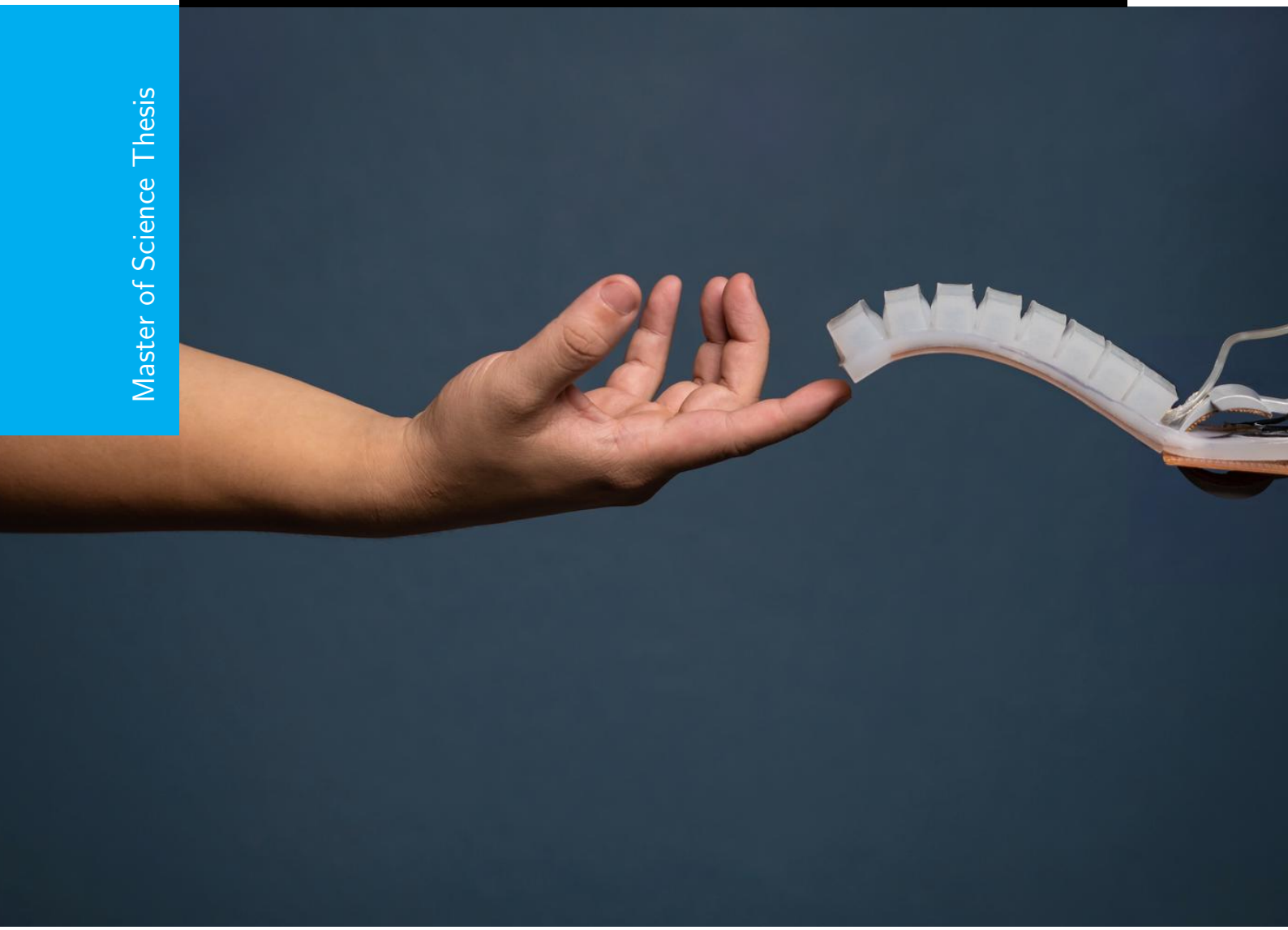
Experimental Validation of Model-Based Control Methods for Shape Regulation in Soft Robots

Ghanishtha Bhatti

Master of Science Thesis



TU Delft Delft University of Technology

Experimental Validation of Model-Based Control Methods for Shape Regulation in Soft Robots

Master of Science Thesis

For the degree of Master of Science in Embedded Systems

Ghanishtha Bhatti

Student Number:	5457025	
Supervisors:	Dr. C. Della Santina	TU Delft (ME - CoR)
	P. Pustina	DIAG, Sapienza University of Rome
Thesis Committee:	Dr. J. Kober	TU Delft (ME - CoR)
	Dr. C. Della Santina	TU Delft (ME - CoR)
	Dr. F. A. Oliehoek	TU Delft (EEMCS - INSY)

Faculty of Electrical Engineering, Mathematics and Computer Science (EEMCS)
Delft University of Technology

*Cover Image Credits: David Baillot (UC San Diego)

Preface

This report is written to document the research undertaken as part of my Master's thesis project titled "Experimental Validation of Model-Based Control Methods for Shape Regulation in Soft Robots." The submission of this report marks the conclusion of my time as a student at the Delft University of Technology. While developing this project, I gleaned numerous lessons, both technical and non-technical, which will resonate with me for a very long time. It fills me with great pride to contribute to this exciting field of research and witness my academic efforts come to fruition.

It goes without saying that the credit for this accomplishment is shared by all those who supported me during this thesis. Foremost, I would like to express my gratitude to my supervisors, Dr. Cosimo Della Santina and Pietro Pustina, for providing me with the opportunity to work on this interesting topic. I am thankful to both of you for the insightful feedback and your astute guidance throughout the project. Thank you, Pietro, for helping me build the hardware platform and for being the most wonderful mentor – someone I could always count on for advice. I would also like to extend my thanks to Francesco Stella for his counsel and assistance in the manufacturing of the robotic platform. Thanks also to the folks at the Phi-lab, with whom I had the pleasure of engaging in many enjoyable conversations during my thesis.

I am certain that I would not have reached this milestone in my academic career without the constant support of my family and friends. Thank you, Mom and Dad, for always believing in me and encouraging me to pursue my ambitions. Thank you, Khyati, for being my biggest cheerleader and inspiration. Thanks to all my friends for making every challenge easier to bear, and turning living abroad into a less lonely and way more interesting adventure. I am delighted that I could share this part of my life with all of you.

Ghanishtha Bhatti.
20^{th} January 2024

Experimental Validation of Model-Based Control Methods for Shape Regulation in Soft Robots

Ghanishtha Bhatti

***Abstract*—Soft robots are characterized by compliant elements that introduce heightened kinematic complexity compared to their rigid counterparts. Such systems, with infinite degrees of freedom, are inherently underactuated, making precise real-time shape regulation a challenging task. Model-based controllers, utilizing tractable reduced-order modeling methods, have emerged as promising solutions. However, practical implementations of these methods often rely on fully-actuated approximations, overlooking the underactuated nature of these continuum structures. In this study, we aim to experimentally validate model-based controllers that explicitly account for underactuation, surpassing the theoretical feasibility demonstrated in simulation. These controllers incorporate gravity cancellation and compliance compensation using the dynamic model of the robot to achieve superior real-time shape regulation compared to conventional PD/PID controllers. To facilitate this experimental validation, we have built a multi-segment soft robot research platform that includes a passively actuated segment, allowing for the utilization of both actuated and unactuated degrees of freedom in the control feedback loop. Through rigorous experimentation, we provide comprehensive evidence of the efficacy of this class of model-based controllers in controlling unconventionally actuated systems. Consequently, our work bridges the gap between theory and practice, resulting in a practical real-time shape regulation framework that is adaptable to a vast variety of soft robotic systems.**

***Index Terms*—Soft Robotics, Model-Based Control, Experimental Validation, Shape Regulation.**

I. Introduction

Over the past decade, there has been a surge of interest in human-machine interaction, which has fueled significant advancements in the field of soft robotics. This specialized category of robotic systems intentionally incorporates compliant elements into their mechanical structure, leveraging unique properties for enhanced functionality [1]. Inspired by biological organisms, soft robots represent a paradigm shift in the way machines interact with their surroundings. Unlike traditional rigid robots, soft robots excel in tasks requiring gentle interactions and adaptability. Their deformable nature provides them with the unique ability to navigate through complex and constrained environments, making them invaluable in applications such as universal grippers [2], underwater navigation [3] and medical surgery [4].

Despite numerous proposed designs for soft robots, achieving precise and manageable regulation of the robot's configuration remains an open control challenge [5]. Soft robots, in contrast to their more conventional rigid counterparts, are categorized as having infinite degrees of freedom, rendering them inherently underactuated systems. Underactuation provides soft robots with a larger reachable configuration space at the same actuator cost and energy consumption compared to fully-actuated systems [6]. However, this property also complicates the modelling of these structures, often leading to scenarios where the system's control actions and outputs are not well-matched, which makes applying well-known robotic control concepts difficult to directly use [7]. This unique quality indicates that the time evolution of such continuum structures is governed by nonlinear partial differential equations and analytically integrating these is a formidable task [8].

While data-driven controllers have successfully tackled this control challenge by circumventing the necessity for intricate models [9], acquiring the extensive data required for a comprehensive control policy often proves impractical. Furthermore, these data-dependent mechanisms risk obsolescence when confronted with alterations in the robot's material properties, operational context, or environment. Consequently, attention shifts towards model-based control strategies, offering the potential to significantly enhance control performance in regulation tasks by leveraging our understanding of the system's dynamics. However, deploying these control policies necessitates a delicate balance between model accuracy and computational feasibility for real-time control [10]. The introduction of innovative reduced-order modelling techniques has played a pivotal role in bridging this gap, yielding a range of robust control strategies with proven stability [11].

Despite successful validation, most works focusing on model-based control of soft robots predominantly rely on

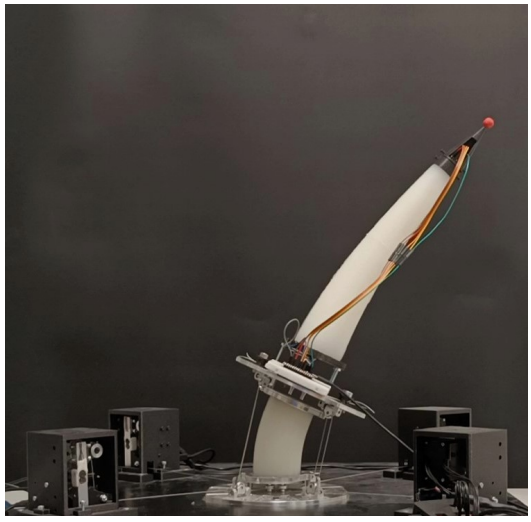


Fig. 1. Multi-segment soft robot testbench developed as part of this study.

fully-actuated approximations. These approximations simplify the control problem by assuming that each part of the robot can be independently controlled, neglecting the intrinsic underactuation characteristic of continuum structures. While effective in specific scenarios, such an approach might lead to suboptimal performance and overlook the unique characteristics that underactuation introduces. A notable instance where this aspect plays a significant role is in passively-actuated soft robots. These unconventional robotic systems, often devoid of traditional motor-driven actuators, rely on inherent material properties and external stimuli for motion. The absence of active actuation introduces novel challenges in control and regulation, as the robot's response is intricately linked to its passive dynamics [12]. Understanding and effectively harnessing these dynamics can unlock new frontiers for applications in environments where traditional robots may encounter limitations.

To address these challenges, a class of control laws explicitly using both actuated and unactuated degrees of freedom has been proposed and validated in simulation by Pustina et al. in [13] and [14]. However, these studies lack the experimental validation needed to empirically ascertain the robustness of the control architecture. Consequently, this work aims to explore the experimental facet of soft robot control by implementing a real-time model-based control strategy using the piecewise constant curvature assumption on the multi-segment soft robotic platform shown in Fig. 1, with a focus on the task of shape regulation. Our objective is to demonstrate on hardware setups that model-based control can significantly enhance shape regulation control performance compared to conventional model-free alternatives. Furthermore, we uniquely contribute by providing empirical proof of robustness for controllers that account for both actuated and unactuated degrees of freedom through experimentation on a (semi-)passively actuated soft robot setup. By addressing the distinct challenges posed by these systems, our goal is to contribute to the broader understanding of soft robotic control strategies and their applicability across a spectrum of robotic platforms.

The organization of the paper is detailed as follows. Following the introduction, the subsequent chapters delve into the core components of our research. A description of the mechanical, actuation, and sensory frameworks present in the soft robot platform is explained in Section II. Section III presents the mathematical modelling foundations, emphasizing the kinematic and dynamic formulations crucial to our control strategies. In Section IV, we detail model-based control systems, discussing their components and operational principles. Moving to Section V, we transition from theory to practice, outlining the methodologies and setups used for our real-world validations. Section VI unveils the outcomes of our experimental endeavours, offering insights into the performance of our control strategies. Finally, Section VII encapsulates the key takeaways and discusses the implications of our work.

TABLE I
ABBREVIATIONS USED IN THE TEXT

Abbreviation	Expansion
PCC	Piecewise Constant Curvature
PID	Proportional-Integral-Derivative
PLA	Polylactic Acid
CAD	Computer-Aided Design
SDK	Software Development Kit
IMU	Inertial Measurement Unit
PC	Personal Computer
S1	First Soft Segment
S2	Second Soft Segment
SV1	Single Segment Robot
SV2	Two Segment Robot
M1	Single Segment Simulation
M2	Two Segment Simulation

II. DEVELOPED SOFT ROBOT PLATFORM

The robot's body consists of lower and upper soft silicon segments, called S1 and S2 respectively, interfaced with several metal and PLA assemblies. The robot was manufactured using a combination of casting, moulding, and 3D printing techniques. Part of the robot's design was proposed and implemented by B. Deutschmann et. al in [15]. The various aspects of the robot's mechanical and electronic design are detailed below and illustrated in Fig. 2.

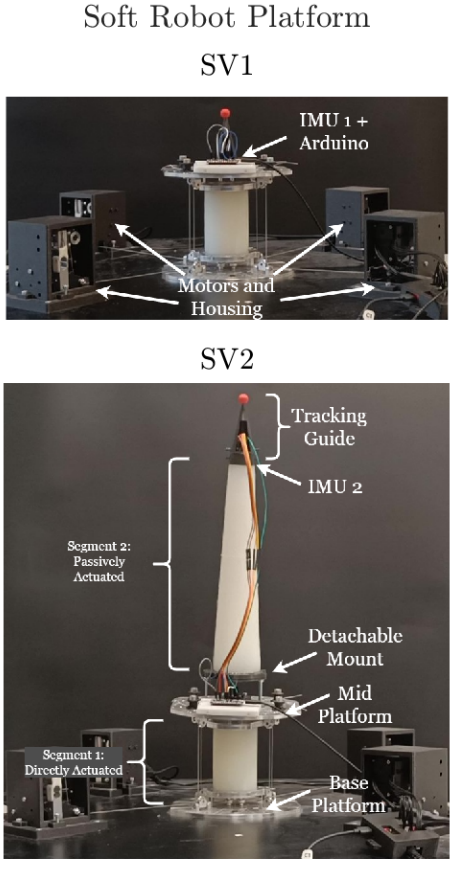


Fig. 2. Mechanical and electronic components of the soft robotic platform. The mount between S1 and S2 can be unfixed to convert between the SV1 and SV2 variants.

TABLE II
MATERIAL PARAMETERS FOR SOFT ROBOT SEGMENTS

Parameter	Units	S1	S2
Mass*	kg	0.543	0.639
Rest Length*	m	0.095	0.3
Radius*	m	0.0325	0.0325
Young's Modulus^{id}	Pa	3.67×10^5	2.01×10^5
Poisson's Ratio^{m}	–	0.3	0.3
Damping Coefficient^{id}	Ns/m	0.854	0.191

(*) measured, (^{id}) identified, (^{m}) manufacturer specification

A. Soft Segments

The soft bodies are produced using a mould-casting procedure, with CAD-designed moulds 3D printed in PLA. Subsequently, the bodies are cast in silicon, specifically Dragon Skin 30A and 10A for S1 and S2, respectively. Mechanical properties crucial for the dynamic model are outlined in the accompanying Table II. These parameters have been acquired through manual measurements, manufacturer specifications, and identification procedures using data from the robot.

Two rigid platforms manufactured are present at the base of the robot and between the two segments. These platforms provide attachment points in four quadrants for the tendons driving the robot. For less stiff soft elements the assembly could instead be 3D printed, but the current implementation requires a platform that can sustain the application of a large force without deformation or deterioration hence it has been manufactured in aluminium.

S2 is attached to the metal platform through a 3D-printed mount. This mount can easily be detached, thus making the setup modular and allowing tests to be carried out with either a single or two soft segments. For the remainder of this article, we will refer to the setup variation with a single directly actuated soft segment as SV1 and the two-segment setup variant including the passively actuated soft body as SV2. A small conical component has also been 3D-printed to serve as a discernable visual guide during the camera-recorded experiments.

B. Actuation

The actuation is provided through a four-tendon system that exerts a force at the central metal platform interfacing the two soft bodies. Each tendon can be tensioned using a dedicated Dynamixel® XH430-W350 coreless servo motor [16]. The motors are connected in a daisy-chained fashion with a U2D2 power hub that provides a central interface for serial communication. Custom software wrappers built around the Dynamixel® SDK provide a channel to issue current commands synchronously to the four motors based on the control action computed by the controller.

As a result of this actuation design, S1 experiences direct actuation, whereas S2 is only passively actuated. Such actuation that functions by virtue of the tilt of the platform, the body's own weight, and the elasticity of the segment is quite unconventional in robots. Passively actuated components can be deliberately added to soft robot designs to have systems that require less energy to navigate than conventional robots while morphing their shape based on the environment making them adaptable and safer to interact with. In our experimental setup, the passive segment serves as an exploration into whether model-based techniques can deliver precise control by taking into account the unactuated dynamics of the system.

C. Sensors

The robot is equipped with two Adafruit® BNO055 IMU sensors [17], one attached at the tip of each of the soft segments. The IMU outputs the orientation of the sensor in the quaternion format (ζ_w, ζ_x, ζ_y, and ζ_z). While other formats such as Euler angle notation can also be used, the quaternion method does not suffer from gimbal lock, indicating that there are no singularities when describing the orientation in this format. The orientation can be represented as a rotation matrix

$$R = \begin{bmatrix} \zeta_w^2 + \zeta_x^2 - \zeta_y^2 - \zeta_z^2 & 2(\zeta_x\zeta_y - \zeta_w\zeta_z) & 2(\zeta_x\zeta_z + \zeta_w\zeta_y) \\ 2(\zeta_x\zeta_y + \zeta_w\zeta_z) & \zeta_w^2 - \zeta_x^2 + \zeta_y^2 - \zeta_z^2 & 2(\zeta_y\zeta_z - \zeta_w\zeta_x) \\ 2(\zeta_x\zeta_z - \zeta_w\zeta_y) & 2(\zeta_y\zeta_z + \zeta_w\zeta_x) & \zeta_w^2 - \zeta_x^2 - \zeta_y^2 + \zeta_z^2 \end{bmatrix}. \tag{1}$$

When this matrix is equated with the kinematic model discussed in the following section, we can uniquely derive the curvature in both the x and y axes for a particular segment. As there are no absolute position sensors on the robot, we choose to rely on the tendon length measurements using the motor's position encoder as a more accurate method to evaluate the elongation or compression of the soft bodies. This method has been chosen rather than deriving this value from the acceleration estimates of the IMU due to the latter's tendency to be sensitive to noise. It also follows from this that we can only compute the change in length for S1 as we have no tendons attached to the tip of S2. However, considering the material properties used for the segment the elongation or compression is known to be significantly smaller than the curvature. Therefore, we have assumed that the elongation and compression of S2 is negligible for our robot. A complete overview highlighting the hardware interfaces and their interconnection is shown in Fig. 3.

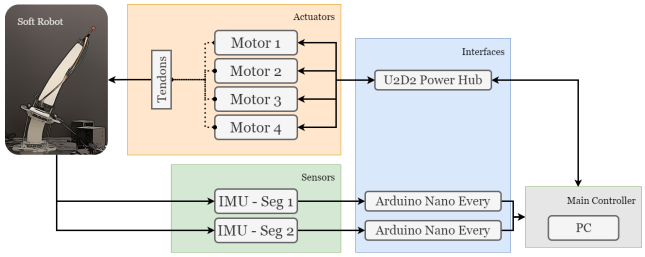


Fig. 3. Overview of the hardware and interfaces that make up the system architecture.

III. SYSTEM MODELLING

In this section, we explore the fundamental principles that serve as the backbone for the kinematic and dynamic modelling techniques applied in our investigation. A comprehensive grasp of these reduced-order modelling principles is

pivotal for devising efficient control strategies. It is important to acknowledge that several principles and mathematical notations in this section have been summarized as reporting them comprehensively is not within the scope of this article.

A. Kinematic Model

To characterize the kinematics of our soft robot, we employ the basic principles of the Cosserat rod theory, as outlined in [18] and [19]. According to this theory, the configuration space of any rod-like continuum body with a rest length L_0 is defined by the curve $g_c(\cdot) : s \rightarrow g_c(s) \in SE(3)$, where $s \in [0 \quad L_0] \subset \mathbb{R}$ represents the material abscissa. To describe the local motion and deformation, we use the strain vector $\xi = (g_c^{-1}\frac{\partial g_c}{\partial s})^\vee \in \mathbb{R}^6$. Here, the $(\cdot)^\vee$ operator maps an element from the Lie algebra to its corresponding vector space. This strain field comprises both linear and angular components.

An exact geometric description of the continuum body's configuration requires computing each strain component by solving a partial differential equation with respect to s. Unfortunately, this is computationally demanding, particularly for real-time control. To overcome this, we adopt the piecewise constant curvature (PCC) assumption, commonly used in soft robotics, making our kinematic formulation computationally tractable. This assumption views the continuum body as composed of a finite number of sequential arcs, each with constant curvature. Mathematically, this implies that the components of the curvature $k(s) \in \mathbb{R}^2$ within a single PCC segment are independent of s at every instant of time.

While the PCC assumption significantly simplifies the computational complexity of the model, it necessitates certain conditions to be met. The robot's structure must resemble a slender rod, with its length much larger than its other dimensions, ensuring that bending and elongation are the dominant modes of deformation. Additionally, each PCC segment must have linearly uniform material properties.

In the case of our soft robot testbench, we can observe that the conditions for the PCC assumption hold when considering S1 and S2 each as individual PCC segments. This division ensures that each segment resembles a homogeneous slender rod, and their deformations stay within limits that do not deviate significantly from the constant curvature hypothesis. Therefore, this modelling approach proves pragmatic and effective for our soft robot platform, balancing computational efficiency with an acceptable level of accuracy for real-time control.

A common set of generalized coordinates used to depict the configuration of a continuum body as a PCC segment is

$$q_\alpha = (\theta, \phi, \delta L) \in \mathbb{R}^3. \tag{2}$$

This parameterisation is sometimes referred to as the α-parameterisation and is illustrated in Fig. 4 . Here, θ is the angle of curvature, ϕ is the angle made by the plane in which the bending occurs, and δL is the change in length of the constant curvature segment with respect to L_0. These coordinates, while widely used, suffer from some shortcomings including the infinite ways to depict a straight configuration

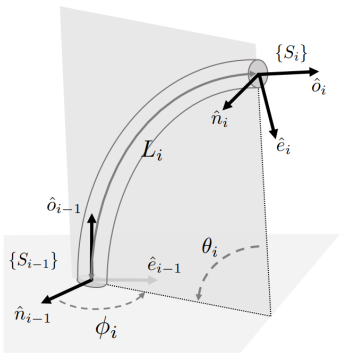


Fig. 4. Kinematic model showing a single segment with configuration characterised by the α parameterisation.

as the value of ϕ is no longer meaningful. Interested readers can refer to [20] for a more elaborate discussion on these limitations. An improved parameterisation that circumvents the issues observed in the α-parameterisation is given by

$$q_\Delta = (\Delta x, \Delta y, \delta L). \tag{3}$$

This is aptly called the Δ-parameterisation, and evaluates the configuration uniquely using two curvature coordinates Δx and Δy which are related to ϕ and θ by

$$\phi(q) = \arccos\left(\frac{\Delta x}{\Delta}\right) = \arcsin\left(\frac{\Delta y}{\Delta}\right), \quad \theta(q) = \Delta, \tag{4}$$

where

$$\Delta = \sqrt{\Delta x^2 + \Delta y^2}. \tag{5}$$

In the remainder of the paper, q always refers to q_Δ unless otherwise specified. It follows that the velocity and acceleration can be defined with the same notation as $\dot{q} = (\dot{\Delta}x, \dot{\Delta}y, \dot{\delta L})$ and $\ddot{q} = (\ddot{\Delta}x, \ddot{\Delta}y, \ddot{\delta L})$. Each PCC segment has two frames, one at the tip $\{S_i\}$ and one at the base $\{S_{i-1}\}$. The base frame is mapped onto the tip frame by means of a homogenous transformation matrix T^i_{i-1} consisting of rotation $R^i_{i-1} \in SO(3)$ and translation $tr^i_{i-1} \in \mathbb{R}^3$ parts. This transformation matrix is formulated as

$$T^i_{i-1} = \begin{bmatrix} R^i_{i-1} & tr^i_{i-1} \\ 0_3 & 1 \end{bmatrix}, \tag{6}$$

where

$$R^i_{i-1} = \begin{bmatrix} 1+\frac{\Delta_x^2}{\Delta^2}\left(\cos\left(\frac{\Delta s}{L0}\right)-1\right) & \frac{\Delta_x\Delta_y}{\Delta^2}\left(\cos\left(\frac{\Delta s}{L0}\right)-1\right) & \frac{\Delta_x}{\Delta}\sin\left(\frac{\Delta s}{L0}\right) \\ \frac{\Delta_x\Delta_y}{\Delta^2}\left(\cos\left(\frac{\Delta s}{L0}\right)-1\right) & 1+\frac{\Delta_y^2}{\Delta^2}\left(\cos\left(\frac{\Delta s}{L0}\right)-1\right) & \frac{\Delta_y}{\Delta}\sin\left(\frac{\Delta s}{L0}\right) \\ -\frac{\Delta_x}{\Delta}\sin\left(\frac{\Delta s}{L0}\right) & -\frac{\Delta_y}{\Delta}\sin\left(\frac{\Delta s}{L0}\right) & \cos\left(\frac{\Delta s}{L0}\right) \end{bmatrix},$$

$$tr^i_{i-1} = \frac{(L0+\delta L)}{\Delta^2}\begin{bmatrix} \Delta_x(1-\cos(\frac{\Delta s}{L0})) \\ \Delta_y(1-\cos(\frac{\Delta s}{L0})) \\ \Delta(\sin(\frac{\Delta s}{L0})) \end{bmatrix}.$$

B. Dynamic Model

We use the Euler-Lagrange approach to compute the quantities necessary to build the equation of motion of the system [21]. The Lagrangian $\mathcal{L}(q, \dot{q})$ of the system is computed as follows using the kinetic $\mathcal{T}(q, \dot{q})$ and potential $\mathcal{V}(q)$ energies

$$\mathcal{L}(q,\dot{q}) := \mathcal{T}(q,\dot{q}) - \mathcal{V}(q), \tag{7}$$

where

$$\begin{aligned}\mathcal{T}(q,\dot{q}) &= \int_0^l \left[\frac{1}{2}\rho A v^T v + \frac{1}{2}\omega^T \rho J \omega\right] ds,\\ \mathcal{V}(q) &= \int_0^l \Bigg[\frac{1}{2}(u-u^*)^T K_{bt}(u-u^*)\\ &\quad + \frac{1}{2}(w-w^*)^T K_{se}(w-w^*)\\ &\quad + \rho A g^T p\Bigg] ds.\end{aligned}$$

In the above equations ρ is the material density, A is the cross-sectional area, $g \in \mathbb{R}^3$ is the gravitational vector, and l is the total rest length of the body. The translation and angular velocities are given in the local frame by v and ω respectively. The curvature vector in the local frame is given by $u := (R^T R_s)^\vee$ where R_s is the partial derivative of the rotation matrix in terms of the arclength parameter and here the $(.)^\vee$ operator specifically maps $\mathfrak{so}(3)$ onto $\mathbb{R}^3$. u^* is its value at initialisation. Analogously, the rate of change of position with respect to arc length is given by $w = R^T tr_s$ where tr_s is the partial derivative of the kinematic translation vector with respect to the arclength parameter and w^* is its value at when the system is devoid of stress. $J \in \mathbb{R}^{3\times3}$ is the rotational inertial matrix

$$J = \begin{bmatrix} I_{xx} & 0 & 0\\ 0 & I_{yy} & 0\\ 0 & 0 & I_{zz}\end{bmatrix}, \tag{8}$$

where

$$\begin{aligned} I_{xx} = I_{yy} &= \frac{\pi r^4}{4},\\ I_{zz} &= I_{xx} + I_{yy}.\end{aligned}$$

Here, r represents the cross-sectional radius of the rod. The stiffness matrices are given by

$$K_{se} = \begin{bmatrix} G & 0 & 0\\ 0 & G & 0\\ 0 & 0 & E\end{bmatrix} A, \qquad K_{bt} = \begin{bmatrix} E & 0 & 0\\ 0 & E & 0\\ 0 & 0 & G\end{bmatrix} J, \tag{9}$$

where K_{se} and $K_{bt} \in \mathbb{R}^{3\times3}$ are the shear/extension and bending/torsion respectively. Here, the shear modulus G is

$$G = \frac{E}{2(1+v_p)}.$$

We have assumed that the soft bodies being modelled satisfy the condition for a diagonal stiffness matrix: a symmetric rod with linear isotropic material properties. Here, v_p is the Poisson's ratio and E is the Young's modulus of the material [22].

From the Lagrangian, we can obtain the dynamics of the system through the Euler-Lagrange equation

$$\frac{\partial}{\partial t}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}}\right)^T - \left(\frac{\partial \mathcal{L}}{\partial q}\right)^T = Q_{nc}, \tag{10}$$

where Q_{nc} denotes the generalised non-conservative forces on the body. This allows us to also derive the resultant dynamic equation of motion in the form

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + D(q)\dot{q} + N(q) = \tau(q), \tag{11}$$

where $M(q)$ is the mass matrix computed using the Lagrangian by

$$M(q) = \frac{\partial^2 \mathcal{L}}{\partial \dot{q}^2}. \tag{12}$$

$C(q,\dot{q})$ is the matrix consisting of the Coriolis and centrifugal terms. We have used the Christoffel symbols approach to compute this term.

$$C(q,\dot{q}) = [c_{ij}], \tag{13}$$

where

$$c_{ij} = \sum_{k=1}^{n} \Gamma_{ijk}\dot{q}^k,$$

and

$$\Gamma_{ijk} = \frac{1}{2}\left(\frac{\partial^2 \mathcal{L}}{\partial q^i \partial q^j}\frac{\partial q^k}{\partial t} + \frac{\partial^2 \mathcal{L}}{\partial q^i \partial q^k}\frac{\partial q^j}{\partial t} - \frac{\partial^2 \mathcal{L}}{\partial q^j \partial q^k}\frac{\partial q^i}{\partial t}\right).$$

$N(q)$ consists of the potential energy vector governing the system. It constitutes both the contributions from the gravitational force as well as the stiffness force.

$$N(q) = \left(\frac{\partial \mathcal{V}}{\partial q}\right)^T = G(q) + Kq. \tag{14}$$

$D(q)$ denotes the damping matrix of the system that is derived from the Kelvin-Voigt type viscous damping as elaborated in [23]. The term τ represents the generalised actuation force exerted on the system. We have assumed that there are no other external forces acting on the body apart from those discussed above. Based on this assumption we can formulate the steady-state equilibrium equation

$$\tau = N(q) = G(q) + Kq. \tag{15}$$

C. Modelling Underactuated Dynamics

The dynamic model detailed in the previous section uses generalised coordinates $q \in \mathbb{R}^n$ which makes the formulation universally applicable to several types of robotic systems. In the specific case of soft robot dynamics which are characterised by inherent underactuation, q consists of both actuated $q_a \in R^{n_a}$ and unactuated components $q_u \in R^{n-n_a}$, therefore $q = (q_a, q_u)$. In order to bypass the need to incorporate the unactuated component into the system dynamics, previous work often relies on the fully-actuated approximation $q \approx q_a$. However, this is a coarse approximation that does not allow us to effectively utilize the unactuated nature of the soft robot which makes for a less accurate model and consequently controllers that cannot account for the motion initiated by the passive dynamics of the system.

In [13] it has been shown that we can incorporate both actuated and unactuated components of q into (11) to derive the equation of motion

$$\begin{bmatrix} M_{aa} & M_{au} \\ M_{ua} & M_{uu} \end{bmatrix} \begin{bmatrix} \ddot{q}_a \\ \ddot{q}_u \end{bmatrix} + \begin{bmatrix} C_{aa} & C_{au} \\ C_{ua} & C_{uu} \end{bmatrix} \begin{bmatrix} \dot{q}_a \\ \dot{q}_u \end{bmatrix} + \begin{bmatrix} G_a \\ G_u \end{bmatrix} + \begin{bmatrix} K_{aa} & 0 \\ 0 & K_{uu} \end{bmatrix} \begin{bmatrix} q_a \\ q_u \end{bmatrix} + \begin{bmatrix} D_{aa} & D_{au} \\ D_{ua} & D_{uu} \end{bmatrix} \begin{bmatrix} \dot{q}_a \\ \dot{q}_u \end{bmatrix} = \begin{bmatrix} \tau \\ 0 \end{bmatrix}. \tag{16}$$

In the above equation, the components of the dynamic model have been further subdivided into elements corresponding exclusively to the actuated or unactuated variables as well as the contribution that arises from their coupling. Using this representation of the system we can deduce how the passive degrees of freedom affect the body's motion and as a result control laws that encapsulate this behaviour can be designed.

IV. Control Strategy

The control strategy has been formulated to fulfil the task of shape regulation. This involves providing a reference configuration $q_d \in \mathbb{R}^3$ for the soft segments to converge through the computed control action. To further this goal, we have utilised variants of model-based feedback regulators which are detailed in the following subsections.

A. Shape Regulation

Shape regulation involves converging an object's spatial configuration to a reference value. In shape regulation, we are not so interested in a specific time-parametrized transition of geometry as we are in accurately achieving a single desired shape over some period of time, which differentiates it from trajectory tracking. We can define this objective more formally through its error criteria as follows [5]:

$$\lim_{t\to\infty} q(t) - q_d = 0. \tag{17}$$

The error is computed purely based on the given spatial reference which remains independent with respect to time. Shape regulation finds primary importance in manipulation tasks involving soft end-effectors (or soft manipulators) for grasping objects with curvature or non-planar contours [24]. This is because we are usually more interested in the accuracy and repeatability of grasping rather than achieving it at a particular time slice. Another area where such control can be suitable is in insertion or navigation tasks where narrow crevices are involved [25]. In case we need to explore a pipe or probing vessels in the human body, deforming precisely to a given configuration is key and speed is a secondary criterion for success.

B. PD+ Controllers

In the realm of shape regulation, a comprehensive model significantly enhances the performance of PID-style regulators. Within this context, our controllers integrate two crucial model-based components: gravity cancellation and elasticity compensation.

Gravity cancellation leverages the body's current configuration to provide a counteractive control action to the gravitational force. This allows the remaining elements of the controller to operate as if the system is unaffected by gravity. Simultaneously, elasticity compensation is employed to partly counteract the effects of the elastic restoring force exerted by the soft bodies. Notably, unlike complete cancellation strategies, this compensation scheme employs the desired configuration rather than the current state of the soft body. This aims to harness the benefits of the body's elasticity for system stabilization without completely nullifying its effects, ensuring that the presence of compliance is not entirely cancelled. Using these methods we can formulate the control law for a model-based PD controller, referred to as the PD+ controller for the remainder of the paper, as

$$\tau_{PD+} = K_P(q_d - q_a) - K_D\dot{q}_a + G_a(q) + K_{aa}q_d. \tag{18}$$

By assimilating these model-based aspects, our controllers exhibit an interesting feature—they require smaller control gains to fulfil the control objective, resulting in less aggressive control actions and fewer oscillations during settling.

While this may provide ideal performance when testing in simulation, application on hardware implies that we are bound to experience mismatches between the physical body and its model. This discrepancy appears due to a number of factors, however, chief among them is the difficulty of accurately measuring parameters such as stiffness and damping from the system. The usual way to achieve good estimates of these parameters is through parameter identification using data collected from the testbench. As the data can never be fully comprehensive enough to account for all operational cases of the robot, model-body mismatches always persist to some extent. To account for this divergence from the model, we can include a compensatory integral term in our controller which can result in higher precision in reference tracking.

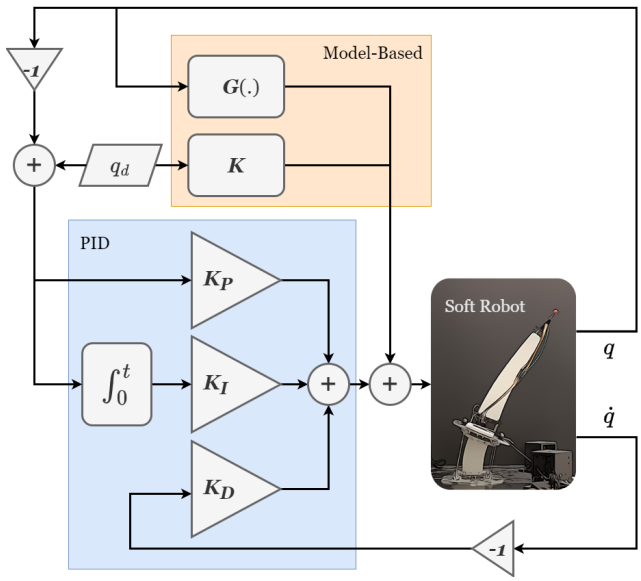


Fig. 5. Layout of PID+ control scheme. Removing the integral component converts it to the PD+ controller.

The resulting control law, called the PID+ controller henceforth, is

$$\begin{aligned}\tau_{PID+} = K_P(q_d - q_a) - K_D\dot{q_a} \\ +K_I\int_0^t [q_d - q(\rho)]\,d\rho \\ +G_a(q) + K_{aa}q_d.\end{aligned} \tag{19}$$

The overall control scheme using PID+ control is presented in Fig 5.

C. P-SatI-D Controllers

Incorporating an integral term demands careful consideration due to potential instabilities arising from couplings with soft robot dynamics. It has been shown that including a saturation function to the integral term results in control actions providing proven globally asymptotic stability to the system. It is also guaranteed that, regardless of initial conditions, the growth rate of the control torque will not become unbounded. These controllers also make the closed-loop robust to matched constant disturbances.

Interested readers can refer to [14] to see the conditions a function must meet to be utilized as a saturation function for this class of controllers. For the purposes of experimental verification, two saturation functions $s(y)$ are chosen: ($s_t(y) = \tanh y$) and ($s_p(y) = \frac{y}{(1+|y|^p)^{\frac{1}{p}}}$) where $p \in \mathbb{Z}^+$. The control law for the P-SatI-D controllers is

$$\begin{aligned}\tau_{PID^{s+}} = K_P(q_d - q_a) - K_D\dot{q_a} \\ +K_I\int_0^t s(q_d - q(\rho))d\rho \\ +G_a(q).\end{aligned} \tag{20}$$

D. Implemented Control Scheme

An overview of the developed control scheme is shown in Fig. 6. The IMU sensors collect information about the current orientation in quaternion format. The information from the sensors is converted into the desired coordinate system to represent the configuration of the body. The four motors provide information about the current position of the servo from which we can deduce the tendon length changes given that we are aware of their initial value. Using the dynamic model and the desired target state provided by the user, the control action is computed using the previously mentioned strategies.

This gives us the overall generalised force τ that needs to be exerted at the platform. However, this information is incompatible in its present form with the four-motor actuation system which takes commands in the form of individual motor currents. Therefore, we need to follow a conversion procedure to transform τ into a form that is actionable by the actuators.

The first step to evaluating this is to compute the tension that needs to be applied by each of the four tendons to achieve the resultant τ. These quantities are related by the equation

$$\begin{bmatrix}\tau_{\Delta x}\\ \tau_{\Delta y}\\ \tau_{\delta L}\end{bmatrix} = \begin{bmatrix}d & d & -d & -d\\ d & -d & d & -d\\ 1 & 1 & 1 & 1\end{bmatrix}\begin{bmatrix}F_1\\ F_2\\ F_3\\ F_4\end{bmatrix}. \tag{21}$$

Here, d refers to the distance between the centre of the platform and the point where the force is being applied. To solve this we formulate it as an optimization problem. We enforce that $F_m = \{F_1, F_2, F_3, F_4\} \in \mathbb{R}^+$ as a constraint to the problem, as all negative values for tendon force have an equivalent physical meaning to a tendon force of 0. The cost function is constructed to prioritize the force combinations with the lowest magnitudes. A single solution to the problem is identified by finding the minimal value of the cost function.

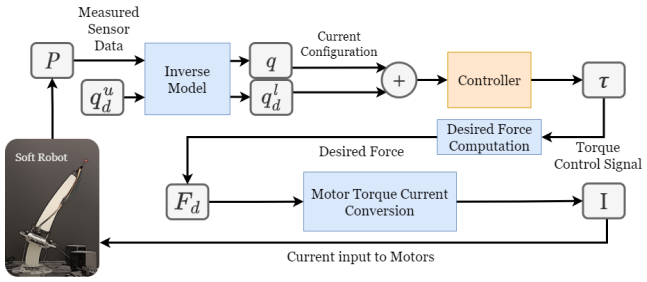


Fig. 6. Overview of implemented model-based control framework using feedback from the soft robot setup.

Following the computation of tendon forces, they can be converted to motor currents (C_m) by using the equations

$$\tau_m = F_m d_{ts}, \tag{22}$$

$$C_m = \frac{\tau_m P_{ct}}{C_{unit}}. \tag{23}$$

The term d_{ts} refers to the radius of the gear on the motor's shaft, P_{ct} is the current torque conversion factor obtained in the region of interest from the manufacturer-provided performance data. One possible drawback of the current arrangement could be that the tendon winds over itself when the motor rotates to tension it, effectively altering the value of d_{ts}. It should be noted that we have assumed that the tendon-motor actuation structures are identical for the four actuators. Furthermore, as there is no additional feedback control implemented on the force applied by the motor, we have assumed that the current-torque profiling provided by the manufacturer is accurate within the robot's operational range and that the motor achieves a commanded current in negligible time after its issue.

The implemented controllers feature a deadband in the vicinity of the setpoint, serving to mitigate the impact of small disturbances and noise around the desired state, preventing unnecessary and frequent minor adjustments. This deadband introduces a level of tolerance to enhance system stability and reduce wear on mechanical components. Additionally, controllers with integral action incorporate an integral reset mechanism at zero-crossing, addressing the issue of integral

windup. This reset occurs when the error changes sign, preventing the integral term from accumulating excessively during sustained disturbances. The integral reset at zero-crossing ensures improved responsiveness and stability in the presence of dynamic changes, contributing to enhanced overall controller performance.

Algorithm 1 Tuning

```
Initialize K_P, K_I, K_D
while no sustained oscillations do
    Increase K_P
end while
Record the critical proportional gain: K_u = K_P
Set K_P = 0.6 × K_u
while system is not critically damped do
    Increase K_D
    if system becomes overdamped then
        Set K_D ← 0.8 × K_D
    end if
end while
while steady-state error is not zero/minimal do
    Increase K_I
    if overshoot observed then
        Set K_I ← 0.8 × K_I
    end if
end while
```

In the process of algorithm development for tuning the controllers, our primary focus lies on optimizing steady-state error, given its significant impact on achieving precise control outcomes. Following closely is the emphasis on minimizing settling time, aiming for a swift response to input changes. Recognizing the system's slower recovery from overshoot, our algorithm places a deliberate effort to prevent excessive overshooting, prioritizing fast settling over rapid responses. This consideration also stems from the observed tendency of velocity-induced oscillations, especially in the passive segment, which lacks sufficient damping. Prioritizing gradual smooth settling seeks to mitigate these oscillations in the system during operation. With these objectives in mind a general framework, based loosely on the Ziegler-Nichols algorithm was developed as shown in Algorithm 1.

V. Simulations and Experiments

A. Software Simulation

Using the model parameters detailed in Table II, we have simulated the behaviour of the robot in both SV1 and SV2 variants. The simulation model for the two setup variants will henceforth be referred to as M1 and M2. Each body is modelled as a single PCC segment and validated for a number of shape regulation tasks. For the M1 variant, all the degrees of freedom considered by the PCC kinematic formulation are actuated $q = q_a$, where q_a is given by

$$q_a = (\Delta x_{S1}, \Delta y_{S1}, \delta L_{S1}). \tag{24}$$

In the case of the M2 variant, we have also the kinematics of S2 which are unactuated or passively actuated based on the design of the robot. This implies that $q = (q_a, q_u)$, which can be written in expanded form

$$q_u = (\Delta x_{S2}, \Delta y_{S2}, \delta L_{S2}), \tag{25}$$

$$q = (\Delta x_{S1}, \Delta y_{S1}, \delta L_{S1}, \Delta x_{S2}, \Delta y_{S2}, \delta L_{S2}). \tag{26}$$

It is evident that by structuring the system's kinematics in this way, we can simulate the contribution of the unactuated degrees of freedom on the motion of the bodies and also use the generated passive dynamics in the controllers described in Section V.

The simulation uses the ode15s solver in the MATLAB Simulink environment and shows tractable performance when processed on a standard PC. While we follow the same general framework for tuning the controllers as described in Algorithm 1, the absence of detrimental real-world effects associated with high gains, such as sluggish recovery from overshoot, affords us greater flexibility in tuning.

The gains determined through the software simulation tuning process serve as valuable initialization parameters for the hardware setup. Additionally, the simulations offer a compelling visualization through 3D rendering, depicting the current configuration of the centerline. This visualization aids in a more nuanced understanding of the robot's performance characteristics in various configurations.

B. Experimental Setup

In order to utilise our model-based controllers on the physical setup, it is necessary to use a mix of measured and identified data to build a realistic profile of the robot. The measured and manufacturer-specified parameters have been highlighted previously in Table II. In addition to this, we have used the experimental rig to obtain data to identify the coefficients of the stiffness matrix and the damping coefficients. Though the stiffness matrix can be obtained from Young's modulus and Poisson's ratio of the material which are specified by the manufacturer, these values are most likely to contribute to model errors due to material degradation over time and individual stiffness variance based on the shape used for casting.

To identify these parameters we have used least-squares data fitting technique, which iteratively updates the parameter values until the squared error between the model's behaviour and the measured behaviour of the robot is minimal. In the first stage of this process, we have used the equilibrium equation from the dynamics of the robot to identify the coefficients of the stiffness matrix. Following this, the damping parameters are identified from the usual equation of motion mentioned in Section III-B. The controller framework runs at a frequency of 100Hz to allow sufficient time to read and write data over the serial channel of the sensors and motors. The kinematic formulation used to compute the model and the model-based components of the controller is consistent with the notation in (24) to (26).

TABLE III
EXPERIMENT LOG

Experiment ID	Setup Variant Used	Controllers Used						No. of Experiments per Controller
		PD	PID	PD+	PID+	P-SatI-D s_p	P-SatI-D s_t	
SI	M1	✓	✓	✓	✓	✓	✓	1
	M2	✓	✓	✓	✓	✓	✓	1
	SV1	✓	✓	✓	✓	✓	✓	1
	SV2	✓	✓	✓	✓	✓	✓	1
M2D	M1	–	✓	–	✓	–	✓	1
	M2	–	✓	–	✓	–	✓	1
	SV1	–	✓	–	✓	–	✓	1
	SV2	–	✓	–	✓	–	✓	1
M3D	M1	–	✓	–	✓	–	✓	1
	M2	–	✓	–	✓	–	✓	1
	SV1	–	✓	–	✓	–	✓	1
	SV2	–	✓	–	✓	–	✓	1
MG	SV1	✓	✓	✓	✓	✓	✓	5
	SV2	✓	✓	✓	✓	✓	✓	7
VP	SV1	✓	✓	✓	✓	✓	✓	9
	SV2	✓	✓	✓	✓	✓	✓	9
VI	SV1	–	✓	–	✓	✓	✓	9
	SV2	–	✓	–	✓	✓	✓	9
Vp	SV1	–	–	–	–	✓	–	6

C. Experiment Design

The main objective of this article is experimental validation of the model-based controllers for shape regulation. In order to obtain a comprehensive evaluation of how the controller performs it is necessary to subject the system to a diverse array of experimental tests.

In addition to obtaining the performance for tracking a single configuration objective, we have experimented with tracking multiple shapes sequentially both planar and three-dimensional with respect to the tip of the robot. Furthermore, the versatility of the controllers after tuning is tested by varying the reference without re-tuning as well as tests where the proportional and integral gains are iteratively varied from the tuned state. These tests are carried out for both SV1 and SV2 variants of the robot. The complete list of experiments carried out is shown in Table III.

A description of the experiments corresponding to each experiment ID given in the table is as follows:

- **SI**: This experiment involves tracking a single configuration reference. For this experiment, we measure the indicative performance parameters commonly used to validate controllers. Specifically, we document the settling time, peak time, steady state error percentage, and overshoot percentage for each of the six controllers.
- **M2D**: For controllers to be deemed suitable for practical use, they must demonstrate the capability to sequentially track multiple shape regulation objectives in all areas of their workspace. This set of experiments specifically evaluates the tracking precision of controllers when presented with multiple planar objectives sequentially. To ensure clarity and efficiency in documentation, these experiments are exclusively presented for the PID, PID+, and P-SatI-D s_t controllers: a representative subset of controllers used in this article.
- **M3D**: This experiment has the same objective as M2D but with configuration references that are three-dimensional relative to one another covering a larger section of the robot's workspace. In the camera-recorded experiments for M2D and M3D, position tracking of the spherical guide at the robot's tip is achieved using the normalized cross-correlation coefficient method (TM_CCOEFF_NORMED) in OpenCV.
- **MG**: This experiment aims to assess the controller's robustness to variations in shape regulation goals without retuning. This objective is particularly demanding, as it tests the controller's ability to adapt to changes in the operating point, considering model mismatches. This makes it a valuable validation experiment. In the SV1 variant, controllers were initially tuned to track the configuration $q = (0.2, 0, 0)$. For the S2 variant, the tuning objective was set as $q_u = (0.3, 0, 0)$. Additional shape regulation goals were chosen to maximize coverage of the robot's workspace without causing damage to the hardware.
- **VP**: In this experiment, the assessment of the controller's performance involves the variation of the proportional gain from the tuned state, ranging between -20% and 20% of the tuned K_p, with the baseline state defined as 0%.
- **VI**: In this experiment, the evaluation of the controller's performance involves varying the integral gain from the tuned state within a range of -20% to 20% of the tuned K_i, with the baseline state considered at 0%. As previously highlighted, the careful addition and adjustment of an integral component to the controller are essential

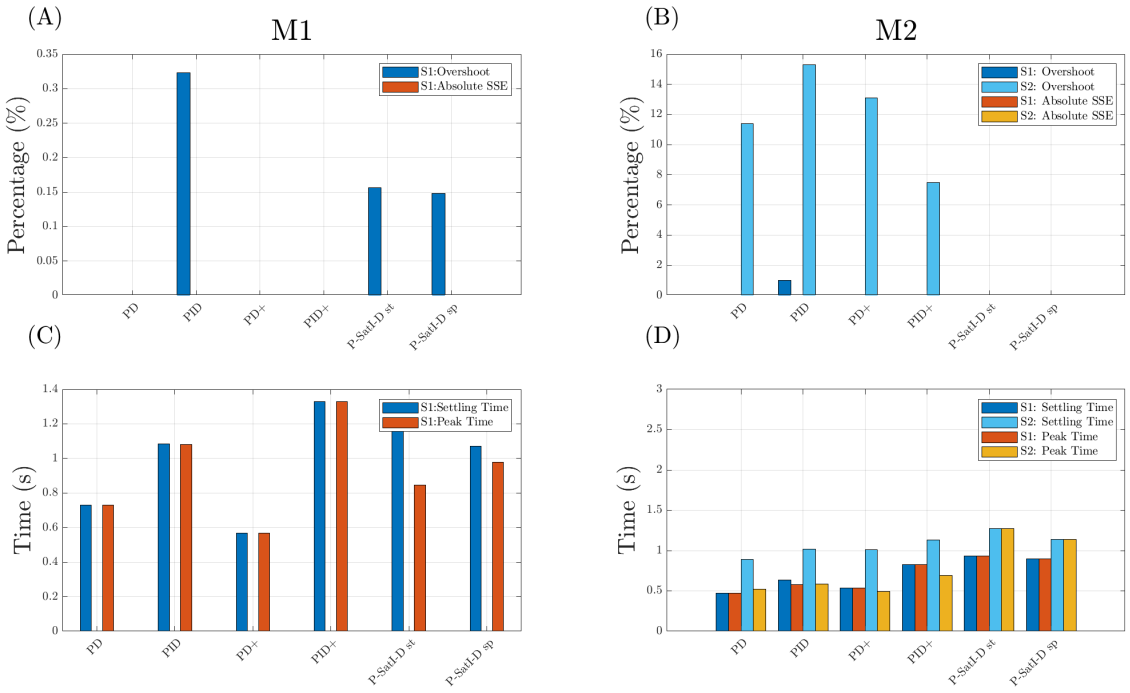


Fig. 7. Results for tracking a single reference configuration in simulation for both M1 and M2 variants. (A) and (B) show the overshoot and steady-state error observed with the different controllers in the M1 and M2 variants respectively. (C) and (D) present the settling and peak times for the same experiments. Overall, the graph provides a side-by-side comparison of the controllers' performance in simulation.

due to the potential to adversely affect the controller's performance. Consequently, this experiment serves as a valuable indicator of the controllers' robustness.

- **Vp**: In the investigation of the impact of the p parameter on P-SatI-D s_p controllers, an additional tuning consideration becomes apparent. A well-tuned p parameter provides the ability to precisely adjust the controllers' behavior, contributing to improved overall performance. This experiment specifically examines how altering the p value influences both the controller's behavior and the saturation function. The p value, being an integer, has been systematically explored across a range of values, including small values at the minimum end of the spectrum as well as values an order of magnitude larger.

VI. Results and Discussion

In this section, we will validate the developed controllers for a set of different shape regulation objectives as per the constructed experiment design. Beyond the controllers detailed in Section IV, we extend our investigation to include a model-free PD and PID controller, serving as a baseline for comparison. This additional exploration helps elucidate whether the introduction of model-based components results in observable performance enhancements. Furthermore, we will discuss the key insights obtained from these results. The section is organised based on experiment categories and followed by the current limitations and future scope.

A. SI: Single Shape Regulation

The simulation results from the simulation (M1 and M2) are shown in Fig. 7 and in Appendix A and B. The simulation results reveal perfect setpoint tracking for all controllers except the PD which has a very small error as expected from testing on an ideal model with no uncertainty. The PD and PD+ require significantly higher proportional gains to achieve their best possible performance, likely due to the lack of a compensating integral term and the prioritisation of fast settling. Comparing segment performances in the M2 variant, S2 consistently exhibits slower settling than S1, with an average difference of 0.3595 seconds attributed to the inertia of the passively actuated S2.

The experimental results from SV1 and SV2 are given in Figure 8 and in Appendix C, D, F, and G. Analyzing both simulation and hardware results, the average settling time in M1 is 61.6% less than that in SV1. This discrepancy is ascribed to the model's omission of velocity constraints inherent in the motor, where the motor's maximum velocity of 32 rpm sets a theoretical limit on the time it takes to achieve specific configurations.

In the evaluation of controllers on the SV1 variant, the PID+ controller incurs the least steady-state error at 0.208%, albeit with the slowest settling time. Conversely, the PD+ controller settles approximately 1.066 seconds faster but with a relatively higher steady-state error of 3.49%. A comparison between model-based PD+ and baseline PD controllers reveals that due to high gains used by the PD controller, aggressive

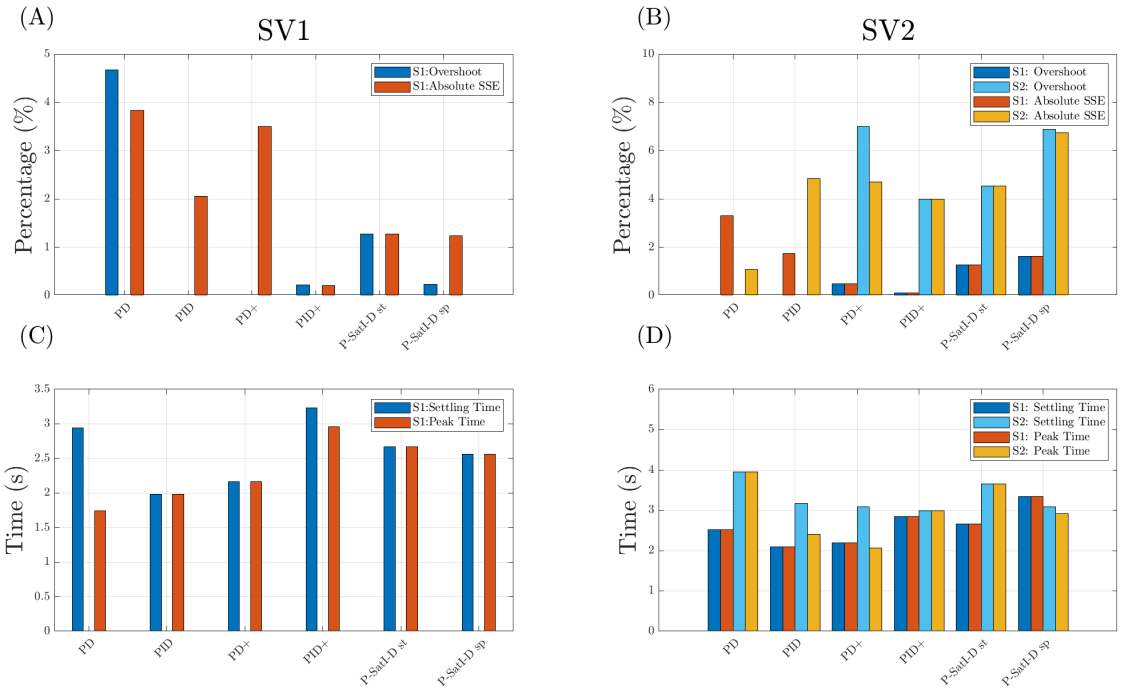


Fig. 8. Results for tracking a single reference configuration on the soft robot platform for both SV1 and SV2 variants. (A) and (B) show the overshoot and steady-state error observed with the different controllers in the SV1 and SV2 variants respectively. (C) and (D) present the settling and peak times for the same experiments. Overall, the graph provides a side-by-side comparison of the controllers' performance on the two robot variants.

control actions often lead to overshoots, increasing settling time by 35.8%. The PID controller partially mitigates this issue, showing improvements in settling time and steady-state error due to the compensating integral term. However, without model-based components, tracking performance is significantly poorer compared to the PID+ controller.

Comparing SV1 and SV2 results, a noticeable 28.21% increase in settling time in the latter is observed. This increase is attributed to reduced damping in S2, causing significant oscillations under sudden acceleration or deceleration and subsequently increasing settling time. While the introduction of passive dynamics is expected to decrease tracking accuracy, it is noteworthy that model-based controllers maintain a steady error below 10%, validating their performance. Further improvements in controlling the passive segment are anticipated with a more comprehensive and better-identified model. Introducing a control scheme with unactuated coordinates yields interesting instances where the accuracy of S1's regulation performance may not proportionally match the tracking precision of S2. In SV2, for instance, the PD controller exhibits a steady-state error approximately 30 times larger than the PID+ controller for tracking the reference for S1, while the reverse is observed for the tracking performance of S2, where the error is close to 4 times larger for the PID+ controller than the PD controller.

B. M2D and M3D: Planar and 3D Shape Tracking

The tracking behaviour can be seen qualitatively in Figures 9, 10, 11, 12 for M1, M2, SV1, and SV2 respectively. Across all hardware setup variations involving the addition of multiple targets, there is a substantial increase in steady state error. This phenomenon is likely attributed to the additional torque required to counteract the friction of the remaining tendons.

Comparing the baseline PID controller with its model-based counterparts reveals that the latter exhibits more graceful convergence. This behaviour is attributed to the incorporation of gravity-cancellation and elasticity compensation, reducing the need for high gains that typically induce aggressive control actions. Due to overshoots observed in the baseline PID controller, the system takes significantly longer to settle, resulting in a steady state error percentage that is one order of magnitude larger than the best-performing model-based controller. The disparities in settling behaviour become more pronounced in the 3D case, where the model-free controller settles below the target, either after an overshoot or when gains are kept conservative.

These experiments further validate the versatility of the developed model-based controllers, showcasing their ability to accurately track targets in various areas of the robot's workspace. Sufficiently accurate regulation is observed even in the presence of an unconventional passively actuated segment and discrepancies between the model and the physical robot.

C. MG: Shape Regulation Goal Variation

The experimental results can be observed in the third column of Appendix H for SV1 and in Appendix I for SV2. Analyzing the baseline PD and PID controllers reveals that

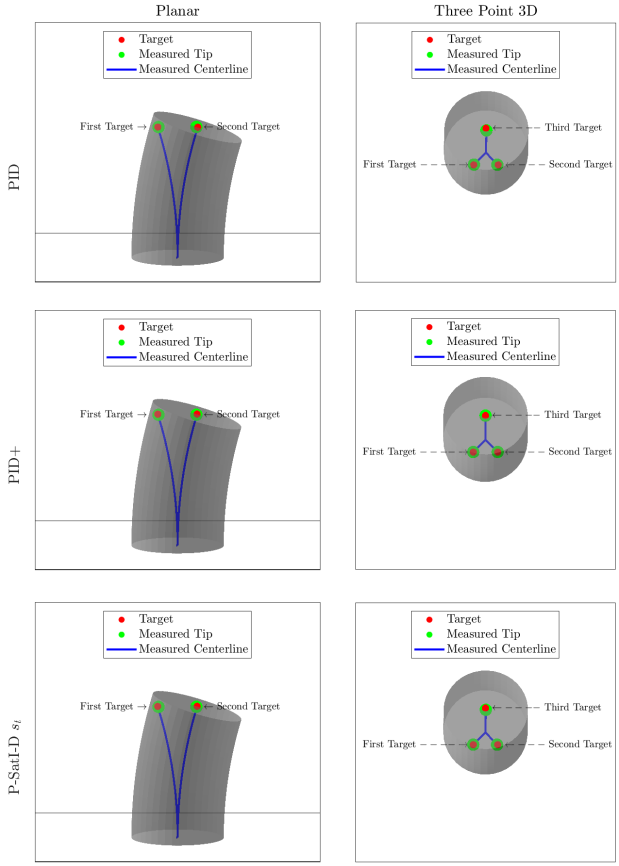


Fig. 9. Results for tracking multiple three-dimensional and planar references sequentially in the M1 simulation. Each row depicts the results for a single type of controller.

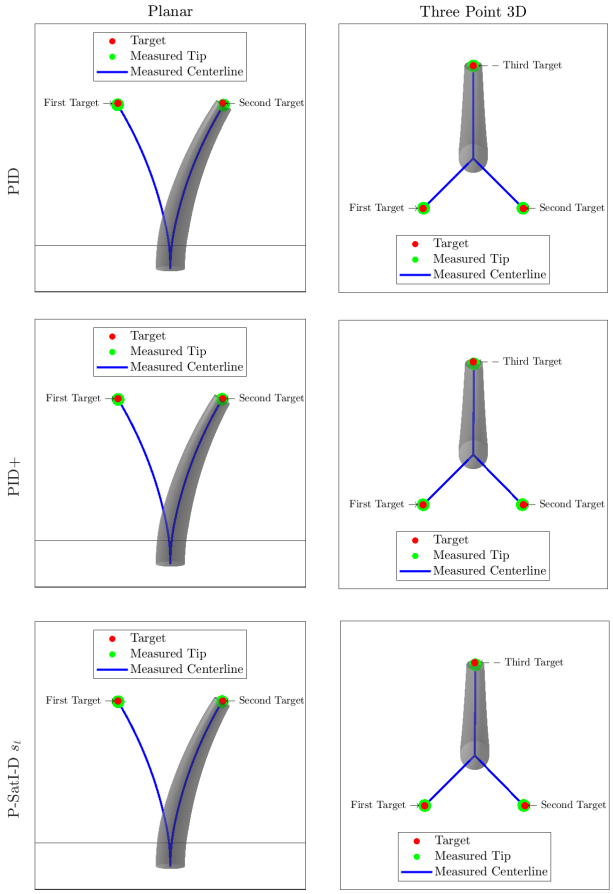


Fig. 10. Results for tracking multiple three-dimensional and planar references sequentially in the M2 simulation. Each row depicts the results for a single type of controller.

changes in reference have a significant impact on tracking accuracy. This effect is pronounced in the PID controller, where tracking higher reference goals leads to a drastic increase in settling time due to overshoots, and the steady state accuracy exceeds the 10% error limits.

The PD+ controller exhibits relatively better performance than the model-free controllers; however, at the extremes of the test set, overshooting results in slow settling and high error percentages. Similarly, the PID+ controller follows a comparable trend, but notably, the tracking performance never exceeds a 10% error.

Both variations of the P-SatI-D controller demonstrate robustness to changes in shape regulation goals. Even at the extreme ends of the spectrum, the configuration tracking error remains below 8% in all cases, maintaining reasonable settling times. These results underscore the adaptability and stability of the P-SatI-D controller across varying shape regulation objectives, making it a promising choice for scenarios with dynamic changes in operating points.

D. VP: Proportional Gain Variation

The experimental results can be observed in the first column of Appendix H for SV1 and in Appendix I for SV2. Among the controllers under examination, the PD+ controller demonstrates notable robustness to variations in proportional gain, being the only one to maintain steady state errors consistently within the 10% threshold. However, it can be noted that the variation in proportional gain negatively impacts the settling time in the PD+ controller.

Conversely, both the PID+ and P-SatI-D s_p controllers exhibit consistently low settling times throughout the experiment. Nonetheless, at the extreme ends of the proportional gain variation, there is a noticeable departure from the 10% threshold for steady state error.

As expected, the PD controller performs less favorably in this experiment, likely attributed to its high reliance on proportional gain and the absence of model-based components capable of adapting to varying gains. The inferior performance of the PD controller further emphasizes the importance of incorporating model-based elements to enhance adaptability within a larger spectrum of gain values.

E. VI: Integral Gain Variation

The experimental results can be observed in the second column of Appendix H for SV1 and in Appendix I for SV2.

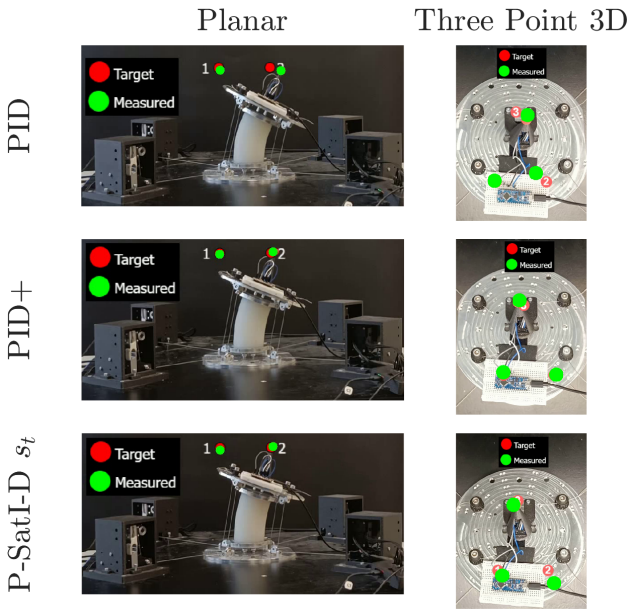


Fig. 11. Experimental results for the SV1 variant of the platform highlighting steady-state accuracy of various controllers in reaching desired planar and three-dimensional configurations.

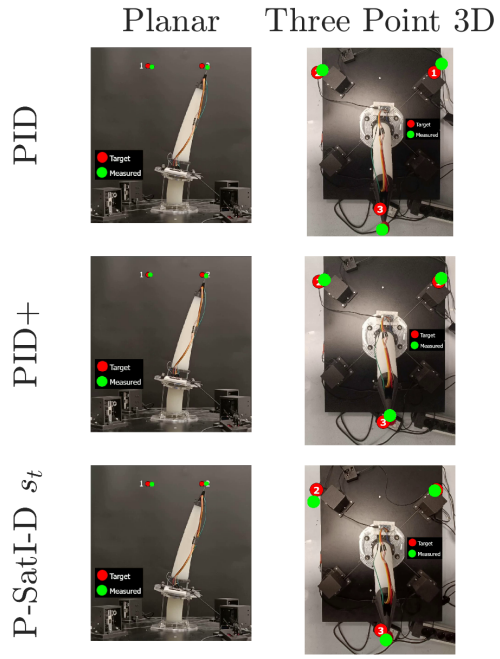


Fig. 12. Experimental results for the SV2 variant of the platform highlighting steady-state accuracy of various controllers in reaching desired planar and three-dimensional configurations.

The P-SatI-D controllers exhibit notable advantages in terms of speed and accuracy compared to the PID and PID+ controllers. Quantitatively, there is a considerable improvement, with a 59.62% reduction in the average settling time observed across all levels of the integral gain for the P-SatI-D s_t controller when compared to the PID+ controller. In assessing accuracy, it is evident that even under the highest levels of variation, the worst steady state error for the P-SatI-D s_p controller remains at 9.14%. In stark contrast, both the PID and PID+ controllers surpass the 10% error limit when subjected to a 20% variation in the integral gain.

These findings underscore the enhanced performance and robustness of the P-SatI-D controllers, particularly in maintaining lower settling times and minimizing steady-state errors compared to their unsaturated integral counterparts in the face of integral gain variations.

F. Vp: Saturation Function Parameter Variation

The results of the experiment can be seen in Appendix E with the tracking performance in the first row and the variation of the saturation functions in the second row. The findings indicate that lower p values result in a more gradual settling response akin to critical damping, aligning with the smoother characteristics inherent in lower p polynomial saturation functions. Conversely, as p values increase, saturation functions exhibit increased jaggedness, with the distinctions between functions becoming less pronounced when $p > 10$. Notably, optimal performance is observed at a p value of 2, emphasizing the significant influence of the parameter on the controllers' behaviour.

G. Current Limitations and Avenues for Future Research

An assumption inherent in our work is the absence of any mismatch between the motor's performance and the current-torque relation as provided by the manufacturer. Introducing an embedded sensor capable of measuring the applied force would address this limitation, offering a more refined and accurate execution of control actions. This improvement becomes particularly crucial for scenarios where precise force applications are imperative for the soft robot's intended functionality.

The servo motors employed in our study feature a discrete task space for control commands, with the finest increments of 2.69 mA. Consequently, the control action is mapped onto a coarsely resolved command space, imposing limitations on the minimum achievable steady-state error. Exploring advanced motor control techniques or transitioning to motors with finer resolution could mitigate this limitation and enhance the system's overall precision.

This work focuses on a single soft robot platform, therefore validation has only been done for soft bodies of a specific stiffness profile. Future work could include validation of these model-based controllers on soft robots with different levels of compliance where the passive dynamics may be much more or less prominent.

In the current configuration, real-time robot positioning relies on IMU-derived orientation. However, these sensors may exhibit drift over prolonged usage periods and provide only relative orientation characteristics. To address this, incorporating absolute sensor mechanisms, such as motion capture technology, could offer a more accurate estimation of the robot's configuration, allowing for precise monitoring of the elongation or compression of the soft robotic body.

VII. Conclusion

In conclusion, this study provides compelling evidence for the efficacy and adaptability of model-based controllers in the shape regulation of soft robots. We have contributed with a specially designed soft robot research platform that allows the utilisation and validation of controllers using both actuated and unactuated degrees of freedom, accounting for the underactuated nature of the robot. Furthermore, through a series of meticulously designed experiments, we have not only validated the theoretical foundations of model-based control but have also demonstrated its practical applicability in real-world scenarios. The superior performance observed in achieving precise and controlled deformations showcases the potential of these controllers to advance the field of soft robotics for various applications demanding dexterity.

A distinctive aspect of our experimental validation is the successful application of model-based controllers to passively actuated soft segments—an unconventional and challenging class of soft robots. This extension of our research underscores the versatility of the proposed control techniques that account for underactuation, as they prove effective in managing complex and less intuitive robotic configurations. The ability to navigate the unique challenges posed by passively actuated structures speaks to the robustness of our approach and expands the scope of model-based shape regulation into unconventional domains.

Additionally, the strategic use of reduced order modelling methods in our control strategy deserves particular emphasis. Opting for reduced order models over full order models ensures computational efficiency facilitating a real-time control implementation. This choice strikes a delicate balance between computational speed and accuracy, addressing the critical need for responsiveness and adaptability in practical applications. When coupled with sophisticated control techniques, such as nonlinear integral action, we are able to obtain frameworks with high precision that are general enough to be applied across a diverse range of soft robotic platforms with multiple segments, marking a significant step forward in the evolution of soft robotic control systems.

References

[1] C. Della Santina, M. Catalano, and A. Bicchi, "Soft robots," 02 2021.

[2] R. L. Truby, R. K. Katzschmann, J. A. Lewis, and D. Rus, "Soft robotic fingers with embedded ionogel sensors and discrete actuation modes for somatosensitive manipulation," *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*, pp. 322–329, 2019.

[3] R. K. Katzschmann, J. DelPreto, R. MacCurdy, and D. Rus, "Exploration of underwater life with an acoustically controlled soft robotic fish," *Science Robotics*, vol. 3, no. 16, p. eaar3449, 2018.

[4] M. Runciman, A. Darzi, and G. P. Mylonas, "Soft robotics in minimally invasive surgery," *Soft Robotics*, vol. 6, no. 4, pp. 423–443, 2019. PMID: 30920355.

[5] C. Della Santina, C. Duriez, and D. Rus, "Model-based control of soft robots: A survey of the state of the art and open challenges," *IEEE Control Systems Magazine*, vol. 43, no. 3, pp. 30–65, 2023.

[6] B. He, S. Wang, and Y. Liu, "Underactuated robotics: A review," *International Journal of Advanced Robotic Systems*, vol. 16, no. 4, p. 1729881419862164, 2019.

[7] M. Keppler, C. Ott, and A. Albu-Schäffer, "From underactuation to quasi-full actuation: Aiming at a unifying control framework for articulated soft robots," *International Journal of Robust and Nonlinear Control*, vol. 32, no. 9, pp. 5453–5484, 2022.

[8] C. Armanini, F. Boyer, A. T. Mathew, C. Duriez, and F. Renda, "Soft robots modeling: A structured overview," *IEEE Transactions on Robotics*, 2021.

[9] X. Wang, Y. Li, and K.-W. Kwok, "A survey for machine learning-based control of continuum robots," *Frontiers in Robotics and AI*, vol. 8, 2021.

[10] D. Bruder, C. D. Remy, and R. Vasudevan, "Nonlinear system identification of soft robot dynamics using koopman operator theory," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 6244–6250, 2019.

[11] R. K. Katzschmann, C. D. Santina, Y. Toshimitsu, A. Bicchi, and D. Rus, "Dynamic motion control of multi-segment soft robots using piecewise constant curvature matched with an augmented rigid body model," in *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*, pp. 454–461, 2019.

[12] C. Della Santina, R. K. Katzschmann, A. Biechi, and D. Rus, "Dynamic control of soft robots interacting with the environment," in *2018 IEEE International Conference on Soft Robotics (RoboSoft)*, pp. 46–53, 2018.

[13] P. Pustina, C. D. Santina, and A. De Luca, "Feedback regulation of elastically decoupled underactuated soft robots," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4512–4519, 2022.

[14] P. Pustina, P. Borja, C. D. Santina, and A. De Luca, "P-sati-d shape regulation of soft robots," *IEEE Robotics and Automation Letters*, vol. 8, no. 1, pp. 1–8, 2023.

[15] B. Deutschmann, J. Reinecke, and A. Dietrich, "Open source tendon-driven continuum mechanism: A platform for research in soft robotics," in *2022 IEEE 5th International Conference on Soft Robotics (RoboSoft)*, pp. 54–61, 2022.

[16] Robotis, "Xh430-w350." https://emanual.robotis.com/docs/en/dxl/x/xh430-w350/.

[17] Adafruit, "Adafruit bno055 absolute orientation sensor." https://learn.adafruit.com/adafruit-bno055-absolute-orientation-sensor.

[18] F. Renda, M. Cianchetti, H. Abidi, J. Dias, and L. Seneviratne, "Screw-Based Modeling of Soft Manipulators With Tendon and Fluidic Actuation," *Journal of Mechanisms and Robotics*, vol. 9, p. 041012, 05 2017.

[19] F. Boyer, V. Lebastard, F. Candelier, and F. Renda, "Dynamics of continuum and soft robots: A strain parameterization based approach," *IEEE Transactions on Robotics*, vol. 37, no. 3, pp. 847–863, 2021.

[20] C. Della Santina, A. Bicchi, and D. Rus, "On an improved state parametrization for soft robots with piecewise constant curvature and its use in model based control," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1001–1008, 2020.

[21] B. J. Caasenbrood, A. Y. Pogromsky, and H. Nijmeijer, "Dynamic modeling of hyper-elastic soft robots using spatial curves," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 9238–9243, 2020. 21st IFAC World Congress.

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

[23] J. Linn, H. Lang, and A. Tuganov, "Geometrically exact cosserat rods with kelvin-voigt type viscous damping," vol. 4, 05 2012.

[24] M. Modabberifar and M. Spenko, "A shape memory alloy-actuated gecko-inspired robotic gripper," *Sensors and Actuators A: Physical*, vol. 276, pp. 76–82, 2018.

[25] X. Zhang, T. Pan, H. Heung, P. Chiu, and Z. Li, "A biomimetic soft robot for inspecting pipeline with significant diameter variation," pp. 7486–7491, 10 2018.

APPENDIX

A. M1 Controller Results

TABLE IV
CONTROLLER RESULTS M1

Controller	K_P	K_D	K_I	p	Steady State Error (%)	Settling Time (s)	Overshoot (%)	Peak Time (s)
PD	32	3	0	-	0.0563	0.73	0	0.73
PID	16	2	11	-	0	1.085	0.323	1.08
PD+	21	1	0	-	0	0.569	0	0.569
PID+	15	0.5	10	-	0	1.33	0	1.33
P-SatI-D st	15	1	11	-	0	1.195	0.156	0.846
P-SatI-D sp	15	1	11	2	0	1.07	0.148	0.977

B. M2 Controller Results

TABLE V
CONTROLLER RESULTS M2

Controller	K_P	K_D	K_I	p	Steady State Error (%)		Settling Time (s)		Overshoot (%)		Peak Time (s)	
					S1	S2	S1	S2	S1	S2	S1	S2
PD	52	3	0	-	0.0534	0.0601	0.474	0.89	0	11.4	0.474	0.52
PID	15	1	7	-	0	0	0.635	1.019	1.02	15.3	0.576	0.587
PD+	47	2	0	-	0	0	0.538	1.011	0	13.11	0.538	0.4905
PID+	10	1	5	-	0	0	0.826	1.13	0	7.5	0.826	0.689
P-SatI-D s_t	15	1	6	-	0	0	0.935	1.271	0	0	0.935	1.271
P-SatI-D s_p	15	1	6	2	0	0	0.897	1.141	0	0	0.897	1.141

C. SV1 Controller Results

TABLE VI
CONTROLLER RESULTS SV1

Controllers	K_P	K_D	K_I	p	Steady State Error (%)	Settling Time (s)	Overshoot (%)	Peak Time (s)
PD	9.5	0.6	0	-	-3.83	2.94	4.68	1.74
PID	4.8	0.3	0.9	-	-2.05	2.73	0	1.98
PD+	1.5	0.6	0	-	-3.50	2.16	0	2.16
PID+	0.56	0.4	0.2	-	0.21	2.87	0.22	2.96
P-SatI-D s_t	0.75	0.4	0.2	-	1.27	2.67	1.27	2.67
P-SatI-D s_p	0.75	0.4	0.3	2	1.24	2.56	0.24	2.56

D. SV2 Controller Results

TABLE VII
CONTROLLER RESULTS M2

Controllers	Kp	Kd	Ki	p	Steady State Error (%)		Settling Time (s)		Overshoot (%)		Peak Time (s)	
					S1	S2	S1	S2	S1	S2	S1	S2
PD	2.4	0.5	0	-	-3.31	-1.08	2.52	3.95	0	0	2.52	3.95
PID	2.15	0.5	0.2	-	-1.74	-4.85	2.09	3.17	0	0	2.09	2.4
PD+	0.57	0.1	0	-	0.47	4.71	2.19	3.09	0.47	7.01	2.19	2.07
PID+	0.7	0.2	0.2	-	0.11	3.99	2.84	2.98	0.11	3.99	2.84	2.98
P-SatI-D st	1.1	0.3	0.5	-	1.27	4.55	2.66	3.65	1.27	4.55	2.66	3.65
P-SatI-D sp	0.55	0.2	0.5	2	1.62	6.73	3.34	3.08	1.62	6.88	3.34	2.91

E. Variation of p parameter in P-SatI-D s_p controller

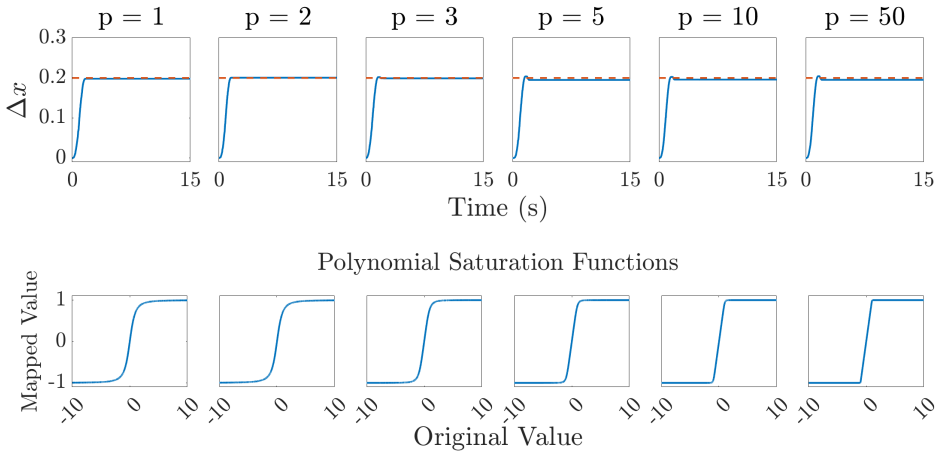


Fig. 13. Effect of variation of p parameter in P-SatI-D s_p controller on step response (first row). The saturation functions corresponding to these p values are also shown for reference (second row).

F. SV1: Tracking single and multiple references

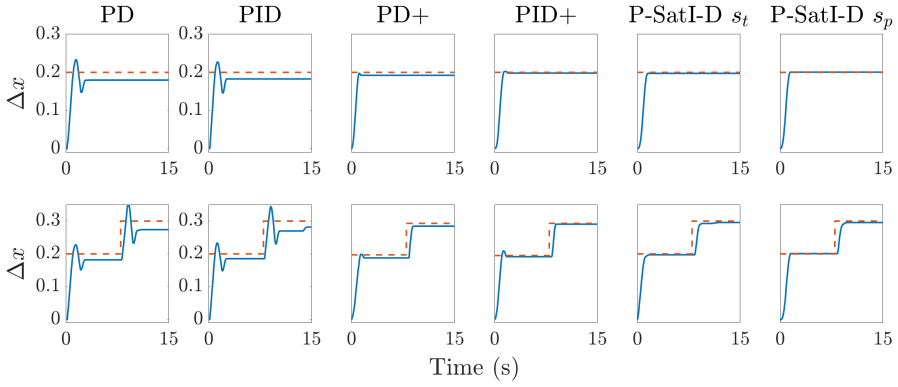


Fig. 14. Experimental results for the single segment robot for tracking single (first row) and multiple (second row) references.

G. SV2: Tracking single and multiple references

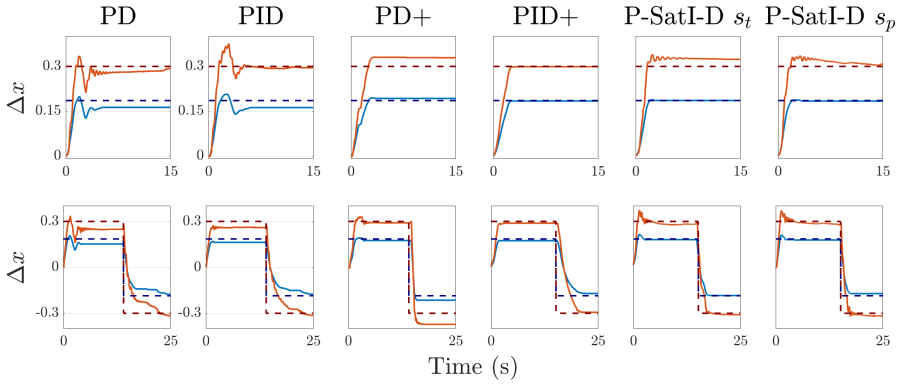


Fig. 15. Experimental results for the two segment robot for tracking single (first row) and multiple (second row) planar references.

H. MG, VP, and VI Experiments for SV1 Setup

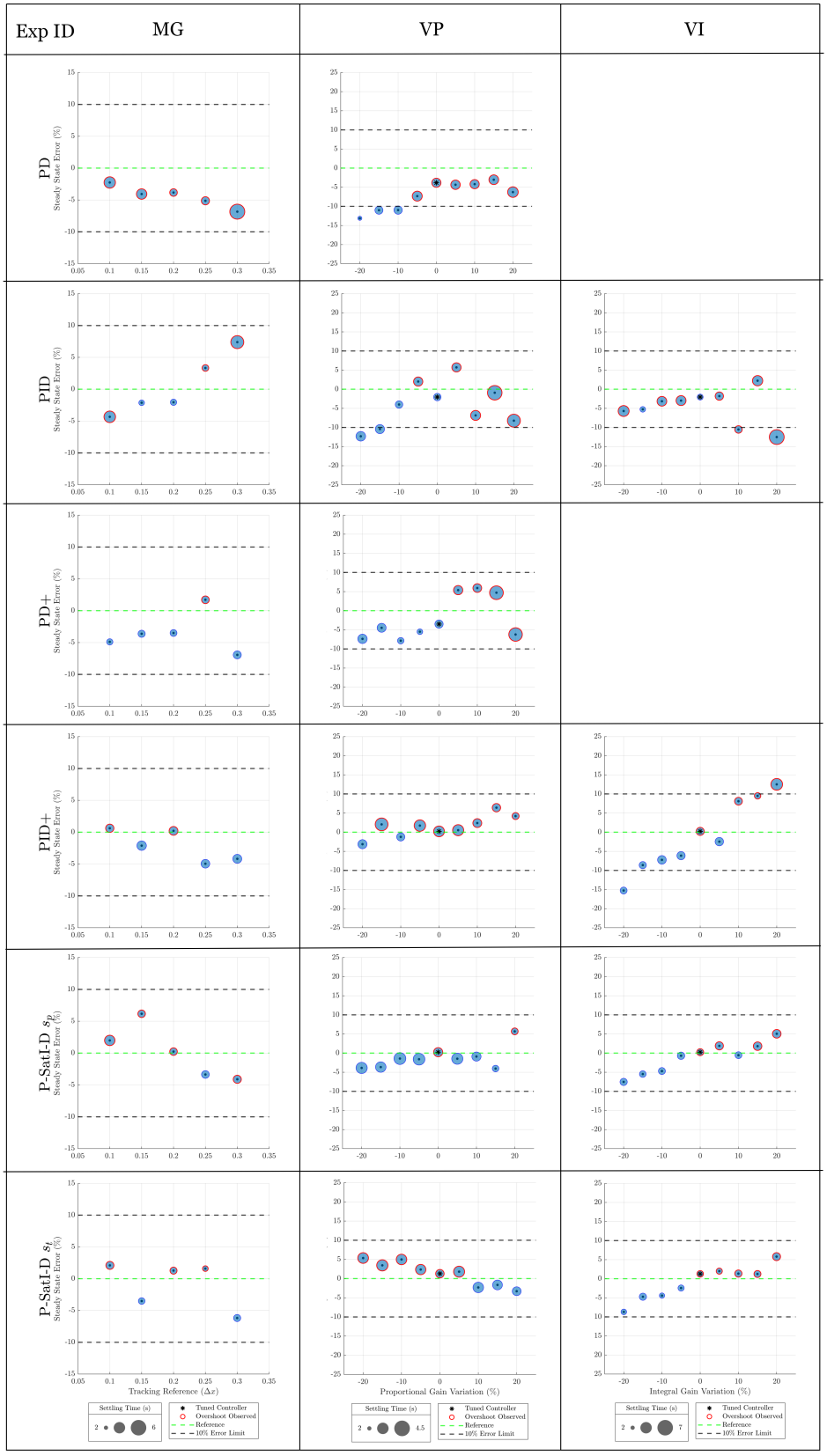


Fig. 16. Effect of variation of K_P, K_I, and the goal q_d for a tuned controller on the SV1 Setup

I. MG, VP, and VI Experiments for SV2 Setup

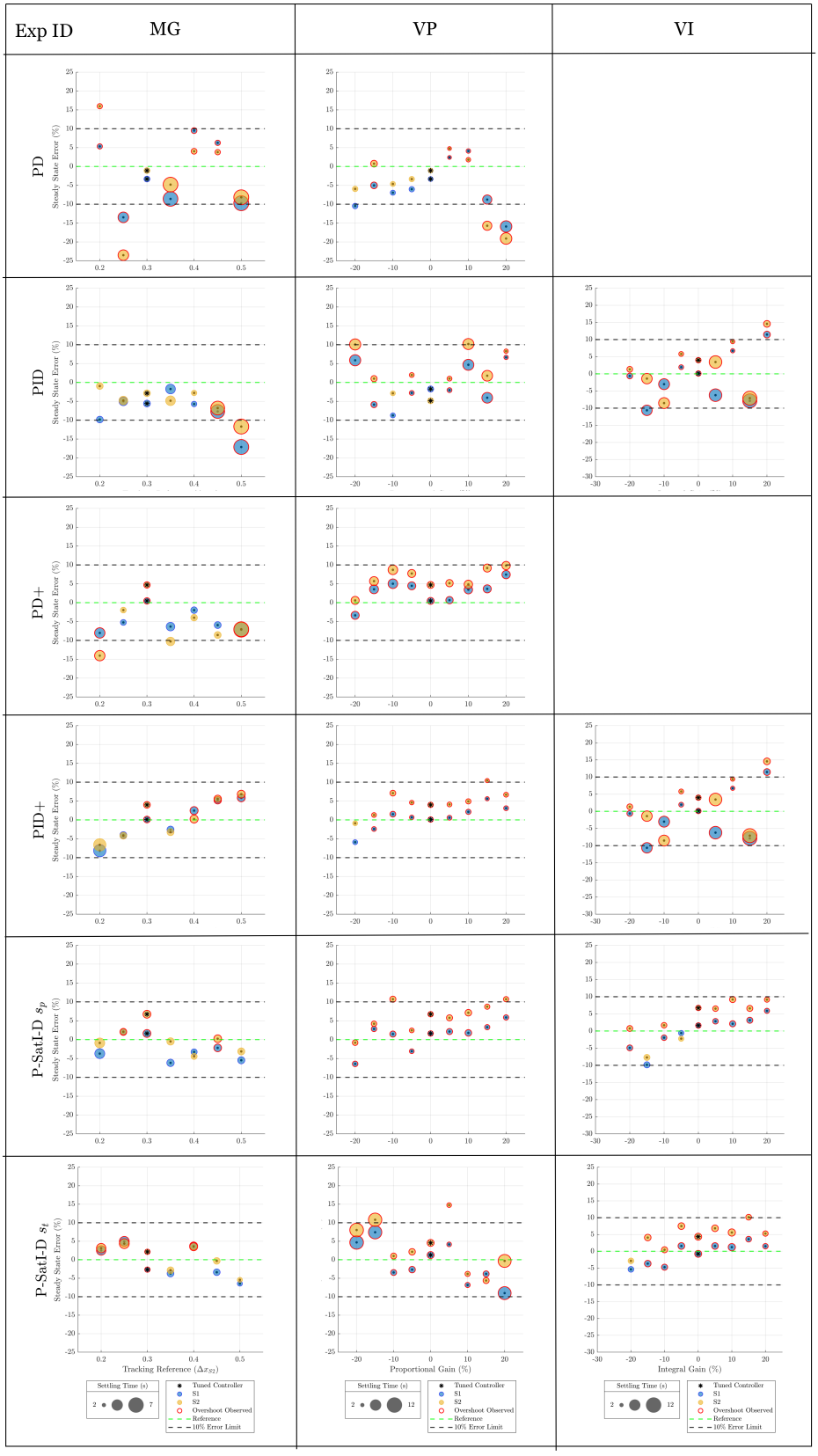


Fig. 17. Effect of variation of K_P, K_I, and the goal q_d for a tuned controller on the SV2 Setup