to generate the ANN model, indicating that GA-ANN could be a more efficient approach for design problems with more than 3 segments.

The results of high-dimensional design problems using GA-ANN are presented in Fig. 4(E)(F) and Table I. Firstly, the average final minimum objective value increases as the number of segments increases. Second, when comparing the performance of high-dimensional design problems using GA-ANN to low-dimensional ones using BiBO-FEM, the mean and standard deviations of the final minimum objective values of the former are generally smaller than those of the latter, indicating that GA-ANN is more robust in terms of objective value convergence. Moreover, the average running time of 1265.2 s for $N = 5$ is 18.2 times smaller than 23031s for $N = 3$ (which is equivalent to the time required to generate our ANN model). Although the running time increases with the increase of the number of segments, the longest time of 7652.8 s for $N = 15$ is still almost 3

TABLE I

EVALUATION RESULTS OF OPTIMIZATION ALGORITHMS FOR DIFFERENT DESIGN PROBLEMS

Algorithm	Number of Segment(s)	Final Min. Objective Value	Respective Iteration Number	Respective Running Time (s)
BO & FEM	1	0.0307 ± 0.0216	132.7 ± 12.4	12414 ± 2105
	2	0.1382 ± 0.0266	304.3 ± 112.5	38653 ± 21439
	3	0.2453 ± 0.0561	566.3 ± 392.9	118590 ± 104372
Bi-BO & FEM	1	0.0117 ± 0.0128	7.3 ± 4.5	9899 ± 10329
	2	0.0276 ± 0.0122	12 ± 7.5	25252 ± 9799
	3	0.0251 ± 0.0036	9 ± 1	23031 ± 5110.7
GA & ANN	5	0.0036 ± 0.0027	188.7 ± 36	1265.2 ± 233.6
	10	0.0171 ± 0.0049	233.3 ± 11.9	3169.4 ± 197.9
	15	0.0208 ± 0.0036	369.3 ± 34.7	7652.8 ± 824

times smaller than the time for generating the ANN model. Additionally, the running time complexity in relation to the increase of the number of iterations and the number of segments is more proportional for GA-ANN than BiBO-FEM, making the former more time-predictable.

Besides, Fig. 4(G) shows the comparison between the designed shape and the target shape for each design problem. The black line with circular markers represents the target shape, and other colored lines represent the designed shapes based on objective values around 0.1, 0.05, 0.02 and 0.01, respectively. As the objective value decreases and the line color transitions from light blue to magenta, the designed shapes gradually fits the desired shape, demonstrating the effectiveness of our optimization method in Sec III-A (clearer representation and detailed design parameters are included in the supplementary video).

V. Discussion and Conclusion

Facing with the challenge from the trade-off between speed and accuracy in design optimization of soft robots using black-box modeling methods, we conducted experiments comparing the convergence performance of (bi-level) BO and GA targeting FEM and ANN models for shape-matching design optimization of bellow-SPAs in a starfish-inspired soft robotic scenario.

Results demonstrate that for low-dimensional design problems, BiBO-FEM outperforms BO-FEM by achieving 2.8 to 9.8 times lower objective values within a given running time, indicating that the bi-level/decomposition method is an effective strategy for reducing the modeling complexity of BO. For high-dimensional design problems, GA-ANN not only achieves low objective values efficiently but also demonstrates robustness and time prediction in the convergence of objective values. Overall, considering computational resources, BiBO-FEM and GA-ANN are recommended for low- and high-dimensional design problems, respectively. However, the boundary between low and high dimensions depends on the collection method and dataset size.

One limitation of this study is that we only varied the dimension of design variables, and the stopping criteria were determined empirically. In future work, we plan to conduct a more comprehensive evaluation of optimization algorithms. This evaluation may involve varying optimization constraints, exploring the impact of different maximum iteration numbers for the lower level in BiBO-FEM, and optimizing the parameters (such as population size, crossover fraction, etc.) of GA. Furthermore, it would be valuable to extend the optimization framework to other types of actuators and consider more complex scenarios.

References

[1] D. Rus and M. T. Tolley, "Design, fabrication and control of soft robots," *Nature*, vol. 521, p. 467–475, 2015.

[2] J. Pinskier and D. Howard, "From bioinspiration to computer generation: Developments in autonomous soft robot design," *Advanced Intelligent Systems*, vol. 4, no. 1, p. 2100086, 2022.

[3] E. Nava, J. Z. Zhang, M. Y. Michelis, T. Du, P. Ma, B. F. Grewe, W. Matusik, and R. K. Katzschmann, "Fast aquatic swimmer optimization with differentiable projective dynamics and neural network hydrodynamic models," in *39th International Conference on Machine Learning*, pp. 16413–16427, 2022.

[4] H. Zhang, A. S. Kumar, J. Y. H. Fuh, and M. Y. Wang, "Design and development of a topology-optimized three-dimensional printed soft gripper," *Soft Robotics*, vol. 5, no. 5, pp. 650–661, 2018. PMID: 29985781.

[5] F. Chen and M. Y. Wang, "Design optimization of soft robots: A review of the state of the art," *IEEE Robotics & Automation Magazine*, vol. 27, no. 4, pp. 27–43, 2020.

[6] M. S. Xavier, A. J. Fleming, and Y. K. Yong, "Finite element modeling of soft fluidic actuators: Overview and recent developments," *Advanced Intelligent Systems*, vol. 3, no. 2, p. 2000187, 2021.

[7] P. Polygerinos, Z. Wang, J. T. B. Overvelde, K. C. Galloway, R. J. Wood, K. Bertoldi, and C. J. Walsh, "Modeling of soft fiber-reinforced bending actuators," *IEEE Transactions on Robotics*, vol. 31, no. 3, pp. 778–789, 2015.

[8] Y. Yao, L. He, and P. Maiolino, "A simulation-based toolbox to expedite the digital design of bellow soft pneumatic actuators," in *2022 IEEE 5th International Conference on Soft Robotics (RoboSoft)*, pp. 29–34, 2022.

[9] S. Wang, L. He, A. Albini, P. Zhang, and P. Maiolino, "A magnetorheological elastomer-based proportional valve for soft pneumatic actuators," *Advanced Intelligent Systems*, vol. n/a, no. n/a, p. 2200238.

[10] G. Runge, J. Peters, and A. Raatz, "Design optimization of soft pneumatic actuators using genetic algorithms," in *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 393–400, 2017.

[11] G. Runge, M. Wiese, and A. Raatz, "Fem-based training of artificial neural networks for modular soft robots," in *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 385–392, 2017.

[12] M. J. Kochenderfer and T. A. Wheeler, *Algorithms for optimization*. 2019.

[13] K. K. Vu, C. D'Ambrosio, Y. Hamadi, and L. Liberti, "Surrogate-based methods for black-box optimization," *International Transactions in Operational Research*, vol. 24, no. 3, pp. 393–424, 2017.

[14] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on genetic algorithm: past, present, and future," *Multimedia Tools and Applications*, vol. 80, p. 8091–8126, 02 2021.

[15] A. Zolfagharian, M. A. P. Mahmud, S. Gharaie, M. Bodaghi, A. Z. Kouzani, and A. Kaynak, "3d/4d-printed bending-type soft pneumatic actuators: fabrication, modelling, and control," *Virtual and Physical Prototyping*, vol. 15, no. 4, pp. 373–402, 2020.

[16] F. Connolly, C. J. Walsh, and K. Bertoldi, "Automatic design of fiber-reinforced soft actuators for trajectory matching," *Proceedings of the National Academy of Sciences*, vol. 114, no. 1, pp. 51–56, 2017.

[17] G. Singh and G. Krishnan, "Designing fiber-reinforced soft actuators for planar curvilinear shape matching," *Soft Robotics*, vol. 7, no. 1, pp. 109–121, 2020. PMID: 31566502.

[18] R. Calandra, A. Seyfarth, J. Peters, and M. P. Deisenroth, "Bayesian optimization for learning gaits under uncertainty," *Annals of Mathematics and Artificial Intelligence*, vol. 76, p. 5–23, 02 2016.

[19] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies," *J. Mach. Learn. Res.*, vol. 9, p. 235–284, jun 2008.

[20] L. Hewing, J. Kabzan, and M. N. Zeilinger, "Cautious model predictive control using gaussian process regression," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 6, pp. 2736–2743, 2020.

[21] M. Binois and N. Wycoff, "A survey on high-dimensional gaussian process modeling with application to bayesian optimization," *ACM Trans. Evol. Learn. Optim.*, vol. 2, aug 2022.

[22] E. Kieffer, G. Danoy, P. Bouvry, and A. Nagih, "Bayesian optimization approach of general bi-level problems," GECCO '17, (New York, NY, USA), p. 1614–1621, Association for Computing Machinery, 2017.

[23] S. Mirjalili, *Genetic Algorithm*, pp. 43–55. Cham: Springer International Publishing, 2019.

[24] Stratasys, *Agilus30 Materials Data Sheet*, 2020.

5.4 Supporting Information

5.4.1 Additional Experimental Results

The remaining experimental results for the evaluation of optimisation algorithms (cf. Fig. 4) are shown in Figs. 5.S1 and 5.S2.

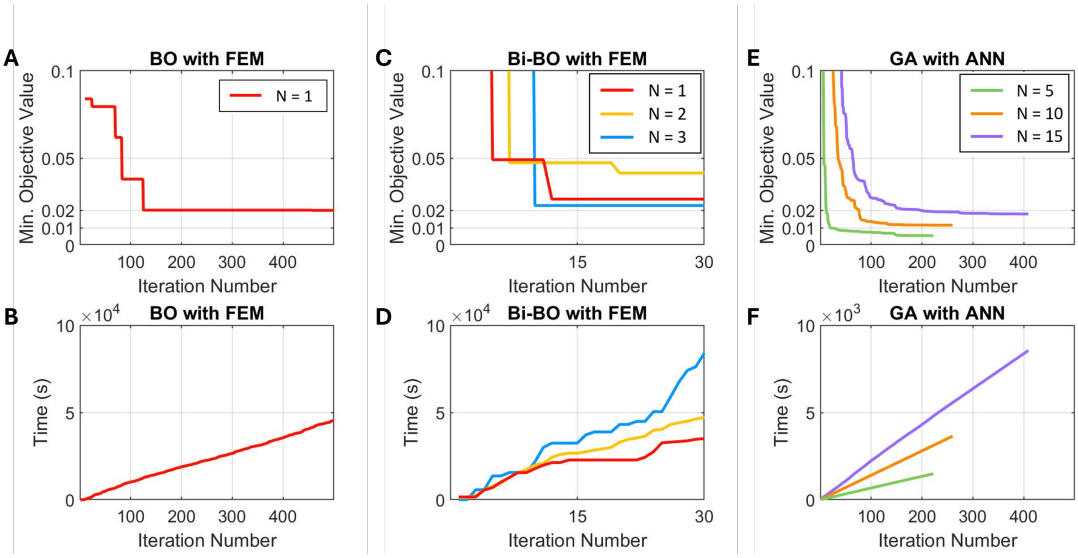


Figure 5.S1: Additional optimisation-algorithm results (set 2). Panel labels match those in Fig. 4.

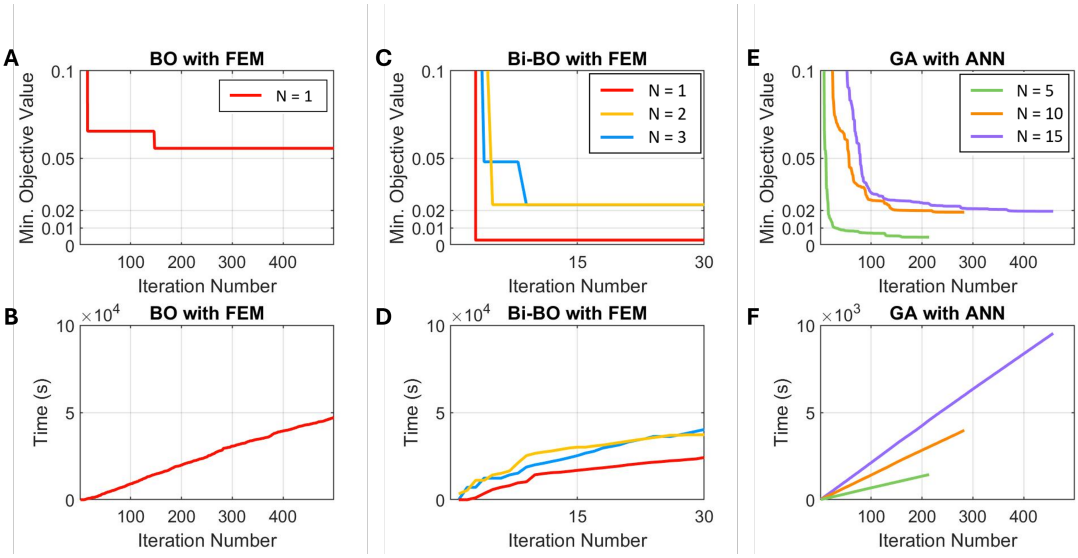


Figure 5.S2: Additional optimisation-algorithm results (set 3). Panel labels match those in Fig. 4.

5.4.2 Shape-Matching Optimisation: Low-Dimensional

We provide details for the shape-matching design optimisation, including segment-wise arc parameters and the optimised actuator parameters—shared (r, t, P) and segment-wise (R, l)—reported at different optimisation levels.

For low-dimensional design problems ($N \in \{1, 2, 3\}$) using (bi-level) BO with FEM, Fig. 5.S3 and Tables 5.S1 present the results for $N = 1$; Fig. 5.S4 and Tables 5.S2–5.S3 for $N = 2$; and Fig. 5.S5 and Tables 5.S4–5.S5 for $N = 3$.

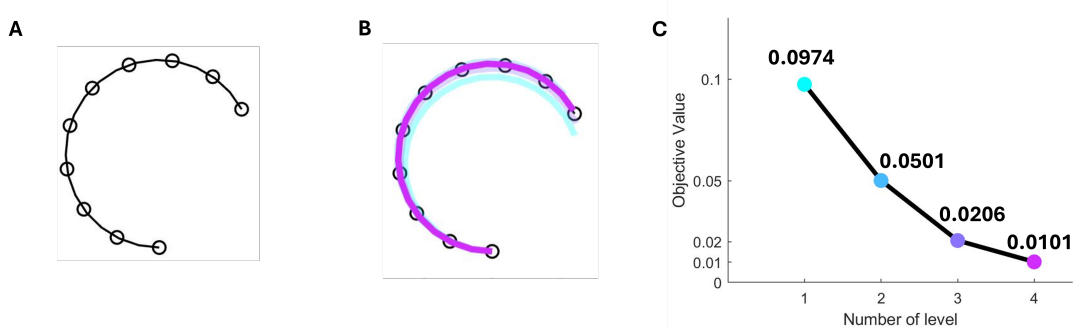


Figure 5.S3: $N = 1$. **(A)** Segmented arc. **(B)** Shapes designed with the optimised parameters at each optimisation level (colours correspond to **(C)**). **(C)** Objective values for the levels.

Segment No.	1
Arc Length (mm)	115.67
Arc Curvature (mm^{-1})	0.037

(a) Segment-wise arc parameters for $N = 1$.

No. of Level	Actuator Parameters		Segmental Parameters	Segment No. 1
1	r (mm)	3.42	R (mm)	6.80
	t (mm)	1.67		
	P (kPa)	8.55	l (mm)	7.49
2	r (mm)	2.80	R (mm)	5.60
	t (mm)	1.09		
	P (kPa)	5.43	l (mm)	7.47
3	r (mm)	4.50	R (mm)	8.10
	t (mm)	1.49		
	P (kPa)	6.40	l (mm)	7.71
4	r (mm)	3.50	R (mm)	6.74
	t (mm)	1.05		
	P (kPa)	4.03	l (mm)	7.70

(b) Optimised actuator parameters (left) and per-segment R and l (right) at each optimisation level for $N = 1$ (colour scheme as in Fig. 5.S3**C**).

Table 5.S1: Shape-matching optimisation results for $N = 1$

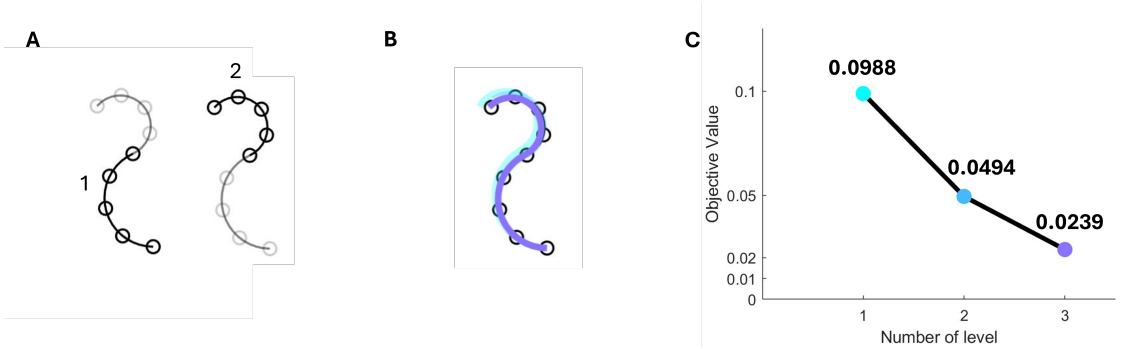


Figure 5.S4: $N = 2$. **(A)** Segmented arcs. **(B)** Shapes designed with the optimised parameters at each optimisation level (colours correspond to **(C)**). **(C)** Objective values for the levels.

Table 5.S2: Segment-wise arc parameters for $N = 2$.

Segment No.	1	2
Arc Length (mm)	70.98	58.53
Arc Curvature (mm^{-1})	0.038	0.061

Table 5.S3: Optimised actuator parameters (left) and per-segment R and l (right) at each optimisation level for $N = 2$ (colour scheme as in Fig. 5.S4**C**).

No. of Level	Actuator Parameters		Segmental Parameters	Segment No.	
				1	2
1	r (mm)	6.33	R (mm)	10.46	6.68
	t (mm)	1.70			
	P (kPa)	7.49	l (mm)	8.25	6.85
2	r (mm)	2.10	R (mm)	4.20	4.20
	t (mm)	1.01			
	P (kPa)	5.69	l (mm)	5.41	4.07
3	r (mm)	3.48	R (mm)	6.61	6.68
	t (mm)	1.42			
	P (kPa)	8.36	l (mm)	7.86	5.69

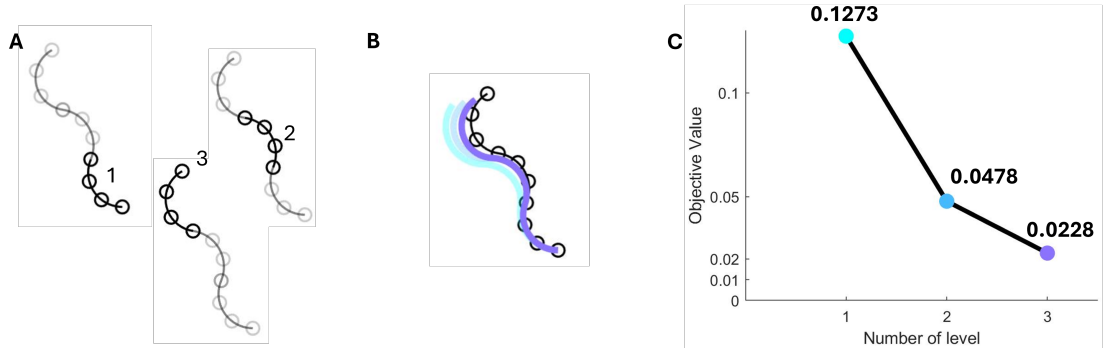


Figure 5.S5: $N = 3$. **(A)** Segmented arcs. **(B)** Shapes designed with the optimised parameters at each optimisation level (colours correspond to **(C)**). **(C)** Objective values for the levels.

Table 5.S4: Segment-wise arc parameters for $N = 3$.

Segment No.	1	2	3
Arc Length (mm)	36.35	35.38	43.00
Arc Curvature (mm^{-1})	0.054	0.052	0.058

Table 5.S5: Optimised actuator parameters (left) and per-segment R and l (right) at each optimisation level for $N = 3$ (colour scheme as in Fig. 5.S5**C**).

No. of Level	Actuator Parameters		Segmental Parameters	Segment No.		
				1	2	2
1	r (mm)	4.02	R (mm)	8.04	8.04	8.03
	t (mm)	1.24				
	P (kPa)	5.00	l (mm)	5.00	4.95	4.95
2	r (mm)	5.87	R (mm)	11.65	11.11	11.47
	t (mm)	1.56				
	P (kPa)	3.77	l (mm)	7.11	6.26	6.25
3	r (mm)	5.94	R (mm)	11.34	10.34	11.58
	t (mm)	1.49				
	P (kPa)	6.88	l (mm)	8.90	7.06	8.65

5.4.3 Shape-Matching Optimisation: High-Dimensional

For high-dimensional design problems ($N \in \{5, 10, 15\}$) using GA with NN-based surrogate model, Fig. 5.S6 and Tables 5.S6–5.S7 present the results for $N = 5$; Fig. 5.S7 and Tables 5.S8–5.S9 for $N = 10$; and Fig. 5.S8 and Tables 5.S10–5.S11 for $N = 15$.

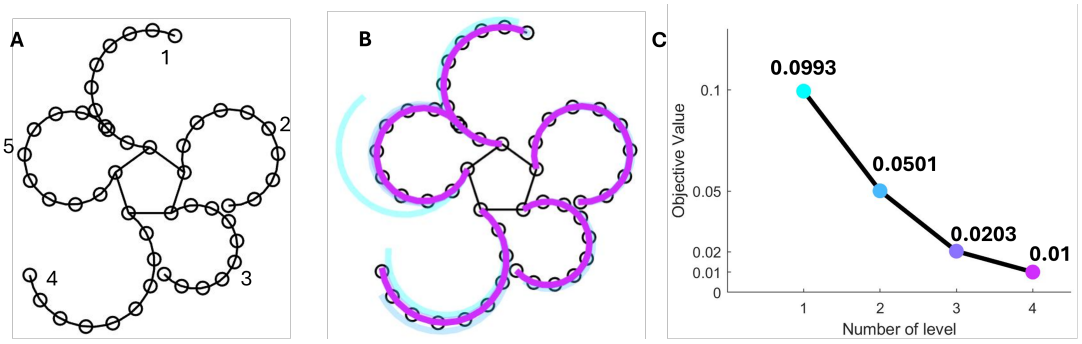


Figure 5.S6: $N = 5$. **(A)** Segmented arcs. **(B)** Shapes designed with the optimised parameters at each optimisation level (colours correspond to **(C)**). **(C)** Objective values for the levels.

Table 5.S6: Segment-wise arc parameters for $N = 5$.

Segment No.	1	2	3	4	5
Arc Length (mm)	115.56	134.26	106.38	135.20	134.21
Arc Curvature (mm^{-1})	0.031	0.038	0.044	0.029	0.039

Table 5.S7: Optimised actuator parameters (left) and per-segment R and l (right) at each optimisation level for $N = 5$ (colour scheme as in Fig. 5.S6**C**).

No. of Level	Actuator Parameters		Segmental Parameters	Segment No.				
				1	2	3	4	5
1	r (mm)	2.24	R (mm)	3.82	4.00	3.96	3.79	3.72
	t (mm)	1.08						
	P (kPa)	9.05	l (mm)	4.88	4.95	4.41	4.54	4.54
2	r (mm)	2.17	R (mm)	3.96	3.89	3.89	3.65	3.81
	t (mm)	1.06						
	P (kPa)	8.45	l (mm)	5.61	4.77	4.42	4.48	4.42
3	r (mm)	2.18	R (mm)	3.93	3.92	3.93	3.70	3.84
	t (mm)	1.06						
	P (kPa)	8.55	l (mm)	5.52	4.83	4.43	4.45	4.44
4	r (mm)	2.22	R (mm)	3.80	3.94	3.95	3.73	3.89
	t (mm)	1.06						
	P (kPa)	8.89	l (mm)	4.73	4.83	4.44	4.55	4.50

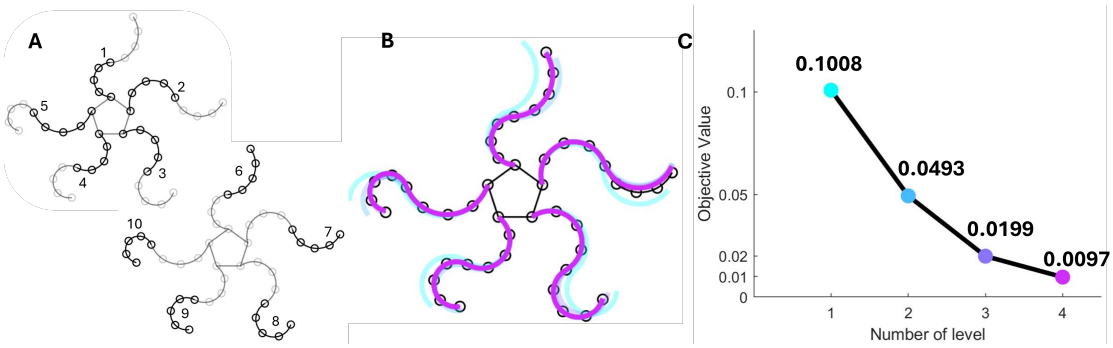


Figure 5.S7: $N = 10$. **(A)** Segmented arcs. **(B)** Shapes designed with the optimised parameters at each optimisation level (colours correspond to **(C)**). **(C)** Objective values for the levels.

Table 5.S8: Segment-wise arc parameters for $N = 10$.

Segment No.	1	2	3	4	5	6	7	8	9	10
Arc Length (mm)	51.93	72.51	71.17	59.90	73.80	60.76	60.41	68.22	60.16	54.23
Arc Curvature (mm^{-1})	0.060	0.044	0.046	0.051	0.033	0.035	0.040	0.057	0.064	0.077

Table 5.S9: Optimised actuator parameters (left) and per-segment R and l (right) at each optimisation level for $N = 10$ (colour scheme as in Fig. 5.S7**C**).

No. of Level	Actuator Parameters		Segmental Parameters	Segment No.									
				1	2	3	4	5	6	7	8	9	10
1	r (mm)	2.01	R (mm)	3.74	3.60	3.49	3.68	3.38	3.53	3.46	3.69	3.75	3.61
	t (mm)	1.00											
	P (kPa)	9.63	l (mm)	4.46	4.27	4.04	4.25	4.2	4.68	4.02	4.02	4.64	4.08
2	r (mm)	2.04	R (mm)	4.05	3.86	3.77	3.85	3.58	3.61	3.68	3.90	4.06	4.03
	t (mm)	1.00											
	P (kPa)	7.27	l (mm)	4.38	4.73	4.24	4.29	4.26	4.26	4.29	4.08	4.32	4.25
3	r (mm)	2.01	R (mm)	3.81	3.60	3.55	3.66	3.43	3.53	3.47	3.70	3.84	3.95
	t (mm)	1.00											
	P (kPa)	9.60	l (mm)	4.48	4.42	4.07	4.26	4.31	4.70	4.04	4.09	4.32	4.12
4	r (mm)	2.07	R (mm)	3.89	3.77	3.67	3.84	3.53	3.54	3.61	3.86	3.93	4.10
	t (mm)	1.01											
	P (kPa)	9.43	l (mm)	4.31	4.74	4.20	4.60	4.39	4.29	4.31	4.33	4.27	4.25

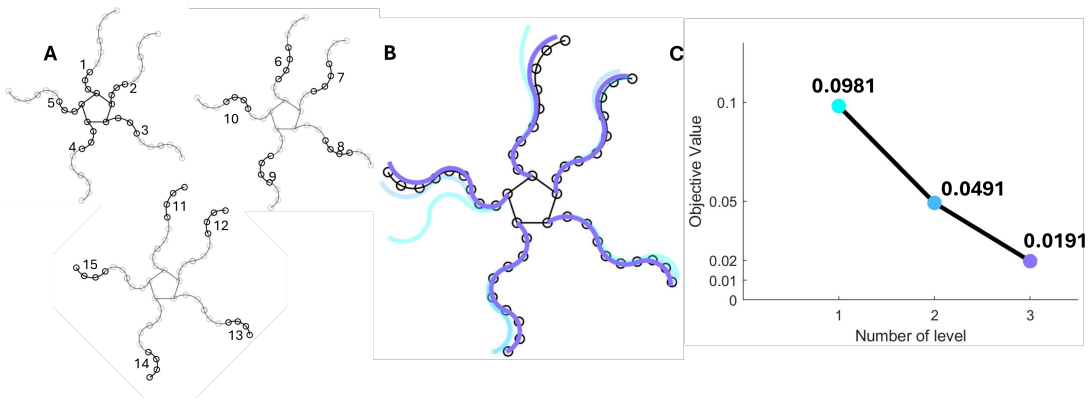


Figure 5.S8: $N = 15$. **(A)** Segmented arcs. **(B)** Shapes designed with the optimised parameters at each optimisation level (colours correspond to **(C)**). **(C)** Objective values for the levels.

Table 5.S10: Segment-wise arc parameters for $N = 15$.

Segment No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Arc Length (mm)	36.93	34.65	48.88	43.38	39.62	44.42	46.54	38.89	44.88	36.73	49.31	45.20	36.12	40.71	43.88
Arc Curvature (mm^{-1})	0.068	0.052	0.040	0.053	0.069	0.028	0.042	0.045	0.057	0.065	0.034	0.049	0.052	0.059	0.046

Table 5.S11: Optimised actuator parameters (left) and per-segment R and l (right) at each optimisation level for $N = 15$ (colour scheme as in Fig. 5.S8**C**).

No. of Level	Actuator Parameters		Segmental Parameters	Segment No.														
				1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	r (mm)	2.07	R (mm)	4.03	3.85	3.65	3.83	3.96	3.57	3.76	3.81	3.95	4.06	3.32	3.87	3.88	3.94	3.88
	t (mm)	1.00																
	P (kPa)	7.69	l (mm)	4.09	4.27	4.15	4.11	4.48	4.54	4.52	4.56	4.33	4.27	4.12	4.39	4.12	4.09	4.40
2	r (mm)	2.07	R (mm)	4.05	3.87	3.67	3.83	3.98	3.57	3.76	3.83	3.95	3.96	3.58	3.87	3.88	3.93	3.82
	t (mm)	1.00																
	P (kPa)	7.69	l (mm)	4.09	4.26	4.16	4.11	4.02	4.55	4.36	4.48	4.33	4.27	4.13	4.41	4.19	4.10	4.37
3	r (mm)	2.00	R (mm)	3.83	3.69	3.50	3.71	3.86	3.33	3.55	3.60	3.70	3.80	3.43	3.63	3.8	3.72	3.59
	t (mm)	1.00																
	P (kPa)	9.33	l (mm)	4.07	4.23	4.09	4.33	4.05	4.10	4.24	4.25	4.03	4.08	4.17	4.14	4.22	4.07	4.14

6

A Generalised Task-Driven Design Pipeline for Soft Actuators

Contents

6.1 Positioning with respect to the SOTA 91
6.1.1 Problem 91
6.1.2 Objective 92
6.1.3 Contributions 93
6.2 Assumptions, Validity Domain and Limitations 93
6.3 Contributed Paper 94
6.4 Supporting Information 112
6.4.1 Supporting Material 112
6.4.2 Supporting Video 121

6.1 Positioning with respect to the SOTA

6.1.1 Problem

Task-driven design has emerged as an important direction in robotics, aiming to automatically synthesise robot morphologies directly from task requirements rather than relying on designers to manually formulate actuator-level performance objectives. This paradigm is particularly relevant to soft robotics, where the physical body plays a fundamental role in task execution, reflecting the broader principles

of embodied intelligence and morphological computation. However, these same characteristics also make task-driven design substantially more challenging, owing to the strong nonlinear coupling between actuator geometry, material deformation, environmental interaction, and task performance.

Existing task-driven design approaches can be broadly categorised into high-fidelity simulation-based pipelines [98, 99, 101], proxy-based optimisation frameworks [43, 80, 102, 103], and differentiable simulation approaches [79, 83, 109, 110]. High-fidelity pipelines provide physically grounded task evaluation but remain computationally expensive for repeated optimisation, whereas proxy-based frameworks improve computational efficiency through abstraction-based representations at the expense of actuator-specific mechanics and fabrication consistency. Differentiable simulation enables end-to-end optimisation by exploiting model sensitivities, but typically relies on simplified differentiable representations or substantial computational acceleration to remain tractable.

Despite their complementary strengths, existing task-driven design approaches remain largely application-specific because their modelling, simulation, and optimisation components are typically developed together for a particular actuator architecture, actuation mechanism, and task formulation. As a result, extending these workflows to new actuator designs or task scenarios often requires substantial redevelopment rather than systematic extension. Accordingly, in this chapter, generalisation refers to the reuse of a computational design pipeline across parameterised actuator families and task scenarios represented in simulation, rather than universal applicability across all soft actuator designs and tasks.

6.1.2 Objective

The objective of this chapter is to address the challenge of generalising task-driven design for soft actuators by developing a unified computational design pipeline that can be reused across parameterised actuator designs and task scenarios while preserving the physical fidelity required for task-level evaluation and the computational efficiency required for optimisation.

6.1.3 Contributions

This work contributes a unified computational pipeline for task-driven design of parameterisable soft actuators. Specifically, it

1. combines high-fidelity FEM simulations with reduced-order pseudo-rigid joint models to balance physical accuracy and computational efficiency, while enabling direct integration into rigid-body simulators for physics-based task-level simulation and optimisation under diverse environmental configurations, including contact, gravity, and external loading;

2. enables design-conditioned behaviour modelling through a meta-model, supporting rapid adaptation to new design parameters;

3. validates the proposed pipeline across different actuator types and representative task scenarios, demonstrating its applicability to parameterised soft actuator design.

6.2 Assumptions, Validity Domain and Limitations

The proposed framework is developed under several modelling assumptions that enable computationally efficient task-driven design while maintaining sufficient physical realism for task-level evaluation. These assumptions concern the reduced-order representation of soft actuators, their implementation in a rigid-body simulation environment, and the modelling of environmental interactions.

Existing studies have shown that representing continuum soft robots as articulated rigid-link systems provides an effective compromise between physical realism and computational efficiency for task-level simulation and learning-based control [68–70, 105]. These approaches typically require system identification to calibrate the equivalent joint properties for each robot instance. The proposed framework adopts the same reduced-order modelling philosophy but replaces robot-specific calibration with an FEM-derived design-conditioned behavioural model,

allowing the joint behaviour to be generated directly from actuator geometry and actuation parameters.

Embedding the reduced-order model into a rigid-body simulator enables efficient task-level evaluation while preserving the dominant kinematic behaviour of the soft actuator. However, the representation approximates deformation through articulated joints rather than continuum mechanics. As a result, global actuator motion is captured, whereas local soft-body deformation—including stress and strain distributions, pressure-induced inflation, cross-sectional expansion, and other material-level deformation—is not explicitly represented.

Environmental interactions are handled by the contact model provided by the rigid-body simulator. In PyBullet, contact is resolved through point-based constraints. For rigid or convex geometries, only a limited number of contact points are typically generated, leading to rapid changes in contact locations and forces that cannot faithfully reproduce the distributed contact behaviour of compliant materials. To better approximate soft contact, friction anchors are enabled to maintain persistent contacts. Nevertheless, the contact model remains point-based and cannot explicitly represent local soft-body deformation or distributed surface contact.

Therefore, the proposed framework is intended for parameterised soft actuator families undergoing quasi-static deformation, where task-level behaviour is primarily governed by the dominant kinematic deformation captured by the reduced-order surrogate model. The framework is intended for use during the design optimisation stage, where computational efficiency and task-level behavioural fidelity are prioritised over detailed continuum mechanics.

6.3 Contributed Paper

The work has been published as a journal paper in Advanced Intelligent Systems. The paper is attached on the following pages.

Generalized Task-Driven Design of Soft Robots via Reduced-Order FEM-based Surrogate Modeling

Yao Yao
Oxford Robotics Institute
University of Oxford, UK

David Howard
CSIRO Robotics
CSIRO, Australia

Perla Maiolino*
Oxford Robotics Institute
University of Oxford, UK

Abstract

Task-driven design of soft robots requires models that are physically accurate and computationally efficient, while remaining transferable across actuator designs and task scenarios. However, existing modeling approaches typically face a fundamental trade-off between physical fidelity and computational efficiency, which limits model reuse across design and task variations and constrains scalable task-driven optimization. This paper presents a unified reduced-order finite element method (FEM)-based surrogate modeling pipeline for generalized task-driven soft robot design. High-fidelity FEM simulations characterize actuator behavior at the modular level, from which compact surrogate joint models are constructed for evaluation within a pseudo-rigid body model (PRBM). A meta-model maps actuator design parameters to surrogate representations, enabling rapid instantiation across a parameterized actuator family. The resulting models are embedded into a PRBM-based simulation environment, supporting task-level simulation and optimization under realistic physical constraints. The proposed pipeline is validated through sim-to-real transfer across multiple actuator types, including bellow-type pneumatic actuators and a tendon-driven soft finger, as well as two task-driven design studies: soft gripper co-design via Reinforcement Learning (RL) and 3D actuator shape matching via evolutionary optimization. The results demonstrate high accuracy, efficiency, and reliable reuse, providing a scalable foundation for autonomous task-driven soft robot design.

1 Introduction

Soft robots are increasingly valued for their adaptability, compliance, and ability to achieve a wide range of behaviors through the combination of material properties, morphology, and control [1, 2, 3]. However, these advantages also make the design of soft robots a significant challenge, leading to vast and complex design spaces where manual or heuristic approaches are typically insufficient to ensure optimal performance in many task applications. This has led to an increasing focus on task-driven design, wherein morphology and control are systematically tailored to meet specific application objectives via computational approaches [4, 5].

Such approaches rely heavily on accurate and scalable modeling to explore large design spaces. However, soft systems pose particular difficulties in this context, especially when predicting forces and self or environmental interactions, due to their inherent softness and material nonlinearity. This not only challenges the commonly discussed issue of accuracy and efficiency, but also reveals a critical modeling bottleneck: the lack of generalizable models that can be reused across parameterized geometries, actuation types, and task scenarios. Here, generalization refers to the ability of a model to retain accuracy across variations within a parameterized design space and across different task-level conditions represented in simulation, rather than being confined to case-specific configurations.

This limitation stems from a fundamental trade-off in modeling soft robots. Despite a wide variety of modeling techniques having been explored, including first-principles-based methods, physics-based simulation, and data-driven or reduced-order models, most approaches follow one of two dominant directions: high-fidelity modeling and simplified or reduced-order representations. Approaches based on high-fidelity modeling, such as FEM, offer accurate predictions but are often too computationally expensive for repeated task-level evaluation, which in practice encourages the use of simplified or proxy objectives tied to specific tasks and assumptions [6, 7]. In contrast, simplified or reduced-order models are computationally efficient, but typically rely on assumptions and approximations for particular designs or operating conditions, limiting their transferability to new geometries, loading regimes, or task settings [8, 9].

*Corresponding author: perla.maiolino@eng.ox.ac.uk

As a result, existing approaches fall short of providing models that are simultaneously accurate and efficient while remaining sufficiently general for task-driven design. This lack of generalization limits model scalability and deployability across broader design contexts, thereby hindering the development of general-purpose modeling tools that the research community can readily build upon. Addressing this limitation is essential for building flexible and reusable design pipelines in soft robotics.

To overcome this challenge, this paper introduces a unified modeling pipeline for task-driven design of soft robots. Specifically, we:

- Use high-fidelity FEM simulations to generate actuator data that captures coupled relationships between actuation, deformation, and force at the modular level;
- Create surrogate models from this data using either polynomial regressions or small neural network (NN)s, serving as joint models within the PRBM framework, to significantly accelerate simulation while maintaining FEM-level accuracy;
- Develop a meta-model that maps actuator design parameters to surrogate features, enabling rapid model interpolation within a parameterized actuator family;
- Integrate the resulting models into a fast and flexible PRBM-based simulation environment, such as PyBullet, supporting efficient task-level simulation and optimization under real-world physical constraints such as contact, loading, and gravity.

Together, these contributions support pipeline reuse across design variations and task conditions, offering a practical pathway toward scalable task-driven design. Notably, the proposed approach is scoped to parameterized actuator families and is not intended to generalize across fundamentally different soft robotic morphologies or actuation principles.

We illustrate our contributions using a range of different soft actuators, including pneumatically actuated bellow-type soft actuators, which are widely used in soft robotic field due to their large deformation, adaptability, straightforward control, and 3D printing compatibility. These actuators remain challenging to model accurately while efficiently due to effectively infinite degrees of freedom and distributed actuation [10]. We also include a tendon-driven finger, demonstrating that the proposed pipeline extends beyond a single actuator type. Across diverse applications including soft gripping and 3D shape-matching, we demonstrate that the pipeline supports both accurate physical modeling and efficient task-level optimization. An overview of the workflow is provided in Figure 1.

2 Literature Review

Modeling plays a critical role in the design of soft robots, as it directly impacts both the accuracy of the predicted behavior and the speed of the automated design process. A wide range of modeling approaches is available, each with its own trade-offs between accuracy, efficiency, and generalizability.

Analytical models, such as rod formulations [11] or minimum potential energy principle [12], are valued for their simplicity and computational efficiency. However, they often rely on simplifying assumptions, such as uniform curvature, constant cross-sections, or linearized material behavior. These assumptions limit flexibility and often require re-derivation or re-calibration when designs, materials, or boundary conditions change.

Geometric models, such as Piecewise Constant Curvature (PCC) and function-based curves, provide efficient representations of the kinematics of soft actuators and intuitive mappings from actuation inputs to robot configurations [13]. However, they tend to break down under unexpected external forces such as gravity or contact [9].

Discrete models, such as PRBMs, approximate continuum deformation by connecting rigid segments with compliant joints [14]. They are lightweight and compatible with existing rigid-body simulators (e.g., PyBullet), enabling reliable and efficient task- and environment-level simulation [15, 16]. However, they typically require careful parameter calibration to map the lumped behavior back to distributed soft structures. This process is time-consuming and often lacks generalizability beyond the specific robot for which the model is tuned. Such limitations highlight the need for modeling approaches that can be reused across actuator designs, enabling broader applicability within the research community.

Numerical methods, especially FEM, offer the highest fidelity by solving governing equations over the continuum. FEM is widely used for simulating soft robots because it provides high accuracy and strong generality, accommodating arbitrary geometries, materials, actuation, and boundary conditions, with established tool support such as COMSOL Multiphysics®, ANSYS®, and Abaqus® [17, 7, 18]. Other numerical approaches have also

been explored for deformable simulation, particularly in the computer graphics community, including extended position-based dynamics (XPBD) [19] and material point methods (MPM) [20]. XPBD enables efficient and stable simulations but often yields less reliable mechanical responses, while MPM can capture large deformations but may introduce numerical diffusion that affects accurate force responses. However, the high computational cost and convergence challenges of FEM under complex contact and large-deformation conditions limit its applicability in large-scale design optimization. As a result, task-level performance is often approximated through actuator-level objectives that are more tractable to evaluate but are closely tied to specific modeling assumptions and task formulations. This limitation has motivated efforts in model order reduction techniques, which seek to retain fidelity while reducing computational expense [6, 21].

Model order reduction (MOR) techniques aim to accelerate FEM by reducing system dimensionality while retaining dominant deformation modes. Examples include proper orthogonal decomposition (POD) [22, 8], condensed FEM formulations [9], and Cosserat rod implementations such as SoRoSim [23]. These methods improve computational efficiency and can reproduce complex deformations with fewer degrees of freedom. However, the reduced bases or governing formulations are typically derived for specific geometries, boundary conditions, or actuation schemes, making the resulting models configuration-specific and difficult to reuse in task-driven design workflows.

Surrogate models learn input–output relationships from high-fidelity simulations or experiments, providing fast evaluation once trained. Approaches include regression models, Gaussian processes, and NNs that map actuation to deformation and force responses [24, 25]. Compared with physics-based reductions, data-driven surrogates can offer greater flexibility. However, their performance is inherently tied to the coverage of the training data, trained under fixed loading conditions or tailored to a specific design, making it hard to extrapolate reliably when tasks, geometries, or operating regimes change.

In summary, existing approaches present clear trade-offs. Analytical, geometric, and discrete models are efficient but rely on restrictive assumptions that limit accuracy and transferability. FEM provides accuracy and generality across arbitrary geometries and boundary conditions, but is too costly to support large-scale task-level design optimization. Physics-based model reduction improves efficiency but remains tied to specific designs or loading conditions. Data-driven surrogates accelerate evaluation and can capture complex behaviors, yet they are often restricted to fixed training scenarios and lack generalization across tasks. Overall, what is still missing is a pipeline that unifies these ideas into models that are accurate, efficient, and reusable across designs and task scenarios. Addressing this gap is essential not only for advancing task-driven design of soft robots but also for providing a modeling foundation that can be extended and adopted by the broader soft robotics community.

3 Materials and Methods

This section introduces the proposed modeling pipeline for task-driven design of soft robots. The pipeline combines four components: (i) high-fidelity FEM simulations to capture coupled actuation–deformation–force responses; (ii) surrogate models that encode these behaviors into compact representations for fast evaluation, and (iii) PRBM-based simulation in PyBullet, embedding surrogates for task-level simulation. (iv) meta-models that map design parameters to surrogate models, supporting scalability across new actuator designs.

To explicitly illustrate this workflow, as presented in Figure 1, the pipeline starts from a parameterized modular actuator design, whose behavior is first characterized using high-fidelity FEM under both free-loading and constrained conditions. Building on the FEM data, compact surrogate joint models are derived to approximate the nonlinear relationship between actuation input and mechanical response at the modular level for use within a PRBM-based simulation. These surrogates significantly reduce computational cost while preserving physical fidelity.

To support design generalization, the surrogate construction process is repeated across multiple actuator designs, and the resulting model parameters are learned by a higher-level meta-model that maps from design parameters to surrogate representations. This enables rapid on-demand generation of behavior models for unseen designs via interpolation within a parameterized actuator family, eliminating the need for repeated FEM simulations. Finally, the design-conditioned surrogate models are embedded into PRBM-based simulation in PyBullet, enabling task-level simulation with realistic physical interactions. This integrated environment also supports efficient task-level design using SOTA learning and optimization algorithms.

3.1 FEM Data Generation

Existing surrogate models are often developed for specific loading conditions, such as free deformation, and may not capture the coupled effects of actuation and external forces. However, task-level simulations within a PRBM

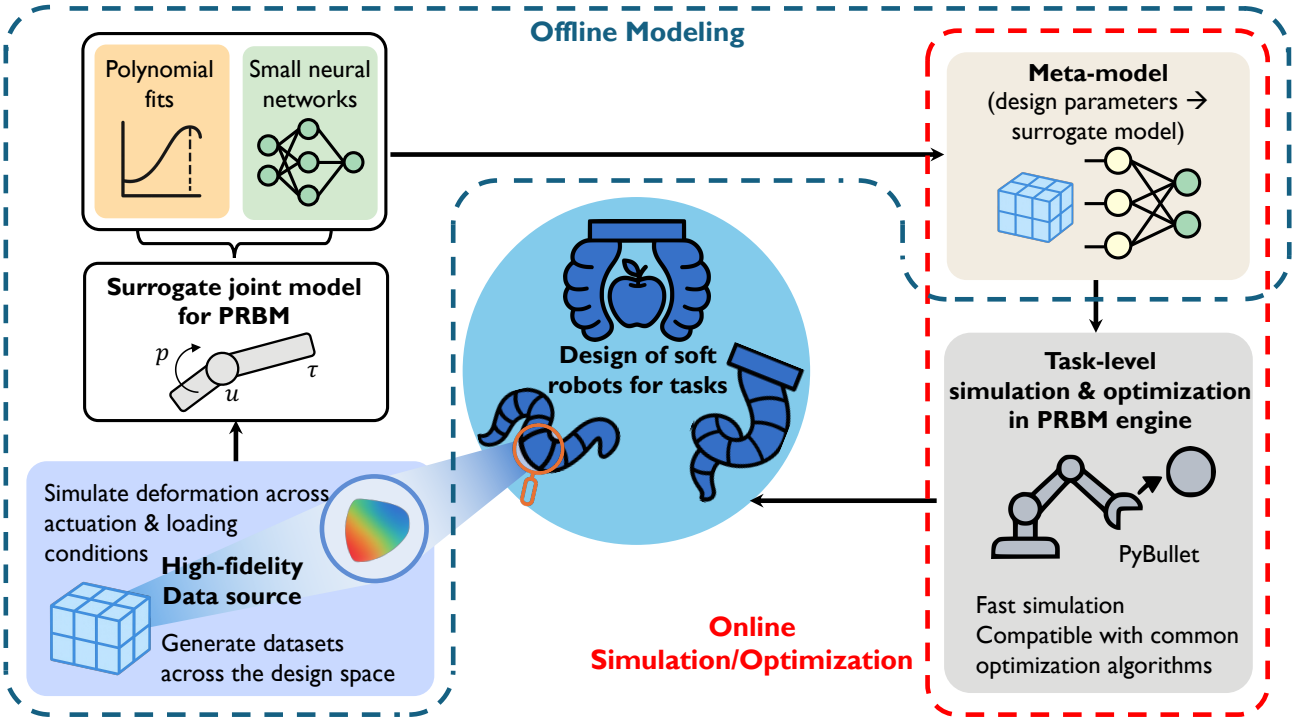


Figure 1: Schematic overview of the proposed pipeline for modeling and task-driven design of soft robots. The process starts from a well-parameterized modular actuator design, which is simulated using FEM to generate high-fidelity training data. Compact surrogate models are then constructed from these data as joint models, represented using either polynomial fits or small NNs, for use in PRBM-based simulations. By repeating this process across different designs, meta-models are constructed to capture the mapping from design parameters to surrogate models. The resulting surrogate and meta-models can be integrated with PRBM-based simulators and SOTA optimization algorithms, enabling fast task-level simulation and task-driven design optimization.

framework require joint laws that reflect both deformation and the corresponding forces. To address this, we design FEM simulations that provide a comprehensive mapping between actuation input, deformation, and reaction forces at the modular level.

The FEM simulations are conducted under quasi-static assumptions to characterize the dominant elastic response of the actuator modules. This is appropriate for the considered systems, as the actuation and deformation processes occur at relatively low speeds, and inertial effects are negligible compared to elastic forces. The resulting actuation–deformation-force relationships are used as joint models, which are subsequently embedded into task-level simulations to account for gravity, contact, and external interactions.

The procedure consists of two complementary conditions. In the free-loading condition, actuation inputs are incrementally applied to the actuator while external loads are absent, yielding the natural actuation–deformation response. In the constrained condition, the actuation input is held fixed while external forces are applied to vary the deformation, enabling explicit capture of the force–deformation relationship under constraint. The FEM data are sampled over actuation ranges determined by the intended task operating conditions, while the external force range is constrained by stable FEM convergence. These simulations collectively yield coupled actuation–deformation–force data used to derive compact surrogate models.

For clarity, an illustrative example using a translational bellow-soft pneumatic actuator (SPA) is provided in the Supplementary Material Section 1.1, where specific FEM configurations, parameter sweeps, and numerical results are detailed.

3.2 Surrogate Model via Polynomial Fits

The FEM data described above provides actuator states under both free-loading and constrained conditions. To enable efficient task-level simulation within a PRBM framework, we derive a compact analytical surrogate that represents the joint force as a function of actuation and deformation. Here τ denotes the generalized joint effort

associated with the deformation coordinate u. Depending on the actuator configuration, this effort corresponds to a torque for rotational deformation or a force for translational deformation. This surrogate is constructed via polynomial approximations of stiffness terms obtained from FEM data.

Based on the principle of virtual work, the joint force τ can be expressed as the derivative of the actuator's potential energy U with respect to the deformation u, which itself depends on both the actuation parameter p and the deformation u:

$$\tau(p,u) = \frac{\mathrm{d}U(p,u)}{\mathrm{d}u} = \frac{\partial U(p,u)}{\partial u} + \frac{\partial U(p,u)}{\partial p(u)} \frac{\mathrm{d}p(u)}{\mathrm{d}u} \tag{1}$$

Given the quasi-static FEM results, this derivative can be approximated using finite differences between adjacent simulation states. Distinguishing between free-loading (actuation varied, no external force) and constrained conditions (actuation fixed, external force applied), the joint forces can be decomposed into actuation and elastic contributions. By differentiating further with respect to u, we obtain stiffness terms $k_a(p)$ and $k_e(u)$, which depend solely on actuation and deformation, respectively. This separability allows each term to be independently approximated using low-order polynomial fits. Without this decomposition, the joint force would need to be approximated using a single high-order polynomial over the combined actuation–deformation space. Such models typically require significantly more data to avoid overfitting and may lead to unstable behavior when embedded within iterative physics simulations. By separating the actuation-dependent and deformation-dependent stiffness terms, the surrogate fitting problem becomes lower-dimensional and more data-efficient. Such a decomposition constitutes an approximation, as it neglects higher-order coupling between actuation and deformation, and is expected to remain valid within the moderate pressure and deformation ranges considered in this study.

The resulting surrogate provides an explicit, closed-form mapping from (p, u) to the net force:

$$\tau_n(p,u) = -\int_0^u \big(k_a(p) + k_e(u)\big)\,\mathrm{d}u + \tau_n(p,0), \tag{2}$$

where $\tau_n(p,0)$ is determined from free-loading conditions. All functions are represented by polynomial regressions fitted to FEM data, yielding a lightweight model suitable for use in rigid-body-based physics engines. Detailed derivations and an illustrative case study on a translational bellow-SPA are provided in the Supplementary Material Section 1.2 and 1.3. When extending the module-level surrogate to multi-module actuators through module-wise composition, additional system-level limitations arise. In particular, cumulative inter-module effects—such as compliance accumulation and boundary constraints—are not explicitly modeled, leading to a systematic underestimation of deformation as the number of modules increases, as quantified in Supplementary Figure S2f.

3.3 Surrogate Model via Small Neural Networks

While polynomial fits offer interpretable closed-form surrogates, their approximation accuracy is inherently limited, especially when actuator behavior becomes strongly nonlinear. For example, in bellow-SPA modules with internal constraints, even higher-order fits struggle to capture the underlying relationships accurately (as demonstrated via Supplementary Figure S3). To address this, we employ small NNs as alternative surrogates.

The network takes the actuation parameter p and deformation u as inputs and outputs the corresponding net joint force $\tau_n(p,u)$:

$$(p,u) \rightarrow \tau_n(p,u) \tag{3}$$

Unlike polynomial regression, NNs flexibly approximate nonlinear relationships without requiring explicit functional assumptions. Importantly, the networks are designed to remain lightweight, ensuring efficient evaluation within PRBM-based task simulations.

After training, the NNs are directly integrated into rigid-body-based physics engines such as PyBullet for task simulation, where they are used as interpolation models within the sampled operating ranges of actuation and deformation, extending applicability to more complex actuator responses.

We implement small feed-forward NNs trained on FEM-derived data spanning both free-loading and constrained conditions. They provide significantly improved accuracy compared with polynomial fits, while remaining lightweight for deployment in task simulations. However, compared with the analytical polynomial surrogate, NN-based models may produce less predictable responses when evaluated slightly outside the training distribution in physics-based simulations, and incur higher computational cost during evaluation. Detailed architecture, training setup, and validation on a representative bellow-SPA are provided in the Supplementary Material Section 1.4.

3.4 Meta-Model for Design-Conditioned Surrogate Generation

The purpose of the meta-model is to establish a direct mapping from actuator design parameters (e.g., geometric variables such as module radius, length, or thickness) to the corresponding reduced-order surrogate representations, enabling task-level simulations for previously unseen designs through design-space interpolation without repeating high-fidelity FEM simulations. By learning this design-to-behavior relationship, the meta-model enables rapid generation of design-conditioned surrogate models and therefore serves as the key interface between offline physical modeling and online task-driven design and optimization.

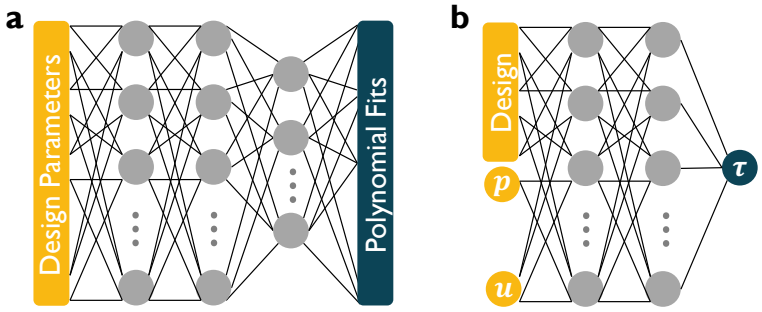


Figure 2: Meta-models for design-conditioned surrogate generation. (a) A NN mapping the design parameters to the corresponding polynomial coefficients of the surrogate model. (b) A NN mapping the combined design parameters and modular actuation inputs to the resulting net force.

For surrogate models constructed using polynomial fits, the meta-model is implemented as a neural network that maps the actuator design parameters to the corresponding set of polynomial coefficients, as illustrated in Figure 2(a). In this way, the polynomial surrogate for a new design can be generated instantly from its design parameters, allowing fast evaluation of actuator behavior within the simulation environment.

For surrogate models constructed using small NNs, all available data across different designs are directly aggregated to form a unified training dataset. For each sample, the design parameters and the modular variables (p, u) are concatenated as the inputs, and the corresponding net force τ is taken as the output. A NN is then trained to capture the global relationship between the combined design–actuation inputs and the resulting behavioral response, as shown in Figure 2(b). This formulation enables the surrogate to generalize jointly over both the design space and the actuation space within a single compact model.

4 Results and Discussion

This section reports the performance of the proposed approach. Two complementary evaluations are presented: (i) the sim-to-real transfer of the surrogates on tasks, and (ii) their ability to support task-driven design through optimization. The goal of these evaluations is to demonstrate that the proposed pipeline is not tailored to a single application, but instead offers a general modeling and optimization approach capable of supporting heterogeneous task requirements. This is addressed by applying the pipeline to tasks that differ fundamentally in both physical interaction and optimization objectives. Specifically, grasping is a contact-based task in which forces are mainly transmitted through friction, and success is defined by the stability of the grasp and the ability to lift objects under load, whereas 3D shape-matching requires continuous geometric accuracy under distributed loading and gravity. The following subsections present detailed, case-specific analyses for each task and actuator type. A brief general discussion at the end of this section summarizes shared assumptions and limitations across these evaluations.

4.1 Surrogate Models in Task Simulations

This subsection focuses on the sim-to-real validation of the proposed surrogate models in several task simulations involving complex deformations and contact interactions. The objective is to verify that the surrogate models can accurately reproduce actuator behavior across different actuation principles and task types, without relying on full FEM simulations at runtime.

4.1.1 Contact-Based Grasping with Polynomial Surrogate

To evaluate the capability of the surrogate models constructed using polynomial fits in task-level simulation, and to validate the proposed pipeline on a representative class of contact-dominated manipulation tasks, we simulated a soft gripper performing grasping, where the smooth and efficient numerical characteristics of polynomial surrogates facilitate stable physics-based simulation. The gripper design is based on the default bellow-SPA from prior work, featuring a strain-limiting side that constrains expansion on one face and thereby promotes directional bending during pressurization [26].

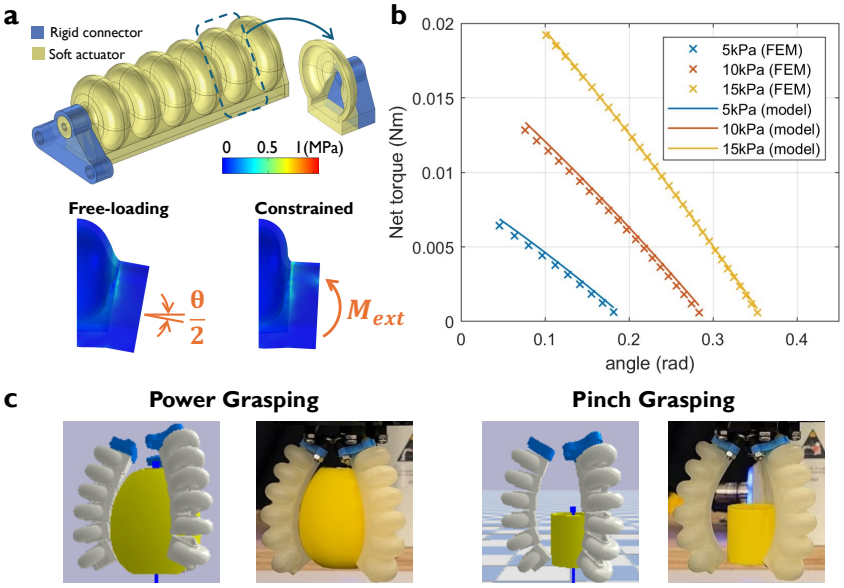


Figure 3: Validation of the polynomial-based surrogate model through a soft gripper grasping task. (a) FEM model of the bellow-SPA used in the gripper. (b) Comparison of net torque vs. angular deflection under different pressure levels obtained from the FEM simulation and the surrogate model. (c) Two grasping modes with different objects, compared between simulation and experiment.

As illustrated in Figure 3a, we simulated a half-module, closed with an end cap to facilitate smooth deformation, under conditions of free pressurization and externally applied moment M_{ext} at various fixed internal pressures, obtaining angular deflection θ. In this study, the pressure is sampled from 0 kPa to 15 kPa with 0.5 kPa as an interval, and the externally applied moment is sampled from 0 N · m to 0.02 N · m with an interval of 0.0006 N · m. Following the previously described modeling process, we compared the model-predicted net torque to the applied external moment across different pressure values, resulting in a maximum error of 8.26% relative to the moment range.

Grasping simulations were then carried out in PyBullet using a physics-based model of the soft gripper. The gripper consists of three actuator fingers equally spaced around a central axis, with each finger modeled as a chain of links connected by revolute joints. Visual and collision geometry are taken directly from the actuator CAD STL mesh files, while the mass properties of each link are estimated from the actuator volume and material density. The lateral friction coefficient between the actuator and PLA-printed objects is set to 0.8 [27], and anchor friction is enabled to mimic soft contact behavior. To better approximate real-world conditions, 5% random noise is introduced to the object placement, friction coefficient, and actuation pressure in the simulations. To ensure a fair sim-to-real comparison, the simulated grasping setup is configured to match the physical experiments in terms of gripper geometry, object placement, and actuation conditions. Details of the ablation study used for selecting the simulation parameters, together with the corresponding single-run simulation time, are reported in the Supplementary Material Section 2.1.

Using this simulation setup, we evaluated sim-to-real transfer by comparing grasping performance against physical experiments. Two representative objects were considered: a 3D-printed egg-shaped object (maximum radius: 30 mm; height: 65 mm) for power grasping, and a cylindrical object (radius: 16 mm; height: 35 mm) for pinch grasping. The objects weighed 50 g and 10 g, respectively. By systematically increasing the object mass by adding standardized 20 g weights (at a predefined central location integrated into the object design prior to

fabrication) and applying different actuation pressures, we evaluated how closely the simulated grasping success rates align with experimental outcomes. Specifically, grasping was conducted under two actuation pressures, 15 kPa and 13 kPa. For each grasping mode, three weight levels were tested: for power grasping of the egg-shaped object, 7, 6, and 5 units of 20 g were added, while for pinch grasping of the cylinder, 4, 3, and 2 units of 20 g were used. Despite the small sample sizes and the use of binary and coarse success metrics, this sim-to-real validation is intended to demonstrate the feasibility of transferring surrogate-based task-level simulation outcomes to physical grasping behavior, rather than to provide a comprehensive performance benchmark.

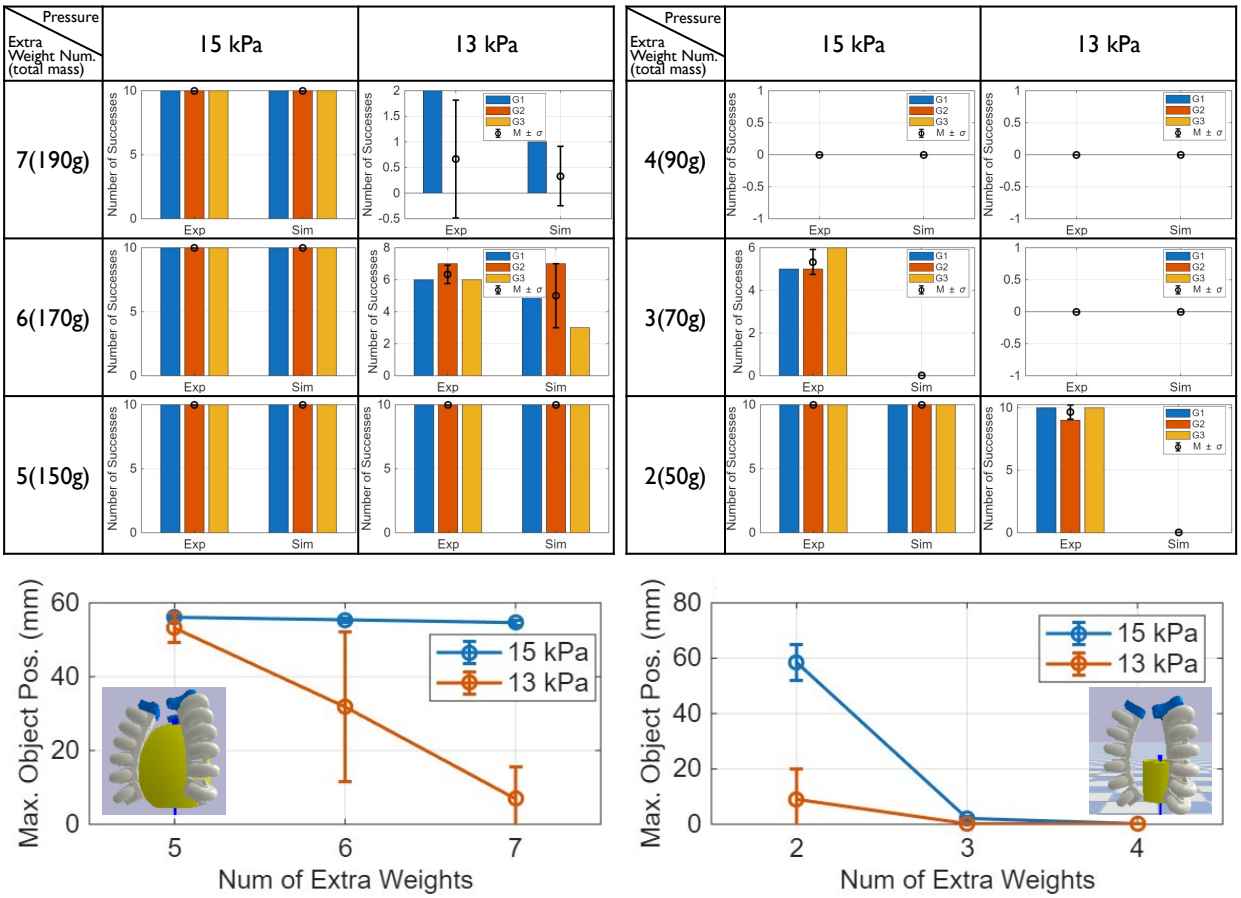


Figure 4: Comparison of experimental and simulated grasping results for the egg (left) and the cylinder (right). The upper tables report the number of successful grasps across different actuation pressure levels and added weight conditions. G1, G2, and G3 denote three independent experimental sets, each consisting of 10 grasping trials conducted under identical conditions. The lower plots show the maximum vertical (z-axis) position of the simulated object relative to the initial placement ($z = 0$), as a function of the applied weight under different actuation pressures.

Physical experiments were conducted using a Franka Emika® Panda arm (Franka Robotics, Germany) with the soft gripper mounted as the end effector and controlled through its standard interface. The actuation pressure was supplied by an Anest Iwata® air compressor (Anest Iwata Corporation, Japan) and regulated by a VPPE-3-1-1/8-2-010-E1 Festo® proportional-pressure regulator (Festo AG & Co. KG, Germany). At each combination of weight level and pressure setting, three sets of ten grasping trials were performed, yielding three success rates (out of 10) for each object under each condition. During each trial, the pneumatic actuation was applied over approximately 2–3 seconds for the grasp attempt, allowing the actuator to establish contact with the object and apply grasping force. Representative examples from both simulations and experiments are provided in the Supplementary Video.

The success rate comparisons are shown in the upper tables of Figure 4. Overall, the simulation tends to underestimate the actual grasping performance, which is expected as certain inter-modular effects are not captured by the model. Nevertheless, the simulated trends are in good agreement with the experimental results.

Additionally, the lower plots in Figure 4 show that, although the success rates are underestimated in simulation,

the simulated maximum z-position $z_{\max}$ of the object correlates with the experimental performance. A higher simulated $z_{\max}$ is associated with a higher success rate in physical experiments, suggesting it can serve as a useful performance indicator.

4.1.2 Continuous Deformation with Small NN Surrogates

To evaluate the capability of the surrogate models constructed with small NNs in task-level simulation, and to further validate the proposed pipeline on a representative class of continuous deformation tasks distinct from contact-dominated grasping, we simulated an elephant-trunk-inspired helical soft actuator, adapted from prior work and incorporating a central thread to constrain elongation [25], under gravity and externally applied tip loads.

As shown in Figure 5a, FEM simulations were performed to characterize the deformation of a half-module under two conditions: (i) free-loading pressurization with increasing internal pressure, and (ii) constant pressure combined with externally applied moments about each principal axis. When moments were applied around the non-active axes (z and x), the rotation about the y-axis was first reversed to isolate angular deflections θ_z and θ_x. In this case, the internal pressure is sampled from 0 kPa to 15 kPa with an interval of 0.5 kPa. The externally applied moments are sampled independently along each axis, with M_y ranging from 0 to 0.8 $\mathrm{N \cdot m}$ at intervals of 0.05 $\mathrm{N \cdot m}$, and M_x and M_z ranging from 0 to 0.006 $\mathrm{N \cdot m}$ at intervals of 0.0002 $\mathrm{N \cdot m}$. Following this axis-wise loading and deformation extraction procedure, three NNs were trained to capture the mappings between pressure, moment, and angular displacement about each axis, each achieving $R^2 > 0.999$. This axis-wise modeling choice constitutes a simplification that reduces learning complexity and improves training stability. Cross-axis coupling effects may become more pronounced under larger deformations or combined multi-axis loading.

To validate the surrogate model, we simulated actuator deformation in PyBullet and compared the results with real-world experiments under self-weight (Figure 5b) and additional tip loads of 10g, 20g, and 50g (Figure 5c-e). In PyBullet, the actuator was modeled as a chain of links connected by spherical joints, with geometry and mass properties matched to the corresponding SPA modules. The tip load was simulated as a downward external force applied to the terminal link. Details of the ablation study used for selecting the helical actuator simulation parameters, as well as the associated computational cost per simulation run, are provided in the Supplementary Material Section 2.2. Recordings of the simulations are provided in the Supplementary Video.

Experimental deformation data were recorded using a Motive OptiTrack™ system (NaturalPoint, Inc., USA) with four Flex 3 cameras, tracking the positions of five rings (each comprising three reflective markers) along the actuator, following the procedure in [25]. The actuation pressure was set to 15 kPa and supplied using the same pneumatic setup as in Section 4.1.1. This pressure level was empirically identified as the maximum representative operating pressure for this actuator. As the surrogate is trained on FEM data spanning 0–15 kPa, its predictions are expected to remain accurate within this range as an interpolation problem, with validation performed at the upper operating limit.

Each experiment was repeated three times to compute average ring positions. To ensure quasi-static behavior during these measurements, we waited approximately 20–30 seconds after each actuation step before recording the actuator configuration. For comparison, the actuator base position in PyBullet was aligned with the experimentally measured mean position of the base ring. Under self-weight, the relative errors at successive ring positions were 1.52%, 1.08%, 2.65%, and 2.85% of the actuator length, with a maximum experimental standard deviation of 1.29 mm observed at the end ring. Under additional tip loads, the relative tip errors were 2.97%, 4.25%, and 5.31% for 10g, 20g, and 50g loads, respectively, with a maximum experimental standard deviation of 1.33 mm at 20g.

4.1.3 A Tendon-Driven Finger Simulation

To demonstrate the applicability of the proposed modeling approach beyond bellow actuators and pneumatic actuation, we also applied the NN surrogate model to a tendon-driven actuator from prior work [28], simulating its deformation under free loading and tip loads of 20g and 50g with the same tendon travel.

As shown in Figure 6a, FEM simulations were conducted to characterize the deformation θ of an actuator module (as indicated by the dashed box in Figure 6e) under free loading and externally applied moments M_{ext}, with the tendon force modeled as a directional load following the tendon path. In this case, the tendon force is sampled from 0 N to 8 N with an interval of 0.05 N. The externally applied moment is sampled from 0 to 0.05 $\mathrm{N \cdot m}$ at intervals of 0.0025 $\mathrm{N \cdot m}$. A NN was trained to capture the mapping between tendon force, external moment, and angular deflection, achieving $R^2 > 0.999$. In addition, a simple regression was used to map tendon force to tendon travel.

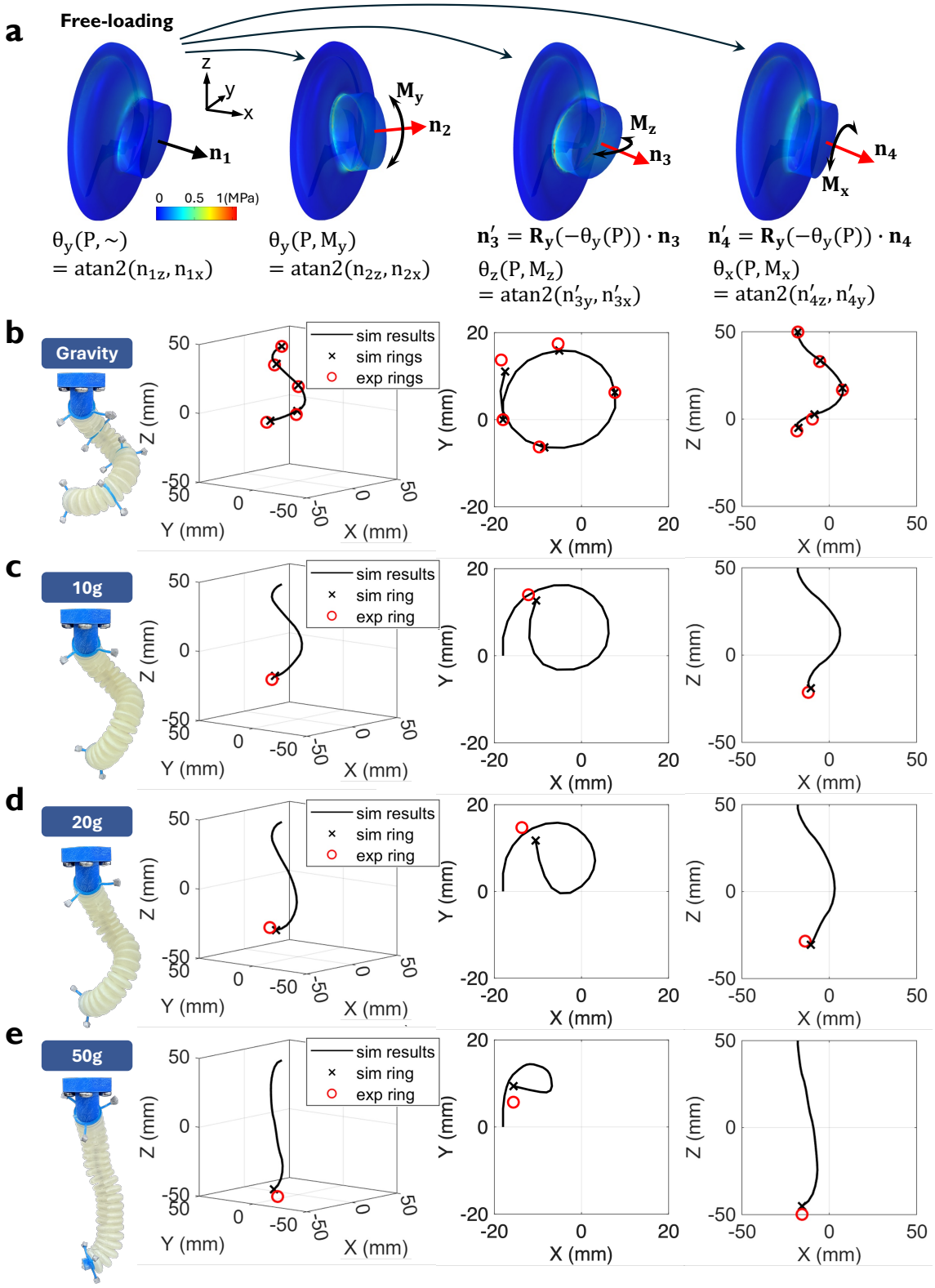


Figure 5: Validation of the surrogate model based on small NNs using a helical actuator under gravity and various tip loads. (a) FEM model of the bellow-SPA used in the helical actuator for all three axes. (b) Comparison of the actuator deformation between simulation and averaged physical experiments over three trials under gravity at five ring positions. (c–e) Comparison of the actuator deformation between simulation and averaged physical experiments over three trials under tip loads of 10, 20, and 50 g applied at the actuator tip.

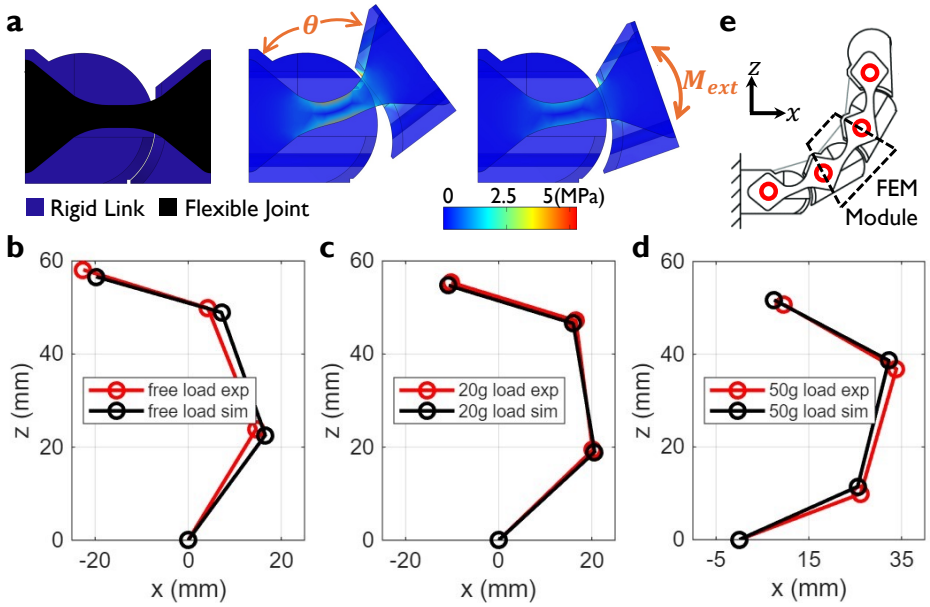


Figure 6: Validation of the surrogate modeling approach beyond the bellow-SPA. (a) FEM model of the joint in the tendon-driven finger developed by [28]. (b–d) Comparison of link positions between simulation and averaged experimental results under free load, 20 g, and 50 g tip loads, respectively. Circle markers correspond to the center positions of each rigid link, with the base link center used as the origin of the reference frame, as indicated in (e), adapted with permission from [28]. Copyright © 2025 IEEE.

To validate the sim-to-real transfer, the tendon-driven actuator was modeled in PyBullet as a chain of links connected by revolute joints, with geometry and mass properties taken from the CAD model. The tip load was simulated as an external force applied to the terminal link and directed toward the pulley position used in the real experiments, which corresponds to the finger's initial tip position. Simulation results were obtained by fine-tuning the tendon force to reproduce the tendon travel reported in [28]. The PyBullet simulation parameters for this tendon-driven actuator were chosen to be identical to those used in the helical actuator experiments, as no object contact was involved. Recordings of the simulations are provided in the Supplementary Video.

The simulation results show good agreement with experimental measurements reported in [28], where each loading condition was experimentally repeated three times, with a maximum standard deviation of 3.51 mm at the end link position under free load. As illustrated in Figure 6b-d, the resulting tip errors were 3.14 mm under free loading, 0.84 mm under a 20 g tip load, and 2.29 mm under a 50 g tip load.

4.2 Surrogate Models for Task-Driven Designs

In this section, we evaluated the effectiveness of the proposed modeling pipeline for task-driven design by coupling the meta-models with SOTA optimization algorithms. Two representative design studies are considered: (i) soft gripper co-design using reinforcement learning, and (ii) 3D actuator shape matching using evolutionary optimization. These two cases employ different meta-model instantiations and optimization strategies, thereby demonstrating that the proposed pipeline can flexibly incorporate task-specific surrogate representations and remain fully compatible with modern learning-based and gradient-free optimization methods.

4.2.1 Designing Soft Gripper with RL

The first case study employs the polynomial-based meta-model in a RL-driven soft gripper co-design setting, as illustrated in Figure 7. Despite its simplified structure, the polynomial surrogate has been shown to provide sufficient accuracy and smoothness for the considered bellow-SPA-based soft gripper within the investigated operating range (Section 4.1.1), while maintaining the computational efficiency required for RL-based design exploration. Here, both morphological design parameters and actuation-related variables are optimized jointly through task-level grasping performance, with the meta-model providing fast predictions of module-level mechanical responses during

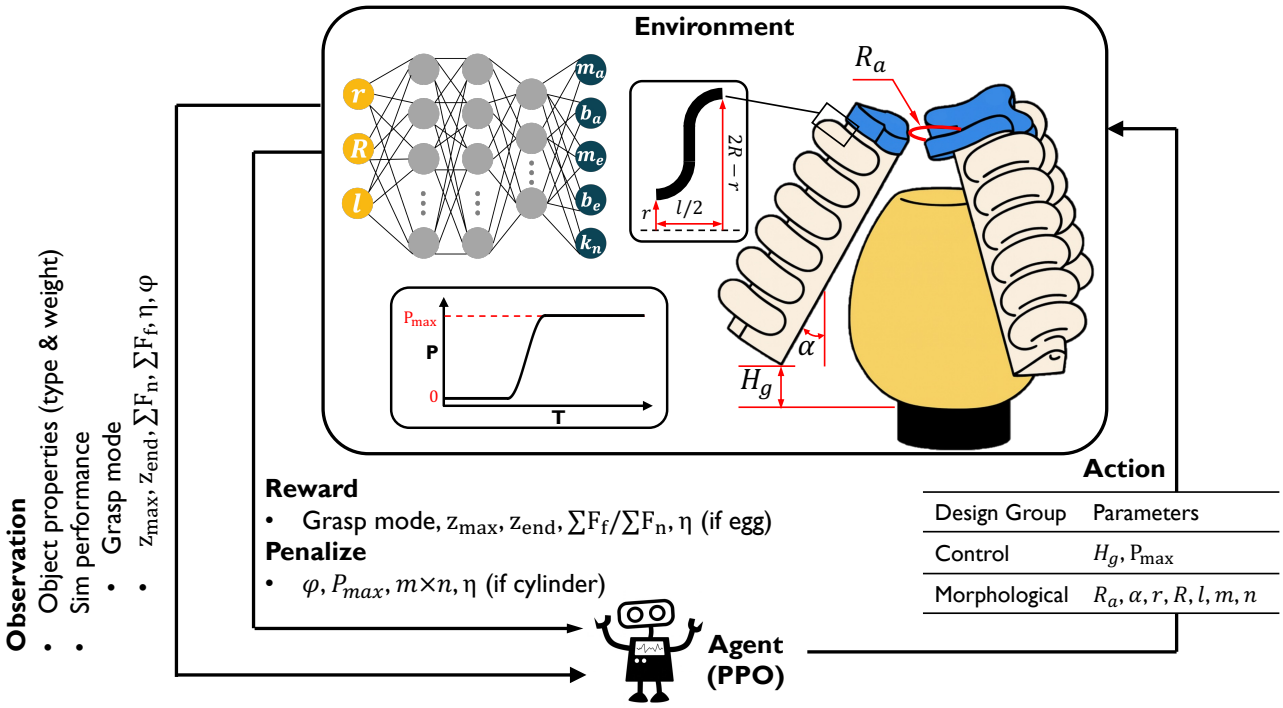


Figure 7: Schematic overview of task-driven soft gripper design using RL. An RL agent (PPO) is trained on a task distribution consisting of grasping an egg and a cylinder with varying weights. The environment corresponds to a simulated grasping trial with randomized object type and weight, where both control and morphological parameters are sampled from the action space and evaluated using the meta-model for polynomial surrogates. The agent observes task properties and simulated grasping performance features, and assigns rewards and penalties accordingly to learn a policy that maps object-specific conditions to optimal gripper design parameters.

simulation.

We first constructed the meta-model from a family of polynomial-based surrogates. The same polynomial surrogate modeling procedure as described in Section 4.1.1 was applied repeatedly over a discretized design space defined by the actuator inner radius $r \in \{2.75, 3.5, 4.25, 5\}$ mm, wall thickness $t = 1.5$ mm, average radius $R \in \{6.75, 7.5, 8.25, 9\}$ mm, and module length $l \in \{9, 10, 11, 12\}$ mm. This design space is chosen as a local neighborhood around the baseline gripper design used in Section 4.1.1, enabling task-driven optimization within a practically relevant actuator configuration range while avoiding unrealistic actuator geometries that would violate the manufacturability and geometric constraints of the bellow-SPA design. This resulted in 54 valid actuator designs, with the remaining combinations excluded by the geometric boundary conditions imposed on the design parameters [26].

A meta-model was then trained as described in Section 3.4, using a three-dimensional input vector composed of the design parameters (r, R, l). The network comprises three hidden layers, two with 64 neurons and one with 32 neurons, each using ReLU activation, followed by a linear output layer predicting the five polynomial coefficients (m_a, b_a, m_e, b_e, k_n), as defined in the polynomial surrogate formulation detailed in the Supplementary Material Section 1.3, Equation S7). The model was trained with the Adam optimizer to minimize the mean squared error (MSE) for up to 1000 iterations and achieves an R^2-score exceeding 0.99 on the full dataset.

This meta-model was embedded into the PyBullet simulation to replace explicit per-design polynomial surrogates, enabling prediction of the actuator's grasping response for continuously varying design parameters. The actuator geometries corresponding to different design parameters are generated by scaling the baseline STL mesh used in Section 4.1.1. The task-level simulation takes three groups of inputs. The control domain contains the grasping height H_g (distance between the actuator tip and the object bottom) and the maximum pressure $P_{\max}$. The morphological domain includes the actuator base radius R_a (radius of the circle formed by the actuator bases), inclination angle α (angle between the actuator longitudinal axis and gravity), the geometric parameters (r, R, l), the module number n (modules per actuator) and the actuator number m (number of actuators). The object properties comprise the object type ("egg" or "cylinder") and the weight number (the number of attached 20 g

weights). All other simulation settings follow those in Section 4.1.1.

The simulator outputs a set of task-relevant features: the maximum and final object positions along the vertical axis $z_{\max}$, z_{end}; the summed normal and friction contact forces over the grasping stage $\sum F_n$, $\sum F_f$; the contact module efficiency η (ratio between the number of active contact modules at the end of grasping and the total module count); the slip ratio φ (ratio between the average object velocity and gripper velocity along the z-axis); and the grasping mode (miss, touch, slip, or stable grasp) determined from object motion and contact forces.

On top of this simulation, we constructed a RL environment for task-driven gripper design. The action space contains all control and morphological design variables listed above, while the observation space includes the object properties together with the simulated grasping performance features. The reward function encourages successful grasping modes, higher lifting height, and larger friction-to-normal force ratios, while penalizing excessive pressure and material usage. It further promotes distributed multi-module contacts when grasping the egg and more localized contacts when grasping the cylinder.

Task-driven design was then performed using a Proximal Policy Optimization (PPO) agent trained on a task distribution consisting of grasping an egg and a cylinder with varying weights. Each episode corresponds to a single simulated grasp attempt, with the object type and weight randomized at reset, so that the learned policy maps task-dependent object conditions to both control commands and gripper morphology. Training was run for 14400 total timesteps in a vectorized environment with four parallel simulations, using the hyperparameters detailed in Supplementary Material Section 2.3. The associated learning curves, shown in Figure S8, indicate stable convergence of the policy and value function.

The resulting distributions of the optimized parameters are summarized in Figure S6 and S7, obtained by evaluating the trained policy on each object type–weight combination and sampling the predicted action parameters over 10 stochastic rollouts per task condition. Most morphological design variables, including inner radius, average radius, module length, module number, and actuator number, converge to nearly identical values across all tasks, whereas the actuator base radius exhibits a clear task-dependent differentiation between the egg and cylinder. This suggests that, under the current reward and task distribution, the policy predominantly converges to a robust general-purpose gripper morphology, with a small subset of parameters adapting to object geometry.

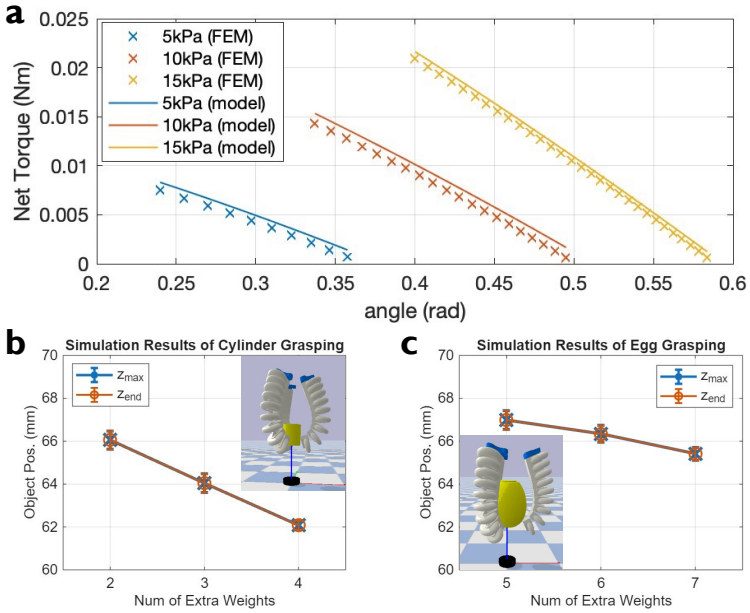


Figure 8: Results of the RL-optimized design. (a) Comparison of net torque versus angular deflection under representative pressure levels between the polynomial surrogate model for the optimized design and its FEM simulation, showing close agreement consistent with the validation in Section 4.1.1. (b) Simulated object grasping results using the polynomial surrogate model for the optimized design, indicating successful and stable grasping under the intended task conditions.

To further verify the quality of the learned design and to remove potential uncertainty introduced by the meta-model, we re-computed the polynomial surrogate for the final optimized design parameters using direct FEM data.

Notably, this design does not belong to the original meta-model training set. The reconstructed surrogate achieved a root-mean-squared error (RMSE) of 3.21% relative to the maximum net torque value when compared with FEM data (Figure 8a). Using this refined surrogate, grasping simulations (10 attempts with added environmental noise) were carried out for all task conditions. The simulation protocol follows the same procedure used in Section 4.1.1 for the sim-to-real validation of the baseline gripper. As shown in Figure 8b and c, the simulated maximum z-position $z_{\max}$ and final z-position z_{end} of the object are nearly identical when grasped by the optimized gripper, indicating stable grasps and corresponding to a success rate of 10/10 in all trials. A quantitative comparison with the baseline gripper configuration reported in Section 4.1.1 is provided in the Supplementary Table S1, which summarizes the grasp success rates and the simulated maximum object position $z_{\max}$ across the tested task conditions. These results confirm that the proposed pipeline can exploit RL together with the polynomial-based meta-model to support task-driven soft gripper design optimization.

4.2.2 3D Shape-Matching with CMA-ES

The second case study uses the NN-based meta-model within a representative evolutionary optimization method for actuator shape-matching design, as shown in Figure 9. The meta-model predicts the deformation behavior of candidate designs, while a gradient-free optimizer updates the design parameters according to a geometric matching objective. In this case study, the shape-matching task is defined using the same right-handed helical target shape as in [25], with a radius of 18 mm and a height of 60 mm.

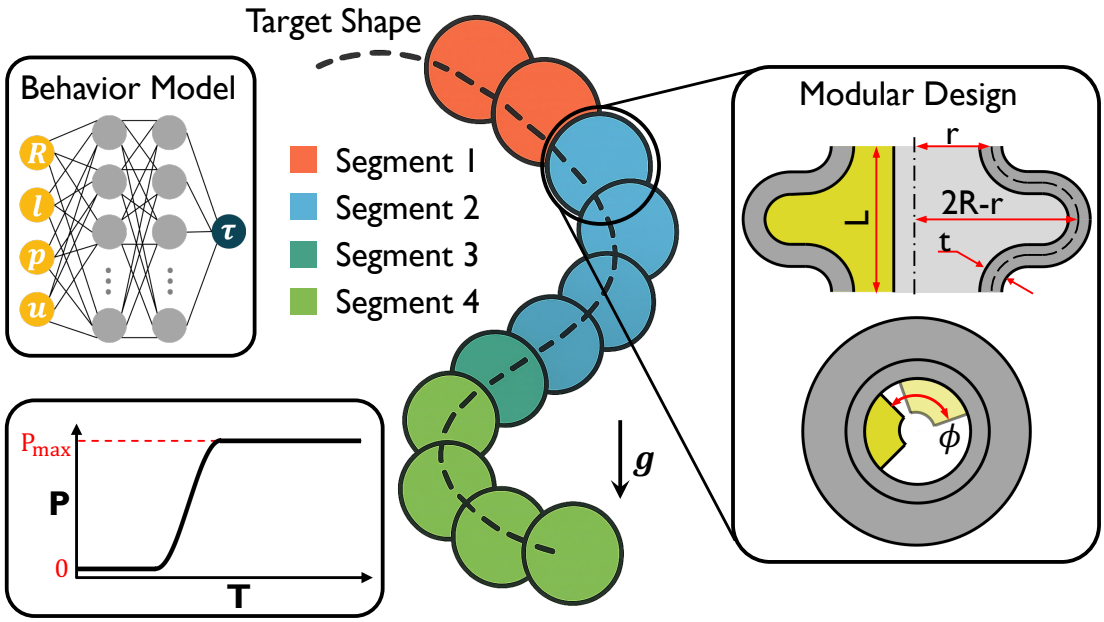


Figure 9: Schematic overview of the shape-matching actuator design task. The actuator is composed of multiple segments (highlighted in different colors), each characterized by the number of identical modules, the geometric design parameters of the module, and the rotation angle relative to the preceding segment. Behavior models that map the modular design and actuation inputs to the output response are employed in simulation to obtain the deformed shape of the designed actuator under actuation pressure and gravity, in order to match a given target shape.

We first applied the NN surrogate model to the same actuator design for deformation prediction along the three principal axes as described in Section 4.1.2. This procedure was repeated over a discretized design space defined by the average radius $R \in \{5, 5.5, 6, 6.5, 7\}$ mm and module length $l \in \{4, 4.5, 5, 5.5, 6\}$ mm, while keeping the inner radius ($r = 3$ mm) and wall thickness ($t = 1.5$ mm) fixed. This design space is centered around the optimal actuator configuration reported in [25], allowing interpolation-based evaluation of the meta-model in the vicinity of a known high-performance design. After excluding combinations that violate the geometric condition $R - r \geq l/4$ [25], 24 valid actuator designs remained.

Following Section 3.4, a behavior model was trained for each axis by aggregating data from all sampled designs.

For a design characterized by (R, l), all tuples (p, u, τ), corresponding to actuation pressure, half joint angle, and resulting joint torque, were collected and stacked into input–output pairs of the form $(R, l, p, u) \rightarrow \tau$. Each axis-wise behavior model was implemented as a feedforward NN with two hidden layers of 64 neurons and ReLU activation, followed by a linear output layer predicting the net joint torque. The models were trained with the Adam optimizer to minimize MSE, achieving R^2 scores exceeding 0.999 for all three axes.

These design-conditioned behavior models were then embedded into the PyBullet simulation environment to replace the explicit small NN models, enabling fast prediction of actuator deformation for varying design parameters. The simulation takes two groups of inputs. The first defines the actuator design: the number of segments N; the average radius R and module length l shared by all modules within each segment; the number of modules per segment n; the relative rotation angle between adjacent segments ϕ, and the actuation pressure $P_{\max}$. The second group specifies the target shape to be matched.

For each candidate design and pressure, the PyBullet simulation produces the deformed centerline of the actuator, reconstructed as a piecewise arc-length-parameterized curve based on link positions and joint rotations. The target helix is re-sampled according to the normalized cumulative arc length of this curve to ensure one-to-one correspondence between simulated and target points. Shape-matching quality is quantified using the RMSE and the maximum point-wise Euclidean distance $e_{\max}$ between corresponding points, and these metrics serve as objective functions for design optimization. In [25], no satisfactory actuator could be identified due to limitations of the underlying model, with the best design yielding an RMSE of 25.68 mm and an $e_{\max}$ of 36.20 mm (Figure 10a).

The optimization problem is formulated as a joint search over structural parameters, geometric design parameters, and actuation pressure. At the structural level, the actuator is partitioned into $N = 4$ segments in this study. The total number of modules is first bounded using a physically motivated estimate based on the target curve length and the range of module lengths, and then five candidate totals are sampled around the center of this interval. For each total, multiple structural configurations $n_i{}_{i=1}^{N}$ are generated by distributing modules across segments under a balance constraint, avoiding extreme structural imbalance while preserving diversity. This procedure yields 81 structural candidates.

For each structural candidate, the continuous design parameters $(R_i, l_i, \phi_i)_{i=1}^{N}$ and actuation pressure p are optimized jointly. To remove global rotational redundancy, the first inter-segment rotation is fixed to $\phi_1 = 0$, while the remaining ϕ_i are treated as optimization variables. Continuous optimization is carried out using the Covariance Matrix Adaptation Evolution Strategy (CMA-ES), which is well suited to strongly nonlinear, non-convex objective landscapes and black-box settings where each function evaluation involves a full physics-based simulation with embedded surrogate models. The initial step size is set to $\sigma_0 = 0.3$ to balance global exploration and local refinement.

During the shape-matching process, CMA-ES iteratively searches the continuous design space for each of the 81 structural configurations, minimizing both RMSE and $e_{\max}$ until either convergence below 5 mm for both metrics or 60 iterations are reached. The resulting optimal actuator design and pressure are reported in Supplementary Material Section 2.4, and summarized in Table S2. Based on the behavior model prediction, the optimized design achieved an RMSE of 4.51 mm and an $e_{\max}$ of 8.46 mm with respect to the target helix.

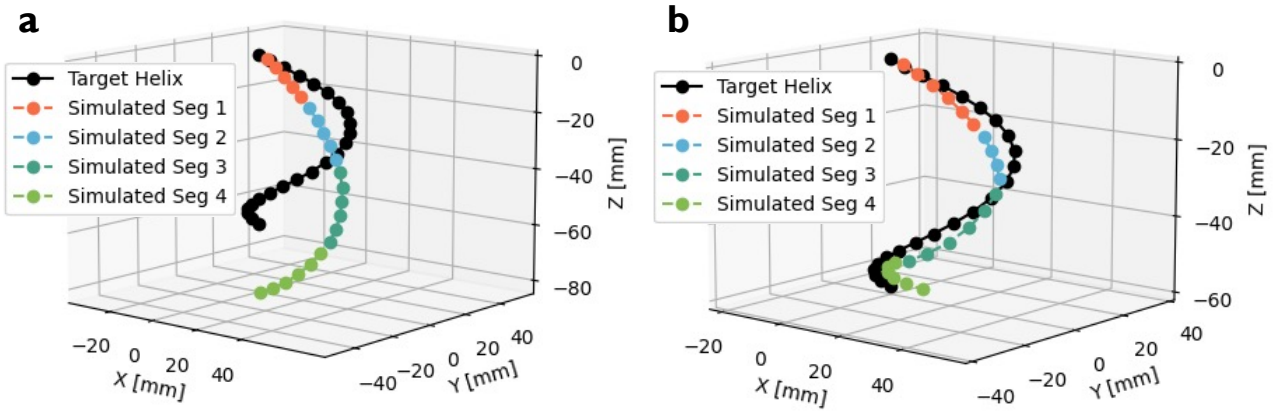


Figure 10: Results of the shape-matching design optimization. (a) Comparison between the target helix and the optimized actuator obtained using the previous modeling approach [25]. (b) Comparison between the target helix and the optimized actuator obtained using the proposed behavior models.

To further evaluate the fidelity of the behavior model prediction, the small NN surrogate models for each segment were subsequently reconstructed using direct FEM data. All reconstructed surrogates achieved R^2 scores above 0.999. The corresponding actuator was then simulated again, yielding an RMSE of 5.05 mm and $e_{\max} = 8.19\ mm$ with respect to the target helix (Figure 10b). Compared with the previous best result, this represents a substantial improvement in shape matching and confirms that the NN-based behavior models, when combined with a representative evolutionary optimizer such as CMA-ES, are effective for task-driven design within the proposed pipeline.

4.3 General Discussion and Limitations

Building on the results presented above, this subsection provides a brief general discussion of the limitations underlying the proposed approach.

First, the proposed pipeline primarily supports interpolation within parameterized actuator design spaces that are explicitly sampled during surrogate construction, and does not aim to generalize across fundamentally different morphologies or actuation principles. Second, the surrogate models were constructed from quasi-static FEM data, which neglect dynamic effects such as rate-dependent material behavior and transient interactions. Third, while extensive validation was performed mainly within the sampled ranges, systematic evaluation outside these ranges remains limited. Finally, hardware validation was demonstrated for selected representative cases, but not for all actuator types and task scenarios considered in simulation.

5 Conclusion

This paper introduced a unified FEM-based surrogate modeling pipeline for generalized task-driven design of soft robots. By integrating high-fidelity FEM data generation, compact surrogate modeling, design-conditioned meta-model learning, and PRBM-based task-level simulation, the proposed pipeline bridges the long-standing gap between physical accuracy and computational efficiency in soft robotic design. Through extensive sim-to-real validation across multiple actuator types and two representative task-driven design studies, namely RL-based soft gripper co-design and evolutionary shape-matching optimization, we demonstrated that the pipeline supports both reliable physical prediction and efficient task-level optimization under realistic environmental interactions. Beyond the specific tasks investigated in this work, the modular structure of the proposed pipeline facilitates extension to additional soft robotic applications within similar modeling assumptions. In particular, the reliance on physically grounded FEM data and surrogate abstraction makes the pipeline well suited for environments involving complex interactions, such as underwater manipulation, fluid–structure coupling, and confined-space operations. Moreover, the decoupling between surrogate modeling and task-level optimization allows the proposed approach to be readily combined with diverse learning-based and gradient-free optimization paradigms. These properties position the pipeline as a fundamental building block toward scalable, autonomous, and general-purpose task-driven design of soft robotic systems in complex real-world environments.

References

[1] D. Rus, M. T. Tolley, *Nature* **2015**, *521*, 7553 467.

[2] C. Lee, M. Kim, Y. J. Kim, N. Hong, S. Ryu, H. J. Kim, S. Kim, *International Journal of Control, Automation and Systems* **2017**, *15*, 1 3.

[3] J. Shintake, V. Cacucciolo, D. Floreano, H. Shea, *Advanced Materials* **2018**, *30*, 29 1707035.

[4] J. Pinskier, D. Howard, *Advanced Intelligent Systems* **2022**, *4*, 1 2100086.

[5] F. Stella, J. Hughes, *Frontiers in Robotics and AI* **2023**, *Volume 9 - 2022*.

[6] M. Pozzi, E. Miguel, R. Deimel, M. Malvezzi, B. Bickel, O. Brock, D. Prattichizzo, In *2018 IEEE International Conference on Robotics and Automation (ICRA)*. **2018** 4206–4213.

[7] M. S. Xavier, A. J. Fleming, Y. K. Yong, *Advanced Intelligent Systems* **2021**, *3*, 2 2000187.

[8] C. Schaff, A. Sedal, S. Ni, M. R. Walter, *Autonomous Robots* **2023**, *47*, 8 1195–1211.

[9] T. Navez, E. Ménager, P. Chaillou, O. Goury, A. Kruszewski, C. Duriez, *IEEE Transactions on Robotics* **2025**, *41* 2441.

[10] M. S. Xavier, C. D. Tawk, A. Zolfagharian, J. Pinskier, D. Howard, T. Young, J. Lai, S. M. Harrison, Y. K. Yong, M. Bodaghi, A. J. Fleming, *IEEE Access* **2022**, *10* 59442.

[11] C. Alessi, C. Agabiti, D. Caradonna, C. Laschi, F. Renda, E. Falotico, Rod models in continuum and soft robot control: a review, **2024**, URL `https://arxiv.org/abs/2407.05886`.

[12] D. Bruder, M. A. Graule, C. B. Teeple, R. J. Wood, *Science Robotics* **2023**, *8*, 81 eadf9001.

[13] R. J. WebsterIII, B. A. Jones, *The International Journal of Robotics Research* **2010**, *29*, 13 1661.

[14] H. Zhang, K. H. Heung, G. J. Naquila, A. Hingwe, A. D. Deshpande, *Advanced Intelligent Systems* **2025**, *7*, 3 2400637.

[15] C. Della Santina, R. K. Katzschmann, A. Biechi, D. Rus, In *2018 IEEE International Conference on Soft Robotics (RoboSoft)*. **2018** 46–53.

[16] M. A. Graule, C. B. Teeple, T. P. McCarthy, G. R. Kim, R. C. St. Louis, R. J. Wood, In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. **2021** 3934–3941.

[17] P. Moseley, J. M. Florez, H. A. Sonar, G. Agarwal, W. Curtin, J. Paik, *Advanced Engineering Materials* **2016**, *18*, 6 978.

[18] P. Ferrentino, E. Roels, J. Brancart, S. Terryn, G. Van Assche, B. Vanderborght, *IEEE Robotics & Automation Magazine* **2024**, *31*, 3 97.

[19] M. Macklin, M. Müller, N. Chentanez, In *Proceedings of the 9th International Conference on Motion in Games*, MIG '16. Association for Computing Machinery, New York, NY, USA, ISBN 9781450345927, **2016** 49–54, URL `https://doi.org/10.1145/2994258.2994272`.

[20] Y. Hu, Y. Fang, Z. Ge, Z. Qu, Y. Zhu, A. Pradhana, C. Jiang, *ACM Trans. Graph.* **2018**, *37*, 4.

[21] M. Wiese, B.-H. Cao, A. Raatz, In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. **2022** 9371–9378.

[22] O. Goury, C. Duriez, *IEEE Transactions on Robotics* **2018**, *34*, 6 1565.

[23] A. T. Mathew, I. B. Hmida, C. Armanini, F. Boyer, F. Renda, *IEEE Robotics & Automation Magazine* **2023**, *30*, 3 106.

[24] J. Liu, J. H. Low, Q. Q. Han, M. Lim, D. Lu, C.-H. Yeow, Z. Liu, *IEEE Robotics and Automation Letters* **2022**, *7*, 2 5803.

[25] Y. Yao, L. He, P. Maiolino, *Robotics Reports* **2024**, *2*, 1 1, pMID: 39263556.

[26] Y. Yao, L. He, P. Maiolino, In *2022 IEEE 5th International Conference on Soft Robotics (RoboSoft)*. **2022** 29–34.

[27] A. A. Shabana, M. A. Abdelrahman, A. M. El-Kassas, H. E. Abdel-Azim, *Journal of the Egyptian Society of Tribology* **2012**, *9*, 4 28, article 3.

[28] E. Capp, M. Pontin, P. Walters, P. Maiolino, In *2025 IEEE 8th International Conference on Soft Robotics (RoboSoft)*. **2025** 1–8.

6.4 Supporting Information

6.4.1 Supporting Material

1 Material and Methods

This section complements Section 3 of the main text by providing the detailed derivations and implementation details. It covers the FEM setup, surrogate model derivations, and illustrative examples, and also provides additional results for polynomial and neural network (NN) surrogates.

1.1 FEM Setup for Data Generation

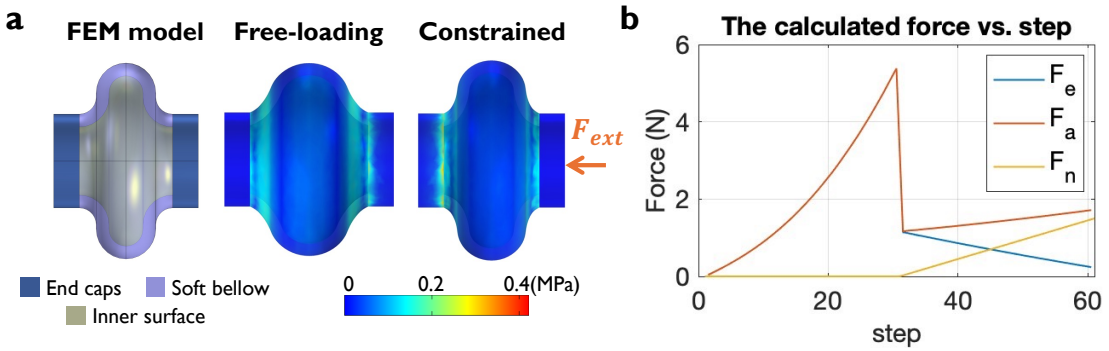


Figure S1: FEM setup and force evolution for a translational bellow-SPA module.(a) The FEM model under free-loading and constrained conditions.(b) Calculated elastic, actuation, and external forces over the simulation steps, illustrating the transition from the free-loading phase to the constrained phase.

To illustrate the FEM procedure described in Section 3.1, we used a modular translational bellow-SPA as an example. The actuator is fabricated with Stratasys digital materials and features a U-shaped bellow geometry driven by pneumatic actuation. All soft pneumatic actuators considered in this work (except for the tendon-driven actuator) are modeled or printed using Agilus30™elastomer (Stratasys Ltd.), and its hyperelastic behavior in the FEM simulations is defined based on the stress–strain characterization reported in [1].

A stationary 3D FEM model, shown in Figure S1a, was constructed in COMSOL Multiphysics® (COMSOL Inc., Sweden) to approximate quasi-static conditions, which are appropriate for low-speed operation. The physical model is based on a prior work [2], using default material and geometric parameters without inner constraints. Gravity is excluded, and symmetry is applied by fixing either one end cap or the central seam.

Two types of simulation sweeps were performed:

- Free-loading condition: Internal pressure P_{in} is applied to the inner surface, increasing from 0 to 15 kPa in 0.5 kPa increments. This produces a displacement x of the free end.
- Constrained condition: Using the final state from the free-loading step as the initial condition, an external force F_{ext} is applied to the unfixed end cap. The force increases from 0 to 1.5 N in 0.05 N increments, counteracting the displacement while keeping P_{in} fixed. The process is repeated for multiple fixed pressures.

The resulting dataset consists of paired actuation inputs, displacements, and reaction forces under both free and constrained loading. These data form the basis for constructing compact surrogate models.

1.2 Detailed Derivation of the Polynomial Surrogate

The actuator's joint force τ is defined from the derivative of the potential energy U with respect to deformation u. With FEM results available only at discrete quasi-static states, this derivative can be approximated by finite differences:

$$\tau_i(p_i, u_i) \approx \frac{U_{i+1}(p_{i+1}, u_{i+1}) - U_i(p_i, u_i)}{u_{i+1} - u_i} \tag{S1}$$

where i indexes the FEM step, and (p_i, u_i) are the actuation–deformation pairs.

When the actuator is unconstrained, deformation varies with the applied actuation p. The equilibrium at each state requires that the actuation-induced force balances the elastic restoring force:

$$\tau_{a,f} + \tau_{e,f} = 0 \tag{S2}$$

with the elastic contribution evaluated as

$$\tau_{e,f} = \frac{\delta U_{e,f}}{\delta u} \tag{S3}$$

In the constrained case, the actuation level p is fixed, while an external load τ_n is applied to affect deformation. The corresponding elastic force can be written as

$$\tau_{e,c}(p = P, u) = \tau_{e,f}(p = P, u_f) + \frac{\delta U_{e,c}(p = P, u)}{\delta u} \tag{S4}$$

where u_f denotes the free-loading deformation under the same actuation. The equilibrium relation then becomes

$$\tau_{a,c} + \tau_{e,c} + \tau_n = 0 \tag{S5}$$

By further differentiating the force contributions with respect to u, the effective stiffness can be separated into an elastic part $k_e(u)$ and an actuation-dependent part $k_a(p)$. This leads to

$$\tau_n(p, u) = \tau_n(p, 0) - \int_0^u (k_e(u) + k_a(p))du \tag{S6}$$

with $\tau_n(p, 0)$ determined from free-loading conditions.

Finally, each term $k_a(p)$, $k_e(u)$, and $\tau_n(p, 0)$ is approximated using low-order polynomial fits to FEM data. This produces a closed-form surrogate mapping $(p, u) \to \tau_n$, which is compact and suitable for fast evaluation in rigid-body-based simulators.

1.3 Example: Translational Bellow-SPA

To illustrate the surrogate construction and validate its viability, we applied the derivation to a translational bellow-SPA module. The FEM model of the module is shown in Figure S1a, and the strain energy and axial-displacement data are extracted under both free-loading and constrained conditions. Figure S1b further shows the calculated elastic, actuation, and external forces across simulation steps, in which the elastic force is derived using Eqs. S2 while the actuation force is obtained from the equilibrium conditions in Eqs. S3 and Eqs. S5. These results highlight the transition from free-loading to constrained conditions and motivate the separation of stiffness contributions, which we quantify as follows:

- Actuation stiffness $k_a(p)$: obtained from constrained FEM simulations at fixed pressures (5–15 kPa), then fitted with a first-order polynomial in p. The comparison in Figure S2a–b shows close agreement, with maximum relative error $\approx 7.7\%$.
- Elastic stiffness $k_e(u)$: obtained from constrained data as the slope of force–displacement curves, fitted linearly in u. Figure S2c shows a maximum relative error $\approx 4.1\%$.
- Offset term $F_n(p, 0)$: determined from free-loading FEM simulations (where $F_n(p, u_f) = 0$, fitted linearly in p (Figure S2d, max relative error $\approx 0.1\%$).

Although simulations were performed across the full range of pressures (5–15 kPa), three representative cases (5, 10, and 15 kPa) are shown in Figure S2a&c for clarity, while regression and error analysis were performed over the complete dataset.

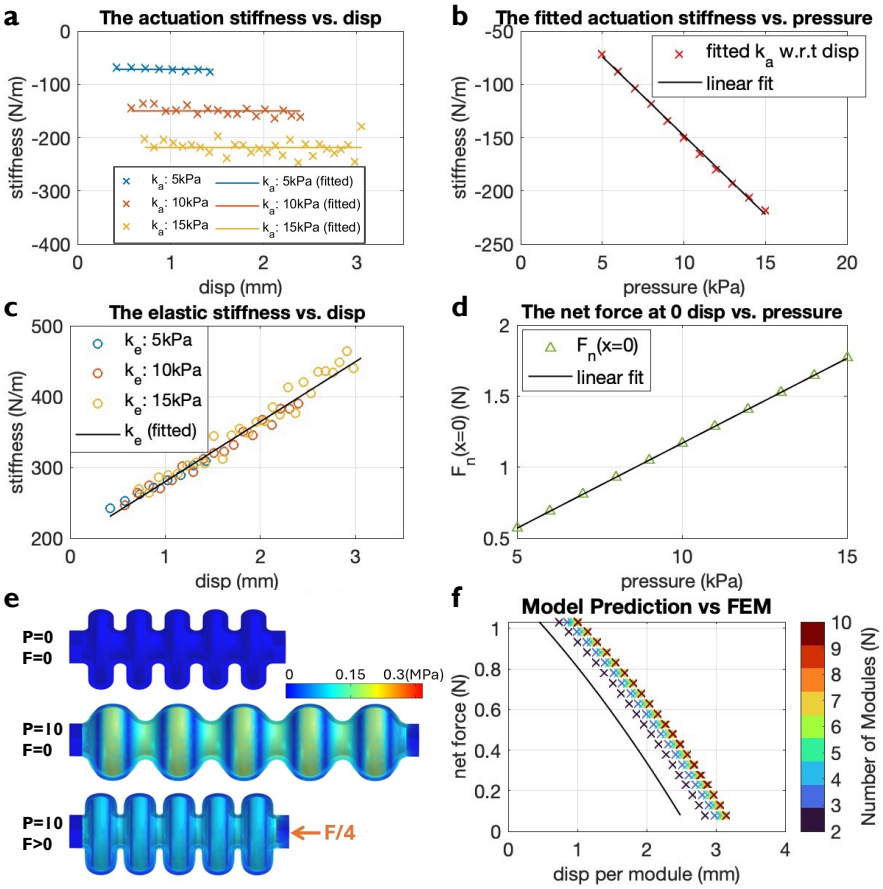


Figure S2: Polynomial-based surrogate modeling process and validation. (a–d) Polynomial fits of the actuation and elastic stiffness across different pressures and displacements. (e) FEM model of connected actuator modules. (f) Comparison between predictions from the polynomial surrogate model and the full FEM simulation for actuators composed of 2–10 modules.

These fits yield the closed-form expression

$$F_n(P,x) = -\int_0^x (m_a P + b_a + m_e x + b_e)dx + k_n P \tag{S7}$$

where coefficients are obtained from regression.

To further validate this formulation, we applied it to multi-module actuators. Figure S2e shows FEM simulations of actuators composed of 2–10 connected modules, each subjected to an internal pressure of 10 kPa. The surrogate model predicts the net force–displacement relationship by treating the full actuator as a series of identical modules, where the displacement scales with the number of modules while the force remains that of a single module.

Figure S2f compares FEM predictions with those from the surrogate model. Under the same applied pressure (10 kPa) and constrained end conditions, the surrogate consistently predicts smaller displacements than FEM for a given net force. The discrepancy increases with the number of modules, from about 16.1% error for 2 modules to 29.8% error for 10 modules, measured relative to the displacement range. Notably, the error growth slows as the number of modules increases, and the predictions converge toward a consistent offset.

This conservative underestimation arises from intermodule effects, such as accumulated compliance and boundary interactions, which are captured in FEM but not explicitly modeled in the surrogate's module-wise composition. Nevertheless, the surrogate accurately preserves the overall trend of the actuator's behavior, offering reliable tendency predictions with a beneficial safety margin for task-driven applications where ensuring feasibility is critical.

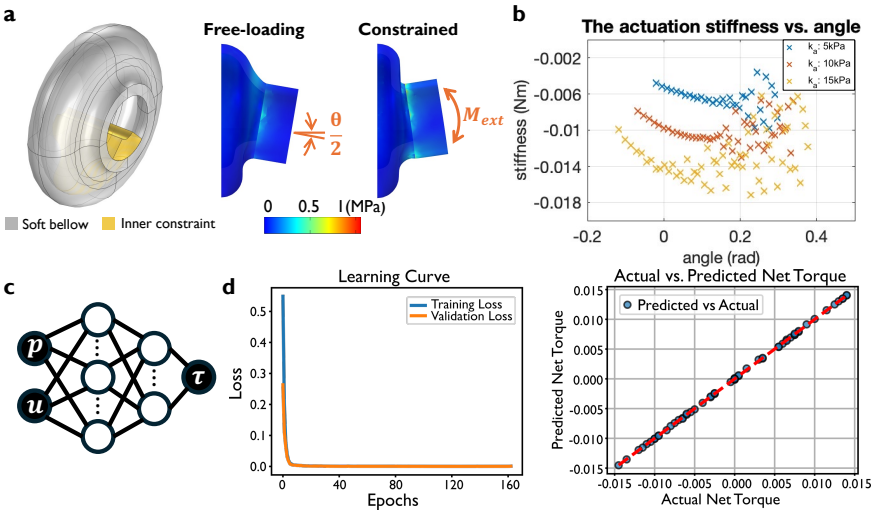


Figure S3: Surrogate modeling using small NNs. (a) FEM model of a bending bellow-SPA with an internal constraint. (b) Calculated actuation stiffness as a function of angular deflection under multiple pressure levels, obtained using the formulation in Section 3.2. (c) Architecture of the small feedforward NN. (d) Training and prediction results of the neural network surrogate.

1.4 Neural Network Surrogate Details

To illustrate the need for NN surrogates, we considered a bending bellow-SPA module with an internal constraint. Following the same modeling process as in Section 3.2, the deformation is defined as an angular deflection θ, and the external force as a moment M_{ext} (Figure S3a). The actuation stiffness of the default design is plotted against the modular angle for multiple pressures in Figure S3b.

The stiffness–angle curves exhibit strong nonlinearities, particularly at larger deflections, which low-order polynomial fits fail to capture. For instance, polynomial regressions of orders 1–3 yield maximum errors between 23.54% and 19.13% relative to the stiffness range, demonstrating the limitations of closed-form approximations in such cases.

To overcome this, we trained a small feed-forward NN to approximate the mapping $(p, \theta) \mapsto M_{ext}(p, \theta)$, as shown in Figure S3c. The network consists of two hidden layers (16 and 8 neurons, ReLU activations) and a linear output, trained using FEM data spanning free-loading and constrained conditions. Data are split into 80% training and 20% validation. The loss is defined as the mean squared error (MSE) between FEM-derived moments and network predictions, optimized with Adam (learning rate 0.005, batch size 16, 400 epochs with early stopping).

The trained network achieved an R^2 score above 0.999 on the validation set, substantially reducing errors compared with polynomial fits (Figure S3d). After training, the model remains sufficiently lightweight to be directly embedded in pseudo-rigid body model (PRBM)-based simulations in PyBullet, thereby enabling task-level simulation with nonlinear actuator behaviors.

2 Results and Discussion

This section complements Section 4 of the main text by providing supporting details on the simulation setup and optimization results.

2.1 Additional information of the Grasping Simulation

To determine appropriate PyBullet simulation parameters for the grasping task, an ablation study was performed on the simulation time step size and the joint damping coefficient, while keeping the actuation pressure, contact friction, and object properties fixed. For each tested parameter combination, a grasping simulation was executed and the dynamic responses of both the actuator and the object were evaluated.

Representative time histories of the actuator joint angles and joint torques are shown in Figure S4 to illustrate typical underdamped and overdamped behaviors. Very small damping values lead to pronounced oscillations in both

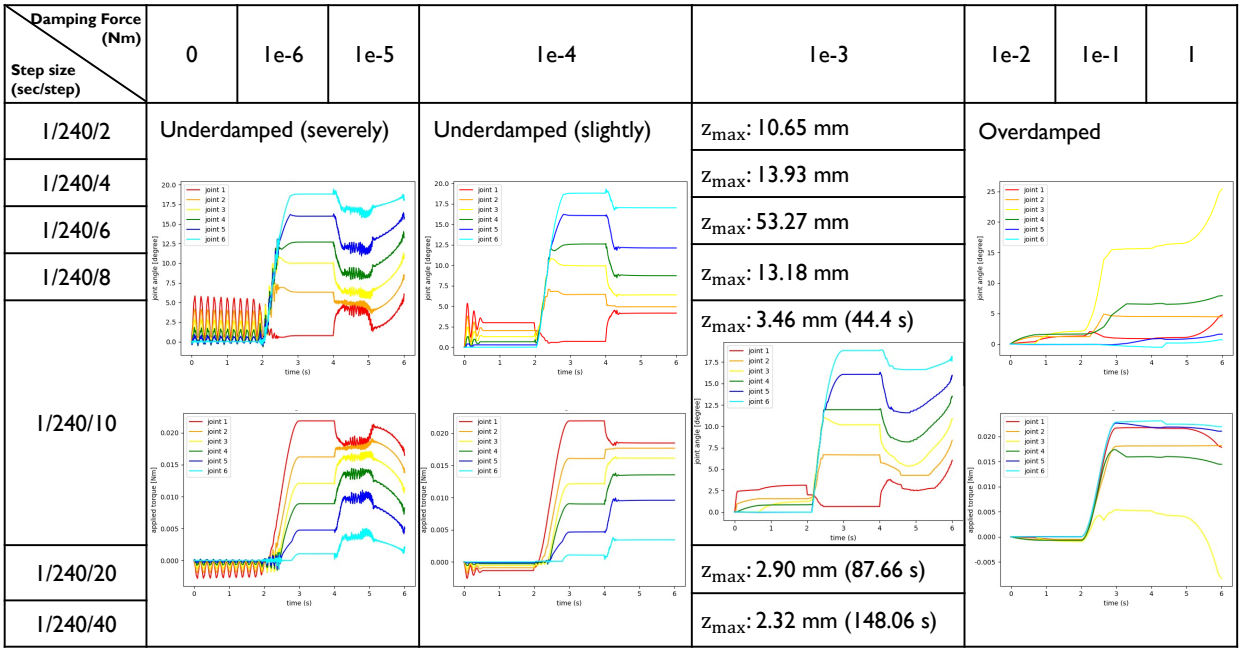


Figure S4: Ablation study of the PyBullet simulation settings for the grasping task. The effects of the simulation time step size and the joint damping coefficient are evaluated under fixed actuation pressure, contact friction, and object properties. Representative actuator joint angle and joint torque responses, together with the maximum vertical position of the object $z_{\max}$, are analyzed.

joint angles and joint torques, indicating underdamped responses, whereas excessively large damping suppresses joint motion and results in overdamped behavior with reduced effective actuation.

In addition, the maximum vertical position of the object, denoted as $z_{\max}$, is used to assess the effect of the simulation time step on the lifting performance. In Figure S4, the corresponding $z_{\max}$ values and wall-clock simulation times are explicitly reported for the cases with a joint damping coefficient of 10^{-3}, thereby highlighting the trade-off between numerical accuracy and computational cost as the time step decreases. For this damping value, $z_{\max}$ increases and tends to converge when the time step reaches 1/2400 and below, while further reducing the time step mainly leads to increased simulation time.

Considering both numerical stability and computational efficiency, a simulation time step of 1/2400 and a joint damping coefficient of 10^{-3} were selected for all subsequent grasping simulations. This setting avoids both underdamped and overdamped behaviour while providing converged lifting performance with a reasonable simulation time.

To quantify the corresponding computational cost, we note that, using an Intel(R) Core(TM) i7-10875H CPU @ 2.30 GHz with 16.0 GB RAM, the execution time of a single simulation run ranged from approximately 10 to 60 s without visualization, depending on the complexity of the contact interactions.

2.2 Additional information of the Continuous Deformation Simulation

To determine suitable PyBullet simulation parameters for the helical actuator, we performed an ablation study on the simulation time step size and the joint damping coefficient. For each parameter combination, the actuator motion was simulated under identical loading conditions, and the actuator tip position (fifth ring) was analyzed after the transient response had settled. The simulated tip position was computed as the mean value over the last half cycle of the motion.

As shown in Figure S5a, the position error between the simulated and experimentally measured actuator tip varies significantly with both the time step size and the damping coefficient. Larger time steps generally lead to increased errors, while excessively small or large damping values also degrade accuracy. Figure S5b reports the standard deviation of the simulated tip position over the last half cycle, which reflects the numerical stability of the simulation. Insufficient damping results in large fluctuations, whereas overly strong damping suppresses motion and introduces bias.

Among all tested combinations, the parameter setting with a time step of 1/2400 and a joint damping coefficient of 10^{-4} achieves both low position error and low standard deviation. This setting therefore provides a favorable balance between simulation accuracy and numerical stability and is adopted for all subsequent helical actuator simulations in this work.

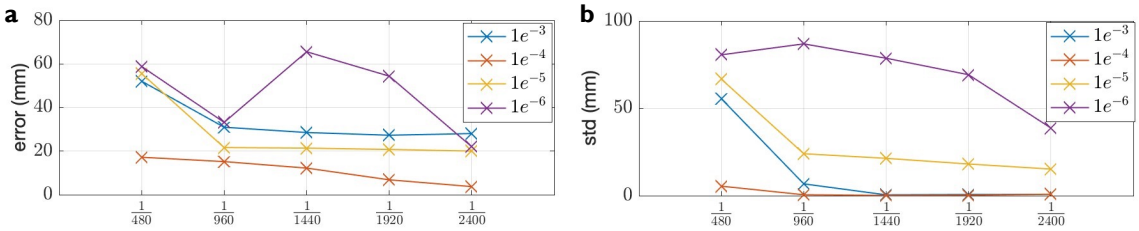


Figure S5: Ablation study of the PyBullet simulation settings for the helical actuator. The effects of the simulation time step size and the joint damping coefficient are evaluated at the fifth ring (actuator tip). (a) Position error between the experimentally measured actuator tip and the simulated tip position, where the simulated position is computed as the mean over the last half cycle after the motion has settled. (b) Standard deviation of the simulated tip position over the last half cycle.

To quantify the associated computational cost under this selected setting, we note that, under the same hardware configuration, the execution time of a single simulation run was approximately 10 s without visualization.

2.3 Details of the Task-Driven Soft Gripper Design with RL

From the learned distributions shown in Figure S6 and S7, the following representative values can be extracted for the optimized grasping strategy. The grasping height consistently converges to 7.5 mm, and the maximum pressure converges to 10 kPa across all tasks. The actuator inclination angle converges to approximately 0.314 rad. The optimized gripper morphology exhibits a strong convergence behavior: the average radius converges to 8 mm, the module length to 10 mm, the module number to 9, and the actuator number to 3.

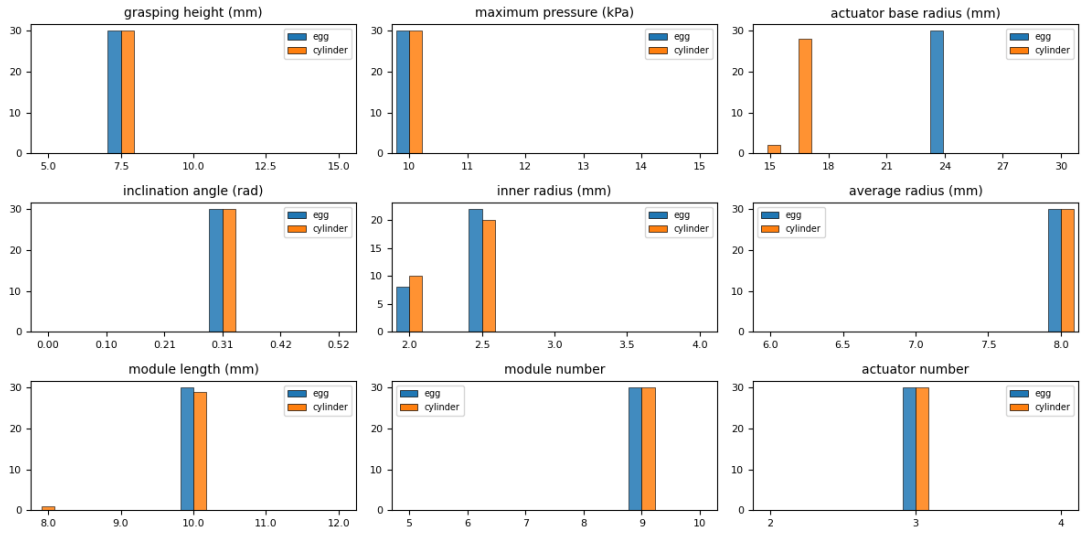


Figure S6: Distributions of the RL-generated design and control parameters for different object types. Histograms show the distributions of the optimized parameters for grasping an egg (black) and a cylinder (orange), including grasping height, maximum pressure, actuator base radius, inclination angle, inner radius, average radius, module length, module number, and actuator number. The y-axis represents the number of evaluation samples, while the x-axis corresponds to the parameter indicated in each subplot title. When two bars appear side by side, they share the same parameter value, defined by the center of the two bars; when only a single bar is present, its center indicates the corresponding parameter value.

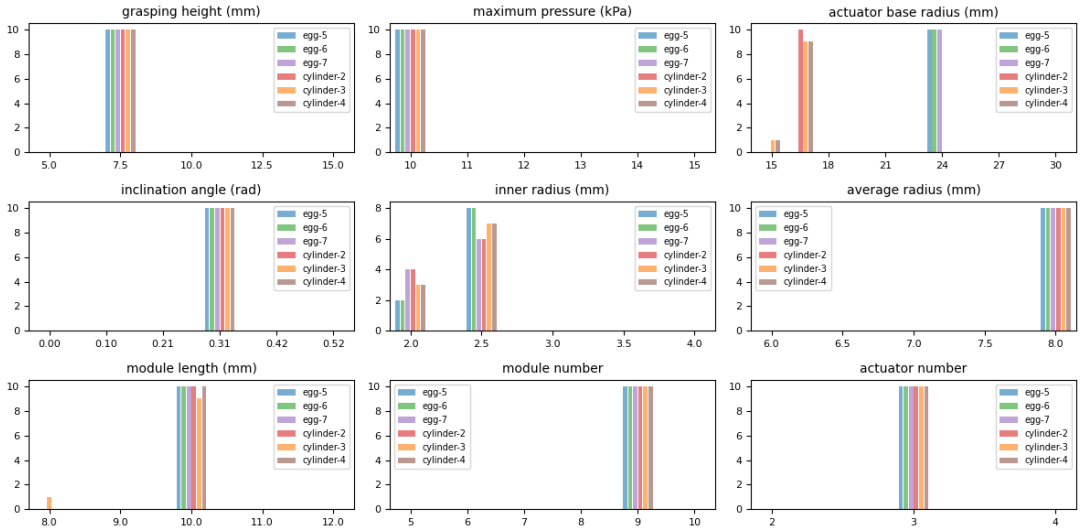


Figure S7: Distributions of the RL-generated design and control parameters across object types and weight conditions. Histograms show the optimized parameter distributions for grasping eggs and cylinders under different weight cases, including grasping height, maximum pressure, actuator base radius, inclination angle, inner radius, average radius, module length, module number, and actuator number. The y-axis represents the number of evaluation samples, while the x-axis corresponds to the parameter indicated in each subplot title. When bars appear side by side, they share the same parameter value, defined by the center of the bars; when only a single bar is present, its center indicates the corresponding parameter value.

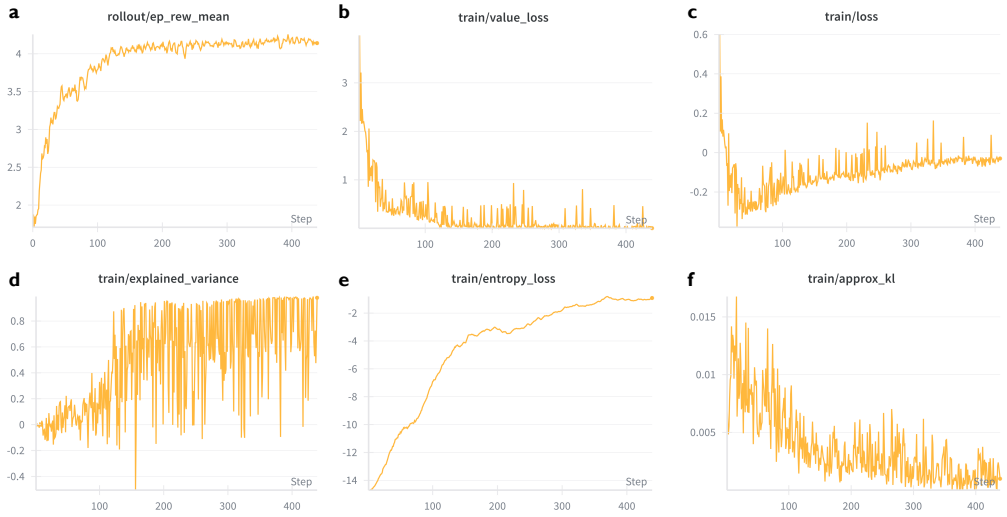


Figure S8: Training performance of the RL agent (PPO). (a) Mean episode reward, (b) value loss, (c) total training loss, (d) explained variance, (e) policy entropy, and (f) approximate Kullback–Leibler (KL) divergence, plotted over the training steps.

A task-dependent differentiation is observed only in the actuator base radius: the optimal value converges to approximately 24 mm for the egg and 17 mm for the cylinder, indicating a clear adaptation of the actuator placement to the object geometry. The inner radius exhibits a bimodal distribution around 2.0 mm and 2.5 mm. Since the distribution at 2.5 mm is more dominant and no clear task-dependent separation is observed, this value is selected as the final optimal design choice.

The task-driven design problem is solved using Proximal Policy Optimization (PPO) implemented with Stable-Baselines3. The agent is trained in a vectorized simulation environment consisting of four parallel environments

(DummyVecEnv), where each environment samples the object type (egg or cylinder) and weight randomly at reset. The policy network is a multilayer perceptron (MLP) operating under the compact observation mode.

The PPO hyperparameters are summarized as follows. The learning rate is set to 1×10^{-4} to improve stability under task randomization. The discount factor is set to $\gamma = 0.9$ since each episode corresponds to a single grasp attempt. The number of rollout steps per update is n_steps = 8, with a batch size of 32. The clipping range is set to 0.3 to allow slightly larger policy updates. To encourage exploration during joint optimization of control and morphology, the entropy coefficient is set to 0.03. Generalized Advantage Estimation is used with $\lambda = 0.9$, and the value function loss coefficient is 0.3. Gradient updates are stabilized using a maximum gradient norm of 0.5, and a target KL-divergence of 0.05 is enforced. Advantage normalization is enabled to further smooth gradient updates.

The training curves are shown in Figure S8. As shown in Figure S8a, the mean episode reward increases rapidly during the early stage of training and gradually converges to a stable plateau, indicating successful policy learning. Figure S8b and Figure S8c show that both the value loss and total training loss decrease substantially and stabilize, demonstrating stable critic learning and policy updates. The explained variance of the value function (Figure S8d) increases and remains consistently high, indicating accurate value function approximation. The policy entropy (Figure S8e) gradually decreases as training progresses, reflecting the transition from exploration to exploitation. Meanwhile, the approximate KL-divergence (Figure S8f) remains well below the target threshold, confirming that PPO updates stay within the desired trust region.

These results confirm that the PPO agent achieves stable and well-converged training behavior under the proposed task-driven design formulation.

Table S1 reports a quantitative comparison between the baseline/default gripper used in Section 4.1.1 and the RL-optimized gripper obtained in Section 4.2.1. The default gripper is evaluated at actuation pressures of 15 kPa and 13 kPa, while the optimized gripper operates at the pressure selected by the RL policy (10 kPa). For each object and weight condition, grasping simulations are repeated in three groups of ten trials. Performance is evaluated using the grasp success rate (out of 10 trials) over three groups and the averaged simulated maximum object lifting position $z_{\max}$ computed across the same trials.

Table S1: Quantitative comparison between the baseline gripper and the RL-optimized gripper, evaluated using the grasp success rate and the averaged simulated maximum object lifting position $z_{\max}$.

Object	Num of Extra Weights (Total Mass)	Default Success (15 kPa)	Default Success (13 kPa)	Optimized Success	Default $z_{\max}$ (15 kPa) [mm]	Default $z_{\max}$ (13 kPa) [mm]	Optimized $z_{\max}$ [mm]
Egg	5 (150 g)	10/10 × 3	10/10 × 3	10/10 × 3	56.10	53.18	66.39
Egg	6 (170 g)	10/10 × 3	5/10, 7/10, 3/10	10/10 × 3	55.41	31.82	65.95
Egg	7 (190 g)	10/10 × 3	1/10, 0/10, 0/10	10/10 × 3	54.65	6.91	65.32
Cylinder	2 (50 g)	10/10 × 3	0/10 × 3	10/10 × 3	58.50	9.02	65.62
Cylinder	3 (70 g)	0/10 × 3	0/10 × 3	10/10 × 3	2.14	0.23	63.43
Cylinder	4 (90 g)	0/10 × 3	0/10 × 3	10/10 × 3	0.22	0.22	61.91

2.4 Details of the 3D Shape-Matching with CMA-ES

In this section, we report the optimal actuator design obtained from the Covariance Matrix Adaptation Evolution Strategy (CMA-ES)–based 3D shape-matching procedure described in Section 4.2.2. The optimized design uses a fixed inner radius $r = 3.0$ mm and wall thickness $t = 1.5$ mm, consistent with the surrogate training dataset. The structure is partitioned into $N = 4$ segments with module counts $n_{i\,i=1}^{4} = [6, 4, 6, 6]$. For each segment i, Table S2 lists the optimized average radius R_i, module length l_i, and inter-segment rotation angle ϕ_i. The corresponding optimal actuation pressure is $P_{\max} = 11.81$ kPa.

Table S2: Optimized actuator design parameters for the 3D helix shape-matching task.

Segment i	n_i	R_i [mm]	l_i [mm]	ϕ_i [deg]
1	6	6.65	4.37	0.0
2	4	5.68	4.82	18.1
3	6	6.22	5.20	40.5
4	6	5.87	4.50	65.3

References

[1] F. F. Abayazid, M. Ghajari, *Additive Manufacturing* **2020**, *33* 101160.

[2] Y. Yao, L. He, P. Maiolino, *Robotics Reports* **2024**, *2*, 1 1, pMID: 39263556.

6.4.2 Supporting Video

Representative simulation and experimental videos demonstrating the proposed task-driven design pipeline across different actuator designs and applications are available here: `https://github.com/ORI-SRL/Task-driven-design-pipeline/blob/main/SuppVideo.mp4`

7
Conclusion

This thesis addresses fundamental challenges in the task-driven design of soft actuators, with a particular focus on SPAs and their use in robotic systems interacting with complex environments. In this concluding chapter, we summarise the major research gaps identified through the literature study, discuss how the individual contributions of this thesis address these gaps, and outline potential future research directions enabled by this work.

As discussed in the literature review (Chapter 2), soft actuators offer significant advantages in safety, adaptability, and compliance, but their behaviour is highly nonlinear and strongly coupled to geometry, materials, actuation, and environmental interaction. As a result, designing soft actuators to meet specific task requirements often involves iterative trial-and-error and remains highly case-specific. Therefore, the overarching vision of this DPhil is to provide an end-to-end task-driven design methodology that supports systematic translation from task requirements to manufacturable soft actuator designs through computational modelling, simulation, and optimisation. After investigating SOTA work in soft actuator design practices, modelling approaches, optimisation methods, and task-driven design pipelines, four major research gaps were identified: (1) limited availability of accurate modelling and simulation tools suitable for iterative soft actuator design; (2) difficulty of inverse design when task complexity leads to large and highly nonlinear design

spaces; (3) lack of systematic understanding of optimisation trade-offs in inverse soft actuator design; and (4) limited generalisation and reuse of task-driven design pipelines across actuator designs and task scenarios. To address these gaps, this thesis presents a sequence of closely related works detailed in Chapters 3 to 6.

Chapter 3 addresses the first research gap by providing an open-source simulation-based design toolbox for soft pneumatic actuators that facilitates intuitive and interactive design exploration. This work enables human intuition to be rapidly translated into physically feasible actuator designs through fast simulation and high-fidelity evaluation. By shortening the iteration loop between design intent, simulation feedback, and fabrication, the proposed toolbox allows designers to explore design variations efficiently while retaining physical realism, thereby reducing reliance on empirical trial-and-error. Building on this modelling foundation, Chapter 4 tackles the second research gap by enabling automated inverse design in large and nonlinear design spaces where intuition alone becomes insufficient. By combining a modular actuator representation with FEM-based surrogate modelling, this chapter enables optimisation-based inverse design for shape matching and delivers SPADA as an end-to-end toolbox that converts target deformation profiles into ready-to-print actuator designs. As a result, task-driven design is no longer solely dependent on manual parameter tuning, allowing designers to explore a broader range of candidate actuator designs for shape-based task objectives. To investigate optimisation trade-offs in task-driven soft actuator design, Chapter 5 investigated the performance of different model–algorithm pairings under a shape-matching design objective. This work compared representative optimisation algorithms combined with different actuator models in terms of computational cost, convergence behaviour, and achieved design quality. The results provide practical guidance on selecting model–algorithm combinations under different design constraints. Finally, Chapter 6 addresses the challenge of generalisation and reuse by proposing a unified modelling and simulation pipeline for task-driven design. This chapter establishes a unified modelling and simulation pipeline in which modelling, optimisation, and task-level simulation can be reused across parameterised actuator designs and task

scenarios by combining high-fidelity modelling with reduced-order representations and design-conditioned behaviour models. In doing so, it enables task-driven design to be performed without reconstructing the entire modelling pipeline when design parameters or task contexts change within the considered design space.

Taken together, this thesis establishes a task-driven design methodology that shifts soft actuator design from case-specific exploration toward systematic, model-informed decision making. Rather than treating modelling, optimisation, and simulation as isolated tools, the proposed approach integrates them into a coherent workflow that connects task objectives with manufacturable actuator designs. This enables designers to move beyond ad hoc parameter tuning and to reason about actuator behaviour, design choices, and task performance within a unified computational framework. Crucially, this methodology is grounded in physically informed models and fabrication-compatible representations, ensuring that the resulting designs remain practical and deployable in real robotic systems. By supporting both human-guided exploration and optimisation-based automation within a common pipeline, the work provides a flexible foundation for task-driven soft actuator design that can be extended and combined in multiple ways, opening up opportunities for further methodological development and application.

Building upon this foundation, future research may be directed not only towards methodological improvements, but also towards a broader scientific understanding of embodied intelligence. More fundamentally, the work presented in this thesis represents an initial step towards a future in which robot morphology becomes an active computational resource rather than a passive mechanical structure, allowing task requirements to be systematically translated into physical design through computational reasoning.

Achieving this vision requires progress on several fundamental scientific questions. First, extending task-driven design beyond individual parameterised actuator families may require a shift in how computational design problems are formulated. Rather than directly optimising actuator geometries for individual tasks, future research should investigate behaviour-level computational abstractions that mediate

between task requirements and actuator-specific geometries. Such a hierarchical formulation could enable transferable design principles across actuator families while maintaining direct links to manufacturable actuator designs.

A second challenge concerns the representation of physical knowledge. While recent advances in robot learning have demonstrated impressive capabilities, learned policies often remain task-specific and require extensive retraining when embodiments, environments, or task conditions change. Rather than relying solely on increasingly capable learning algorithms, future research should investigate how transferable physical knowledge can be represented and integrated into learning systems. From this perspective, physically meaningful behavioural abstractions may provide a bridge between first-principles modelling and data-driven learning, enabling robots to generalise through reusable physical understanding rather than task-specific experience alone.

Together, these directions suggest that the next stage of task-driven design research lies not simply in developing increasingly sophisticated optimisation frameworks, but in establishing scientific principles for representing morphology and physical knowledge as reusable computational abstractions. Such advances would significantly broaden the applicability of task-driven design while contributing towards a deeper understanding of embodied intelligence in robotic systems.

Appendices

A

List of Supporting Videos

- A video demonstrating how to use the simulation toolbox (discussed in Chapter 3) is provided here: `https://github.com/ORI-SRL/Bellow_SPA_Toolbox/blob/main/SupplementaryVideo.mp4`
- A video demonstrating how to use the shape-matching toolbox SPADA (discussed in Chapter 4) is provided here: `https://github.com/ORI-SRL/SPADA/blob/main/SupplementaryVideo.mp4`
- A video including recordings of experiments and simulations for our proposed task-driven design pipeline (discussed in Chapter 6) is provided here: `https://github.com/ORI-SRL/SPADA/blob/main/SupplementaryVideo.mp4`

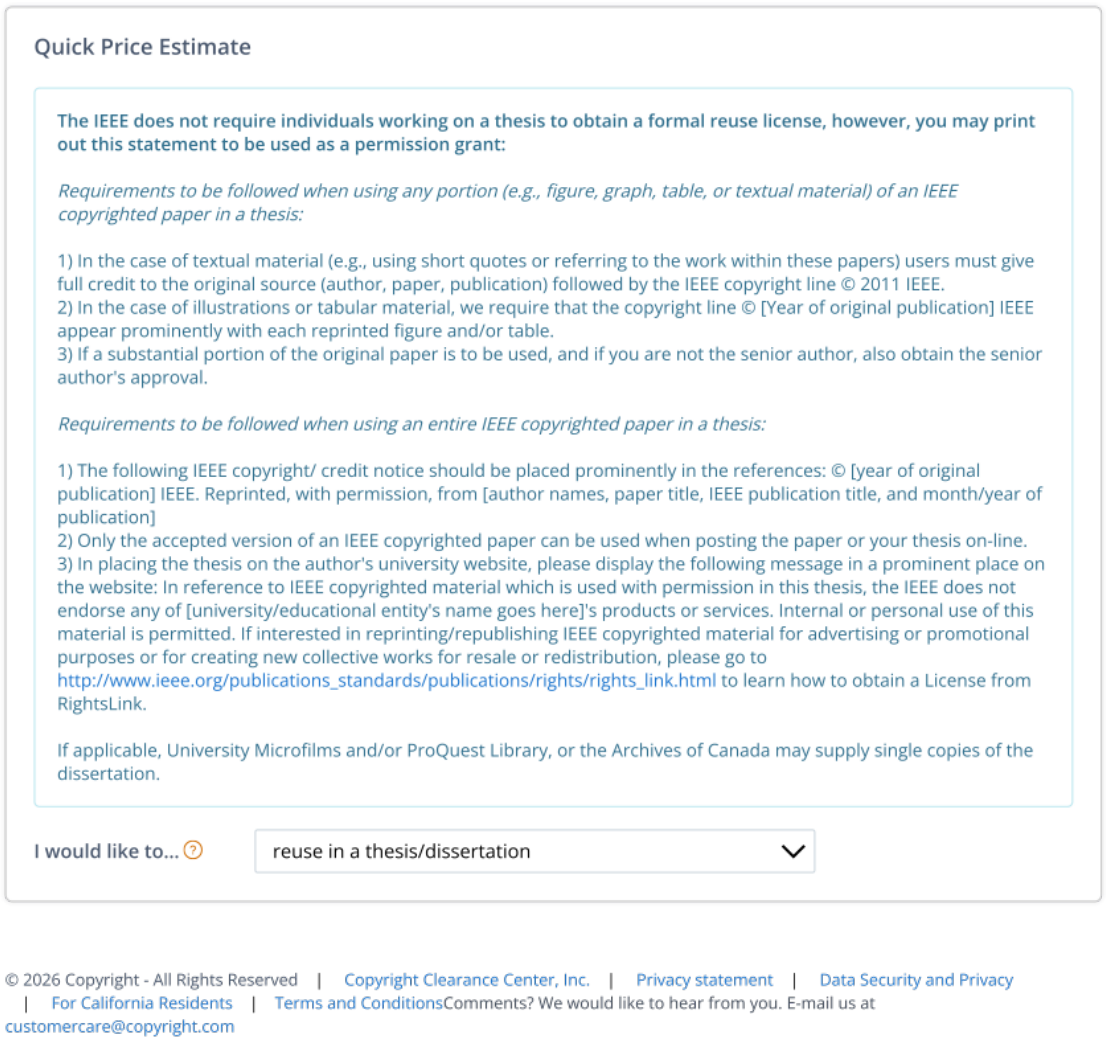
B

Copyright Permissions

The Literature Review chapter reuses figures from several published research articles. Here are the permissions for reusing those figures.

B.1 Permissions Granted by Publishers

Here is the permission for papers published by IEEE.


Quick Price Estimate

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

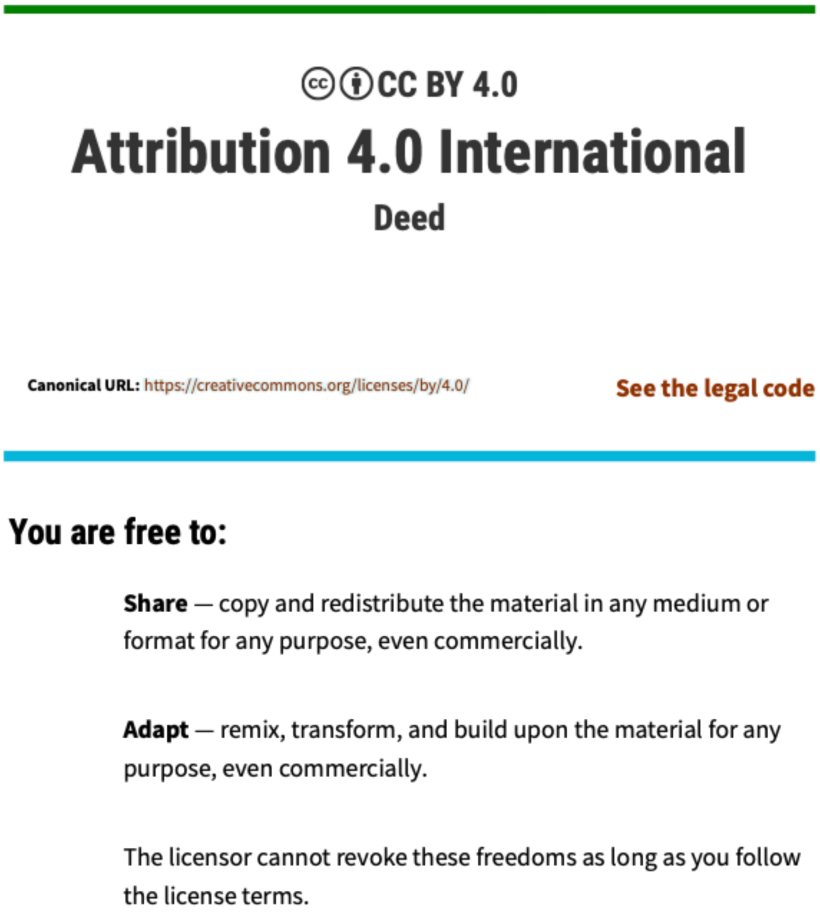
If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

I would like to... reuse in a thesis/dissertation

© 2026 Copyright - All Rights Reserved | Copyright Clearance Center, Inc. | Privacy statement | Data Security and Privacy | For California Residents | Terms and ConditionsComments? We would like to hear from you. E-mail us at customercare@copyright.com

Figure B.1: Permissions from IEEE for reusing figures for thesis.

Here is the permission licensed under CC BY 4.0 for papers published by Advanced Intelligent Systems.



CC BY 4.0

Attribution 4.0 International

Deed

Canonical URL: https://creativecommons.org/licenses/by/4.0/ **See the legal code**

You are free to:

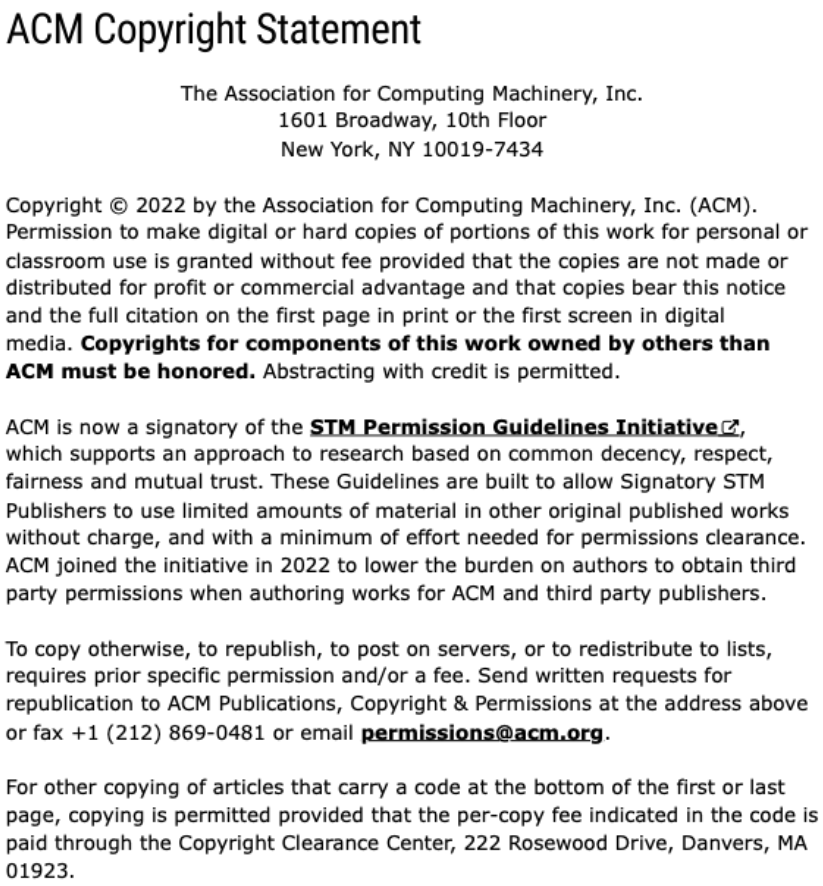
Share — copy and redistribute the material in any medium or format for any purpose, even commercially.

Adapt — remix, transform, and build upon the material for any purpose, even commercially.

The licensor cannot revoke these freedoms as long as you follow the license terms.

Figure B.2: Permissions from Advanced Intelligent Systems licensed with CC BY 4.0 for reusing figures for thesis.

Here is the permission for papers published by the Association for Computing Machinery.



ACM Copyright Statement

The Association for Computing Machinery, Inc.
1601 Broadway, 10th Floor
New York, NY 10019-7434

Copyright © 2022 by the Association for Computing Machinery, Inc. (ACM). Permission to make digital or hard copies of portions of this work for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page in print or the first screen in digital media. **Copyrights for components of this work owned by others than ACM must be honored.** Abstracting with credit is permitted.

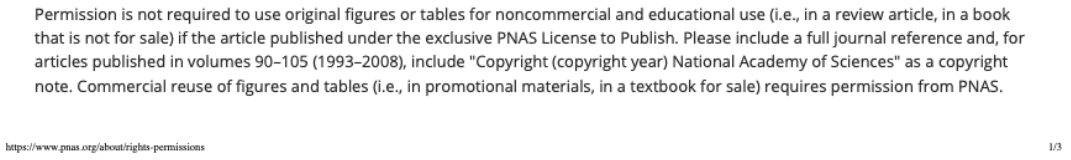
ACM is now a signatory of the **STM Permission Guidelines Initiative**, which supports an approach to research based on common decency, respect, fairness and mutual trust. These Guidelines are built to allow Signatory STM Publishers to use limited amounts of material in other original published works without charge, and with a minimum of effort needed for permissions clearance. ACM joined the initiative in 2022 to lower the burden on authors to obtain third party permissions when authoring works for ACM and third party publishers.

To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee. Send written requests for republication to ACM Publications, Copyright & Permissions at the address above or fax +1 (212) 869-0481 or email **permissions@acm.org**.

For other copying of articles that carry a code at the bottom of the first or last page, copying is permitted provided that the per-copy fee indicated in the code is paid through the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923.

Figure B.3: Permissions from the Association for Computing Machinery for reusing figures for thesis.

Here is the permission for papers published by the Proceedings of the National Academy of Sciences.



Permission is not required to use original figures or tables for noncommercial and educational use (i.e., in a review article, in a book that is not for sale) if the article published under the exclusive PNAS License to Publish. Please include a full journal reference and, for articles published in volumes 90–105 (1993–2008), include "Copyright (copyright year) National Academy of Sciences" as a copyright note. Commercial reuse of figures and tables (i.e., in promotional materials, in a textbook for sale) requires permission from PNAS.

https://www.pnas.org/about/rights-permissions 1/3

Figure B.4: Permissions from the Proceedings of the National Academy of Sciences for reusing figures for thesis.

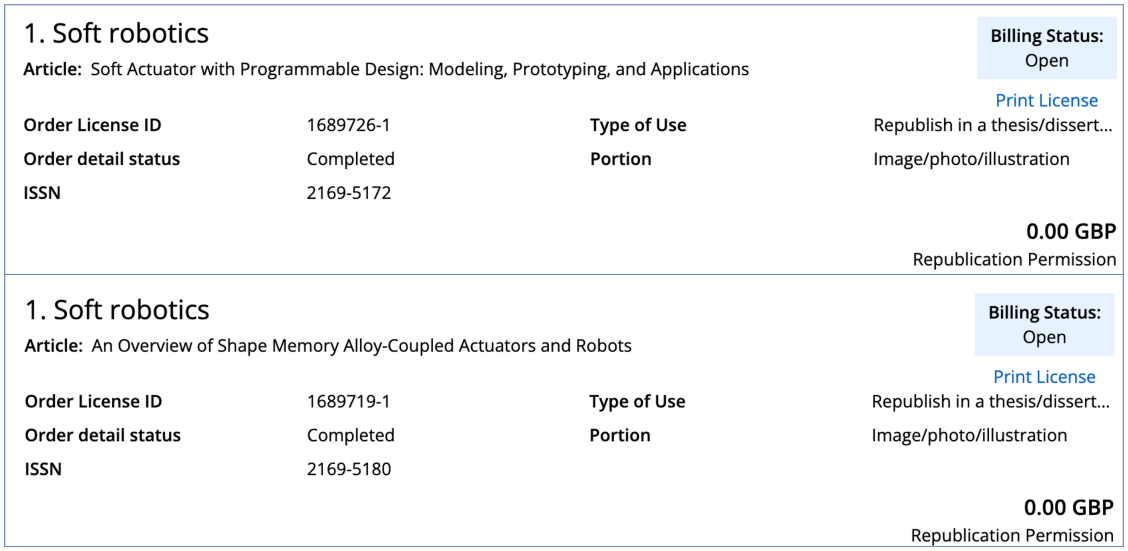
B.2 Permissions of other papers

Here are the copyright permissions for papers via CCC RightsLink.

Date	Article Title	Publication	Type Of Use	Price	Status	Expiration Date	Order Number
12-Jan-2026	Sim-to-real transfer of co-optimized soft robot crawlers	Autonomous Robots	Thesis/Dissertation	0.00 £	Completed		6186540172099
12-Jan-2026	Design and Kinematic Modeling of Constant Curvature Continuum Robots: A Review	The International Journal of Robotics Research	Dissertation/Thesis	0.00 £	Completed		6186530836086
12-Jan-2026	Modeling analysis of soft Pneu-Net actuators from low to high pressure in free space	Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science	Dissertation/Thesis	0.00 £	Completed		6186520582987
12-Jan-2026	Pneumatic Networks for Soft Robotics that Actuate Rapidly	Advanced Functional Materials	Dissertation/Thesis	0.00 £	Completed		6186520115149
12-Jan-2026	Vibration transport system for lunar and Martian regolith using dielectric elastomer actuator	Journal of Electrostatics	reuse in a thesis/dissertation	0.00 £	Completed		6186511386631

Figure B.5: Permissions for papers via RightsLink.

Here are the copyright permissions for papers via CCC Marketplace.



1. Soft robotics

Article: Soft Actuator with Programmable Design: Modeling, Prototyping, and Applications

Billing Status: Open

Print License

Order License ID	1689726-1	**Type of Use**	Republish in a thesis/dissert...
Order detail status	Completed	**Portion**	Image/photo/illustration
ISSN	2169-5172		

0.00 GBP

Republication Permission

1. Soft robotics

Article: An Overview of Shape Memory Alloy-Coupled Actuators and Robots

Billing Status: Open

Print License

Order License ID	1689719-1	**Type of Use**	Republish in a thesis/dissert...
Order detail status	Completed	**Portion**	Image/photo/illustration
ISSN	2169-5180		

0.00 GBP

Republication Permission

Figure B.6: Permissions for papers via Marketplace

References

[1] Daniela Rus and Michael T. Tolley. "Design, fabrication and control of soft robots". In: *Nature* 521.7553 (2015), pp. 467–475. URL: https://doi.org/10.1038/nature14543.

[2] Chiwon Lee et al. "Soft robot review". In: *International Journal of Control, Automation and Systems* 15.1 (2017), pp. 3–15. URL: https://doi.org/10.1007/s12555-016-0462-3.

[3] Cecilia Laschi, Barbara Mazzolai, and Matteo Cianchetti. "Soft robotics: Technologies and systems pushing the boundaries of robot abilities". In: *Science Robotics* 1.1 (2016), eaah3690. eprint: https://www.science.org/doi/pdf/10.1126/scirobotics.aah3690. URL: https://www.science.org/doi/abs/10.1126/scirobotics.aah3690.

[4] Jun Shintake et al. "Soft Robotic Grippers". In: *Advanced Materials* 30.29 (2018), p. 1707035. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/adma.201707035. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/adma.201707035.

[5] M. Calisti, G. Picardi, and C. Laschi. "Fundamentals of soft robot locomotion". In: *Journal of The Royal Society Interface* 14.130 (2017), p. 20170101. eprint: https://royalsocietypublishing.org/doi/pdf/10.1098/rsif.2017.0101. URL: https://royalsocietypublishing.org/doi/abs/10.1098/rsif.2017.0101.

[6] Matteo Cianchetti et al. "Biomedical applications of soft robotics". In: *Nature Reviews Materials* 3.6 (2018), 143–153.

[7] Panagiotis Polygerinos et al. "Modeling of Soft Fiber-Reinforced Bending Actuators". In: *IEEE Transactions on Robotics* 31.3 (2015), pp. 778–789.

[8] Bobak Mosadegh et al. "Pneumatic Networks for Soft Robotics that Actuate Rapidly". In: *Advanced Functional Materials* 24.15 (2014), pp. 2163–2170. eprint: https://advanced.onlinelibrary.wiley.com/doi/pdf/10.1002/adfm.201303288. URL: https://advanced.onlinelibrary.wiley.com/doi/abs/10.1002/adfm.201303288.

[9] Hugo Rodrigue et al. "An Overview of Shape Memory Alloy-Coupled Actuators and Robots". In: *Soft Robotics* 4.1 (2017). PMID: 29182099, pp. 3–15. eprint: https://doi.org/10.1089/soro.2016.0008. URL: https://doi.org/10.1089/soro.2016.0008.

[10] M. Adachi et al. “Vibration transport system for lunar and Martian regolith using dielectric elastomer actuator”. In: *Journal of Electrostatics* 89 (2017), pp. 88–98. URL: https://www.sciencedirect.com/science/article/pii/S0304388617300165.

[11] Dylan Drotman et al. “Application-Driven Design of Soft, 3-D Printed, Pneumatic Actuators With Bellows”. In: *IEEE/ASME Transactions on Mechatronics* 24.1 (2019), pp. 78–87.

[12] Ching-Ping Chou and B. Hannaford. “Measurement and modeling of McKibben pneumatic artificial muscles”. In: *IEEE Transactions on Robotics and Automation* 12.1 (1996), pp. 90–102.

[13] Frank Daerden and Dirk Lefeber. “Pneumatic artificial muscles: Actuators for robotics and automation”. In: *European journal of mechanical and environmental engineering* 47 (2002), pp. 11–21. URL: https://api.semanticscholar.org/CorpusID:14944063.

[14] Kevin C. Galloway et al. “Mechanically programmable bend radius for fiber-reinforced soft actuators”. In: *2013 16th International Conference on Advanced Robotics (ICAR)*. 2013, pp. 1–6.

[15] Tianyu Wang, Lisen Ge, and Guoying Gu. “Programmable design of soft pneu-net actuators with oblique chambers can generate coupled bending and twisting motions”. In: *Sensors and Actuators A: Physical* 271 (2018), pp. 131–138. URL: https://www.sciencedirect.com/science/article/pii/S0924424717316977.

[16] Filip Ilievski et al. “Soft Robotics for Chemists”. In: *Angewandte Chemie* 123.8 (2011), 1930–1935.

[17] Yoseph Bar-Cohen. “Electroactive Polymers as Artificial Muscles: A Review”. In: *Journal of Spacecraft and Rockets* 39.6 (2002), pp. 822–827. eprint: https://doi.org/10.2514/2.3902. URL: https://doi.org/10.2514/2.3902.

[18] Dileep Kumar Soother, Jawaid Daudpoto, and Bhawani Shankar Chowdhry. “Challenges for practical applications of shape memory alloy actuators”. In: *Materials Research Express* 7.7 (2020), p. 073001.

[19] Hyun-Joong Chung, Andrew M. Parsons, and Lelin Zheng. “Magnetically Controlled Soft Robotics Utilizing Elastomers and Gels in Actuation: A Review”. In: *Advanced Intelligent Systems* 3.3 (2021), p. 2000186. eprint: https://advanced.onlinelibrary.wiley.com/doi/pdf/10.1002/aisy.202000186. URL: https://advanced.onlinelibrary.wiley.com/doi/abs/10.1002/aisy.202000186.

[20] Ailish O’Halloran, Fergal O’Malley, and Peter McHugh. “A review on dielectric elastomer actuators, technology, applications, and challenges”. In: *Journal of Applied Physics* 104.7 (Oct. 2008), p. 071101. eprint: https://pubs.aip.org/aip/jap/article-pdf/doi/10.1063/1.2981642/15020155/071101_1_online.pdf. URL: https://doi.org/10.1063/1.2981642.

[21] Ganesha Udupa et al. “Asymmetric Bellow Flexible Pneumatic Actuator for Miniature Robotic Soft Gripper”. In: *Journal of Robotics* 2014.1 (2014), p. 902625. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1155/2014/902625. URL: https://onlinelibrary.wiley.com/doi/abs/10.1155/2014/902625.

[22] Kanchana Crishan Wickramatunge and Thananchai Leephakpreeda. “Study on mechanical behaviors of pneumatic artificial muscle”. In: *International Journal of Engineering Science* 48.2 (2010), pp. 188–198. URL: https://www.sciencedirect.com/science/article/pii/S0020722509001268.

[23] Bu Hyun Shin et al. “A modular torsional actuator using shape memory alloy wires”. In: *Journal of Intelligent Material Systems and Structures* 27.12 (2016), pp. 1658–1665. eprint: https://doi.org/10.1177/1045389X15600084. URL: https://doi.org/10.1177/1045389X15600084.

[24] Siddharth Sanan, Peter S. Lynn, and Saul T. Griffith. “Pneumatic Torsional Actuators for Inflatable Robots”. In: *Journal of Mechanisms and Robotics* 6.3 (Apr. 2014), p. 031003. eprint: https://asmedigitalcollection.asme.org/mechanismsrobotics/article-pdf/6/3/031003/6251333/jmr_006_03_031003.pdf. URL: https://doi.org/10.1115/1.4026629.

[25] Vincent Wall, Raphael Deimel, and Oliver Brock. “Selective stiffening of soft actuators based on jamming”. In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*. 2015, pp. 252–257.

[26] Callie Branyan et al. “Soft snake robots: Mechanical design and geometric gait implementation”. In: *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. 2017, pp. 282–289.

[27] M Calisti et al. “An octopus-bioinspired solution to movement and manipulation for soft robots”. In: *Bioinspiration & Biomimetics* 6.3 (2011), p. 036002.

[28] Raphael Deimel and Oliver Brock. “A novel type of compliant and underactuated robotic hand for dexterous grasping”. In: *The International Journal of Robotics Research* 35.1-3 (2016), pp. 161–185. eprint: https://doi.org/10.1177/0278364915592961. URL: https://doi.org/10.1177/0278364915592961.

[29] Mariangela Manti et al. “A Bioinspired Soft Robotic Gripper for Adaptable and Effective Grasping”. In: *Soft Robotics* 2.3 (2015), pp. 107–116. eprint: https://doi.org/10.1089/soro.2015.0009. URL: https://doi.org/10.1089/soro.2015.0009.

[30] Jianshu Zhou, Shu Chen, and Zheng Wang. “A Soft-Robotic Gripper With Enhanced Object Adaptation and Grasping Reliability”. In: *IEEE Robotics and Automation Letters* 2.4 (2017), pp. 2287–2293.

[31] Giovanni Rateni et al. “Design and development of a soft robotic gripper for manipulation in minimally invasive surgery: a proof of concept”. In: *Meccanica* 50.11 (2015), 2855–2863.

[32] Robert F. Shepherd et al. “Multigait soft robot”. In: *Proceedings of the National Academy of Sciences* 108.51 (2011), pp. 20400–20403. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.1116564108. URL: https://www.pnas.org/doi/abs/10.1073/pnas.1116564108.

[33] Sangok Seok et al. “Meshworm: A Peristaltic Soft Robot With Antagonistic Nickel Titanium Coil Actuators”. In: *IEEE/ASME Transactions on Mechatronics* 18.5 (2013), pp. 1485–1497.

[34] Matteo Cianchetti et al. “STIFF-FLOP surgical manipulator: Mechanical design and experimental characterization of the single module”. In: *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2013, pp. 3576–3581.

[35] Hong Kai Yap et al. “A soft exoskeleton for hand assistive and rehabilitation application using pneumatic actuators with variable stiffness”. In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*. 2015, pp. 4967–4972.

[36] Yanyan Wu et al. “Modeling analysis of soft Pneu-Net actuators from low to high pressure in free space”. In: *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* 239.13 (2025), pp. 5034–5049. eprint: https://doi.org/10.1177/09544062251323751. URL: https://doi.org/10.1177/09544062251323751.

[37] Costanza Armanini et al. “Soft Robots Modeling: A Structured Overview”. In: *IEEE Transactions on Robotics* 39.3 (2023), pp. 1728–1748.

[38] Zicheng Kan et al. “Soft Actuator with Programmable Design: Modeling, Prototyping, and Applications”. In: *Soft Robotics* 9.5 (2022). PMID: 35005997, pp. 907–925. eprint: https://doi.org/10.1089/soro.2020.0148. URL: https://doi.org/10.1089/soro.2020.0148.

[39] Matheus S. Xavier, Andrew J. Fleming, and Yuen K. Yong. “Finite Element Modeling of Soft Fluidic Actuators: Overview and Recent Developments”. In: *Advanced Intelligent Systems* 3.2 (2021), p. 2000187. eprint: https://advanced.onlinelibrary.wiley.com/doi/pdf/10.1002/aisy.202000187. URL: https://advanced.onlinelibrary.wiley.com/doi/abs/10.1002/aisy.202000187.

[40] Robert J. WebsterIII and Bryan A. Jones. “Design and Kinematic Modeling of Constant Curvature Continuum Robots: A Review”. In: *The International Journal of Robotics Research* 29.13 (2010), pp. 1661–1683. eprint: https://doi.org/10.1177/0278364910368147. URL: https://doi.org/10.1177/0278364910368147.

[41] Haiyun Zhang et al. “A Novel Parameter Estimation Method for Pneumatic Soft Hand Control Applying Logarithmic Decrement for Pseudo-Rigid Body Modeling”. In: *Advanced Intelligent Systems* 7.3 (2025), p. 2400637. eprint: https://advanced.onlinelibrary.wiley.com/doi/pdf/10.1002/aisy.202400637. URL: https://advanced.onlinelibrary.wiley.com/doi/abs/10.1002/aisy.202400637.

[42] Sam Kriegman et al. “Simulating the evolution of soft and rigid-body robots”. In: *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. GECCO ’17. Berlin, Germany: Association for Computing Machinery, 2017, 1117–1120. URL: https://doi.org/10.1145/3067695.3082051.

[43] Yuanming Hu et al. “ChainQueen: A Real-Time Differentiable Physical Simulator for Soft Robotics”. In: *2019 International Conference on Robotics and Automation (ICRA)*. 2019, pp. 6265–6271.

[44] Pasquale Ferrentino et al. “Finite Element Analysis-Based Soft Robotic Modeling: Simulating a Soft Actuator in SOFA”. In: *IEEE Robotics & Automation Magazine* 31.3 (2024), pp. 97–105.

[45] Myunggi Ji et al. “Rapid Design and Analysis of Microtube Pneumatic Actuators Using Line-Segment and Multi-Segment Euler–Bernoulli Beam Models”. In: *Micromachines* 10.11 (2019). URL: https://www.mdpi.com/2072-666X/10/11/780.

[46] Guoliang Zhong, Yangdong Hou, and Weiqiang Dou. “A soft pneumatic dexterous gripper with convertible grasping modes”. In: *International Journal of Mechanical Sciences* 153-154 (2019), pp. 445–456. URL: https://www.sciencedirect.com/science/article/pii/S0020740318323324.

[47] Kristin M. de Payrebrune and Oliver M. O’Reilly. “On constitutive relations for a rod-based model of a pneu-net bending actuator”. In: *Extreme Mechanics Letters* 8 (2016). Nanomechanics: Bridging Spatial and Temporal Scales, pp. 38–46. URL: https://www.sciencedirect.com/science/article/pii/S2352431615300122.

[48] Simon R. Eugster et al. “Soft Pneumatic Actuator Model Based on a Pressure-Dependent Spatial Nonlinear Rod Theory”. In: *IEEE Robotics and Automation Letters* 7.2 (2022), pp. 2471–2478.

[49] Lingxiao Xun, Gang Zheng, and Alexandre Kruszewski. “Cosserat-Rod-Based Dynamic Modeling of Soft Slender Robot Interacting With Environment”. In: *IEEE Transactions on Robotics* 40 (2024), pp. 2811–2830.

[50] Fionnuala Connolly, Conor J. Walsh, and Katia Bertoldi. “Automatic design of fiber-reinforced soft actuators for trajectory matching”. In: *Proceedings of the National Academy of Sciences* 114.1 (2017), pp. 51–56. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.1615140114. URL: https://www.pnas.org/doi/abs/10.1073/pnas.1615140114.

[51] Guoying Gu et al. “Analytical Modeling and Design of Generalized Pneu-Net Soft Actuators with Three-Dimensional Deformations”. In: *Soft Robotics* 8.4 (2021). PMID: 32822253, pp. 462–477. eprint: https://doi.org/10.1089/soro.2020.0039. URL: https://doi.org/10.1089/soro.2020.0039.

[52] Max Bartholdt et al. “A Parameter Identification Method for Static Cosserat Rod Models: Application to Soft Material Actuators with Exteroceptive Sensors”. In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2021, pp. 624–631.

[53] Philip Moseley et al. “Modeling, Design, and Development of Soft Pneumatic Actuators with Finite Element Method”. In: *Advanced Engineering Materials* 18.6 (2016), pp. 978–988. eprint: https://advanced.onlinelibrary.wiley.com/doi/pdf/10.1002/adem.201500503. URL: https://advanced.onlinelibrary.wiley.com/doi/abs/10.1002/adem.201500503.

[54] Fionnuala Connolly et al. "Mechanical Programming of Soft Actuators by Varying Fiber Angle". In: *Soft Robotics* 2.1 (2015), pp. 26–32. eprint: https://doi.org/10.1089/soro.2015.0001. URL: https://doi.org/10.1089/soro.2015.0001.

[55] Anas Kanan and Michael Kaliske. "Finite element modeling of electro-viscoelasticity in fiber reinforced electro-active polymers". In: *International Journal for Numerical Methods in Engineering* 122.8 (2021), pp. 2005–2037. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.6610. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.6610.

[56] Y. Elsayed et al. "Design optimisation of soft silicone pneumatic actuators using finite element analysis". In: *2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. 2014, pp. 44–49.

[57] Fei Yang et al. "Design and Optimize of a Novel Segmented Soft Pneumatic Actuator". In: *IEEE Access* 8 (2020), pp. 122304–122313.

[58] Gundula Runge, Jan Peters, and Annika Raatz. "Design optimization of soft pneumatic actuators using genetic algorithms". In: *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. 2017, pp. 393–400.

[59] COMSOL AB. *COMSOL Multiphysics® v6.1*. Available at: https://www.comsol.com/. COMSOL AB. Stockholm, Sweden, 2022.

[60] ANSYS, Inc. *ANSYS Mechanical™*. Available at: https://www.ansys.com/. ANSYS, Inc. Canonsburg, PA, USA, 2022.

[61] Dassault Systèmes Simulia Corp. *Abaqus® Unified FEA, Version 2022*. Available at: https://www.3ds.com/products-services/simulia/products/abaqus/. Dassault Systèmes. Johnston, RI, USA, 2022.

[62] Zixi Chen et al. "Data-Driven Methods Applied to Soft Robot Modeling and Control: A Review". In: *IEEE Transactions on Automation Science and Engineering* 22 (2025), pp. 2241–2256.

[63] Olivier Goury and Christian Duriez. "Fast, Generic, and Reliable Control and Simulation of Soft Robots Using Model Order Reduction". In: *IEEE Transactions on Robotics* 34.6 (2018), pp. 1565–1576.

[64] Gundula Runge, Mats Wiese, and Annika Raatz. "FEM-based training of artificial neural networks for modular soft robots". In: *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. 2017, pp. 385–392.

[65] Carlos Relaño, Javier Muñoz, and Concepción A. Monje. "Gaussian process regression for forward and inverse kinematics of a soft robotic arm". In: *Engineering Applications of Artificial Intelligence* 126 (2023), p. 107174. URL: https://www.sciencedirect.com/science/article/pii/S0952197623013581.

[66] Robert K. Katzschmann et al. "Dynamic Motion Control of Multi-Segment Soft Robots Using Piecewise Constant Curvature Matched with an Augmented Rigid Body Model". In: *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*. 2019, pp. 454–461.

[67] Fan Xu et al. “Adaptive Visual Servoing Shape Control of a Soft Robot Manipulator Using Bézier Curve Features”. In: *IEEE/ASME Transactions on Mechatronics* 28.2 (2023), pp. 945–955.

[68] Sreeshankar Satheeshbabu and Girish Krishnan. “Modeling the Bending Behavior of Fiber-Reinforced Pneumatic Actuators Using a Pseudo-Rigid-Body Model”. In: *Journal of Mechanisms and Robotics* 11.3 (Sept. 2019), p. 031011. eprint: https://asmedigitalcollection.asme.org/mechanismsrobotics/article-pdf/11/3/031011/6518408/jmr_11_3_031011.pdf. URL: https://doi.org/10.1115/1.4042632.

[69] Venkatasubramanian Kalpathy Venkiteswaran, Jakub Sikorski, and Sarthak Misra. “Shape and contact force estimation of continuum manipulators using pseudo rigid body models”. In: *Mechanism and Machine Theory* 139 (2019), pp. 34–45. URL: https://www.sciencedirect.com/science/article/pii/S0094114X18318706.

[70] Moritz A. Graule et al. “SoMo: Fast and Accurate Simulations of Continuum Robots in Complex Environments”. In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2021, pp. 3934–3941.

[71] Emanuel Todorov, Tom Erez, and Yuval Tassa. “MuJoCo: A physics engine for model-based control”. In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2012, pp. 5026–5033.

[72] Sam Kriegman et al. “Scalable sim-to-real transfer of soft robot designs”. In: *2020 3rd IEEE International Conference on Soft Robotics (RoboSoft)*. 2020, pp. 359–366.

[73] Jonathan Hiller and Hod Lipson. “Dynamic Simulation of Soft Multimaterial 3D-Printed Objects”. In: *Soft Robotics* 1.1 (2014), pp. 88–101. eprint: https://doi.org/10.1089/soro.2013.0010. URL: https://doi.org/10.1089/soro.2013.0010.

[74] Andrew Spielberg et al. “Learning-In-The-Loop Optimization: End-To-End Control And Co-Design Of Soft Robots Through Learned Deep Latent Representations”. In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach et al. Vol. 32. Curran Associates, Inc., 2019. URL: https://proceedings.neurips.cc/paper_files/paper/2019/file/438124b4c06f3a5caffab2c07863b617-Paper.pdf.

[75] Julie Legrand et al. “Reconfigurable, Multi-Material, Voxel-Based Soft Robots”. In: *IEEE Robotics and Automation Letters* 8.3 (2023), pp. 1255–1262.

[76] Joshua Davy et al. “A Framework for Simulation of Magnetic Soft Robots Using the Material Point Method”. In: *IEEE Robotics and Automation Letters* 8.6 (2023), pp. 3470–3477.

[77] Siwei He et al. “A material point-based simulation method for soft robots with free boundary interactions”. In: *Robotica* 43.2 (2025), 582–595.

[78] James M. Bern et al. “Soft Robot Control With a Learned Differentiable Model”. In: *2020 3rd IEEE International Conference on Soft Robotics (RoboSoft)*. 2020, pp. 417–423.

[79] Mathieu Dubied et al. "Sim-to-Real for Soft Robots Using Differentiable FEM: Recipes for Meshing, Damping, and Actuation". In: *IEEE Robotics and Automation Letters* 7.2 (2022), pp. 5015–5022.

[80] Tsun-Hsuan Wang et al. *SoftZoo: A Soft Robot Co-design Benchmark For Locomotion In Diverse Environments*. 2023. arXiv: 2303.09555 [cs.RO]. URL: https://arxiv.org/abs/2303.09555.

[81] John Z. Zhang et al. "Sim2Real for Soft Robotic Fish via Differentiable Simulation". In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2022, pp. 12598–12605.

[82] Elvis Nava et al. "Fast Aquatic Swimmer Optimization with Differentiable Projective Dynamics and Neural Network Hydrodynamic Models". In: *Proceedings of the 39th International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri et al. Vol. 162. Proceedings of Machine Learning Research. PMLR, 2022, pp. 16413–16427. URL: https://proceedings.mlr.press/v162/nava22a.html.

[83] Shaohong Zhong et al. "An End-to-End Framework for Modeling Pneumatic Soft Robots Based on Differentiable Finite Element Methods". In: *IEEE Robotics and Automation Letters* 10.12 (2025), pp. 13003–13010.

[84] Christian Duriez. "Control of elastic soft robots based on real-time finite element method". In: *2013 IEEE International Conference on Robotics and Automation*. 2013, pp. 3982–3987.

[85] Stefan Escaida Navarro et al. "An Open Source Design Optimization Toolbox Evaluated on a Soft Finger". In: *IEEE Robotics and Automation Letters* 8.9 (2023), pp. 6044–6051.

[86] Hongying Zhang et al. "Topology Optimized Multimaterial Soft Fingers for Applications on Grippers, Rehabilitation, and Artificial Hands". In: *IEEE/ASME Transactions on Mechatronics* 24.1 (2019), pp. 120–131.

[87] Wu-Te Yang, Hannah S. Stuart, and Masayoshi Tomizuka. "Mechanical Modeling and Optimal Model-based Design of a Soft Pneumatic Actuator". In: *2023 IEEE International Conference on Soft Robotics (RoboSoft)*. 2023, pp. 1–7.

[88] Feifei Chen et al. "Topology Optimized Design, Fabrication, and Characterization of a Soft Cable-Driven Gripper". In: *IEEE Robotics and Automation Letters* 3.3 (2018), pp. 2463–2470.

[89] Jun Liu et al. "Simulation Data Driven Design Optimization for Reconfigurable Soft Gripper System". In: *IEEE Robotics and Automation Letters* 7.2 (2022), pp. 5803–5810.

[90] Mohamed G. B. Atia et al. "Reconfigurable Soft Robots by Building Blocks". In: *Advanced Science* 9.33 (2022), p. 2203217. eprint: https://advanced.onlinelibrary.wiley.com/doi/pdf/10.1002/advs.202203217. URL: https://advanced.onlinelibrary.wiley.com/doi/abs/10.1002/advs.202203217.

[91] Xing Wang et al. "Fin-Bayes: A Multi-Objective Bayesian Optimization Framework for Soft Robotic Fingers". In: *Soft Robotics* 11.5 (2024). PMID: 38498017, pp. 791–801. eprint: https://doi.org/10.1089/soro.2023.0134. URL: https://doi.org/10.1089/soro.2023.0134.

[92] Ali Zolfagharian et al. "4D printing soft robots guided by machine learning and finite element models". In: *Sensors and Actuators A: Physical* 328 (2021), p. 112774. URL: https://www.sciencedirect.com/science/article/pii/S0924424721002375.

[93] Seyede Fatemeh Ghoreishi et al. "Bayesian Optimization for Design of Multi-Actuator Soft Catheter Robots". In: *IEEE Transactions on Medical Robotics and Bionics* 3.3 (2021), pp. 725–737.

[94] Zhongkui Wang and Shinichi Hirai. "Chamber dimension optimization of a bellow-type soft actuator for food material handling". In: *2018 IEEE International Conference on Soft Robotics (RoboSoft)*. 2018, pp. 382–387.

[95] Zhongkui Wang and Shinichi Hirai. "Geometry and Material Optimization of a Soft Pneumatic Gripper for Handling Deformable Object". In: *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. 2018, pp. 612–617.

[96] Jiayi Jin, Siyuan Feng, and Shuguang Li. "Computational Design of Customized Vacuum-Driven Soft Grippers". In: *IEEE Robotics and Automation Letters* 10.2 (2025), pp. 1641–1648.

[97] Daniel M. Bodily, Thomas F. Allen, and Marc D. Killpack. "Multi-objective design optimization of a soft, pneumatic robot". In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. 2017, pp. 1864–1871.

[98] Raphael Deimel et al. "Automated co-design of soft hand morphology and control strategy for grasping". In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2017, pp. 1213–1218.

[99] Charles Schaff et al. "Sim-to-real transfer of co-optimized Soft Robot Crawlers". In: *Autonomous Robots* 47.8 (2023), 1195–1211.

[100] Mahtab Behzadfar, Yue Feng, and Ki-Young Song. "Advancing Soft Robotics: A High Energy-Efficient Actuator Design Using Single-Tube Pneumatic for Bidirectional Movement". In: *Journal of Mechanical Design* 147.4 (Mar. 2025), p. 043301. eprint: https://asmedigitalcollection.asme.org/mechanicaldesign/article-pdf/147/4/043301/7439053/md-24-1706.pdf. URL: https://doi.org/10.1115/1.4068073.

[101] Tanguy Navez et al. "Modeling, Embedded Control, and Design of Soft Robots Using a Learned Condensed FEM Model". In: *IEEE Transactions on Robotics* 41 (2025), pp. 2441–2459.

[102] Jagdeep Bhatia et al. "Evolution Gym: A Large-Scale Benchmark for Evolving Soft Robots". In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato et al. Vol. 34. Curran Associates, Inc., 2021, pp. 2201–2214. URL: https://proceedings.neurips.cc/paper_files/paper/2021/file/118921efba23fc329e6560b27861f0c2-Paper.pdf.

[103] Merel van Diepen and Kristina Shea. "Co-Design of the Morphology and Actuation of Soft Robots for Locomotion". In: *Journal of Mechanical Design* 144.8 (June 2022). eprint: https://asmedigitalcollection.asme.org/mechanicaldesign/article-pdf/144/8/083305/6886716/md_144_8_083305.pdf. URL: https://doi.org/10.1115/1.4054522.

[104] Jonathan Hiller and Hod Lipson. "Automatic Design and Manufacture of Soft Robots". In: *IEEE Transactions on Robotics* 28.2 (2012), pp. 457–466.

[105] Moritz A. Graule et al. "SoMoGym: A Toolkit for Developing and Evaluating Controllers and Reinforcement Learning Algorithms for Soft Robots". In: *IEEE Robotics and Automation Letters* 7.2 (2022), pp. 4071–4078.

[106] Nick Cheney et al. "Unshackling evolution: evolving soft robots with multiple materials and a powerful generative encoding". In: *SIGEVOlution* 7.1 (Aug. 2014), 11–23. URL: https://doi.org/10.1145/2661735.2661737.

[107] Agrim Gupta et al. "Embodied intelligence via learning and evolution". In: *Nature Communications* 12.1 (Oct. 2021). URL: http://dx.doi.org/10.1038/s41467-021-25874-z.

[108] Huy Ha, Shubham Agrawal, and Shuran Song. "Fit2Form: 3D Generative Model for Robot Gripper Form Design". In: *Proceedings of the 2020 Conference on Robot Learning*. Ed. by Jens Kober, Fabio Ramos, and Claire Tomlin. Vol. 155. Proceedings of Machine Learning Research. PMLR, 2021, pp. 176–187. URL: https://proceedings.mlr.press/v155/ha21b.html.

[109] Tsun-Hsuan Wang et al. "DiffuseBot: Breeding Soft Robots With Physics-Augmented Generative Diffusion Models". In: *Thirty-seventh Conference on Neural Information Processing Systems*. 2023. URL: https://openreview.net/forum?id=1zo4iioUEs.

[110] Liwei Wang. "Co-design of magnetic soft robots with large deformation and contacts via material point method and topology optimization". In: *Computer Methods in Applied Mechanics and Engineering* 445 (2025), p. 118205. URL: https://www.sciencedirect.com/science/article/pii/S0045782525004773.

[111] Gaurav Singh and Girish Krishnan. "Designing Fiber-Reinforced Soft Actuators for Planar Curvilinear Shape Matching". In: *Soft Robotics* 7.1 (2020). PMID: 31566502, pp. 109–121. eprint: https://doi.org/10.1089/soro.2018.0169. URL: https://doi.org/10.1089/soro.2018.0169.

[112] Chengru Jiang et al. "Modeling and inverse design of bio-inspired multi-segment pneu-net soft manipulators for 3D trajectory motion". In: *Applied Physics Reviews* 8.4 (Dec. 2021), p. 041416. eprint: https://pubs.aip.org/aip/apr/article-pdf/doi/10.1063/5.0054468/20725925/041416_1_5.0054468.pdf. URL: https://doi.org/10.1063/5.0054468.

[113] John Rieffel et al. "Evolving soft robotic locomotion in PhysX". In: *Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers*. GECCO '09. Montreal, Québec, Canada: Association for Computing Machinery, 2009, 2499–2504. URL: https://doi.org/10.1145/1570256.1570351.