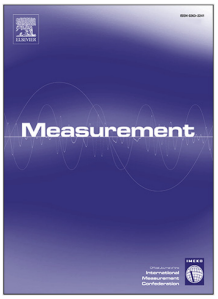
Journal Pre-proof



Dynamic manipulation of pneumatically controlled soft finger for home automation

Ameer Hamza Khan, Shuai Li

PII: S0263-2241(20)31189-1
DOI: https://doi.org/10.1016/j.measurement.2020.108680
Reference: MEASUR 108680

To appear in: *Measurement*

Received date : 30 January 2020
Revised date : 24 September 2020
Accepted date : 27 October 2020

Please cite this article as: A.H. Khan and S. Li, Dynamic manipulation of pneumatically controlled soft finger for home automation, *Measurement* (2020), doi: https://doi.org/10.1016/j.measurement.2020.108680.

- Proposing an approximate model based control strategy for soft robots.
- Using data-driven approaches to approximate the robot's model and formulate an inverse dynamics controller.
- Controlling a soft robotic finger to prove the efficacy of the controller and their application in home automation.
- Extensive experimental result of and comparison with other controllers are presented.

Dynamic Manipulation of Pneumatically Controlled Soft Finger for Home Automation

Ameer Hamza Khan[a], Shuai Li[b,*]

[a]*Department of Computing, Hong Kong Polytechnic University, Hong Kong.*
[b]*Department of Electronics and Electrical Engineering, Swansea University, Swansea SA18EN, UK.*

Abstract

Soft robots have the advantage of inherent flexibility, adaptability, compliance, and safety in human interaction, and therefore attracted significant research attention in recent years. They have found interesting applications in industrial automation where soft robotic hands are fitted as end-effector on traditional rigid robotic arms to handle delicate objects. Their inherent compliance with the shape of the object reduces the complexity of sensing and actuation mechanisms required for the safe operation of traditional robotic hands. They also have the potential application in the home automation, since the operation of robots in indoor environment impose a stringent requirement on safety and compliant design. Despite this, the dynamic manipulation of soft robots remains challenging because their inherent flexibility makes their mathematical model highly nonlinear. Existing works either use model-free control, e.g., PID, which owing to its general formulation, does not account for the peculiarity of soft robots, or they use the Finite-Element-Method based approach, which, apart from being computationally expensive, requires an exact model of the soft robots. In this paper, we take a holistic approach by first developing a low-order approximate mathematical model for computational efficiency and then adding a feedback loop using an inverse dynamics controller to compensate for modeling

*Corresponding author
Email addresses: ameer.h.khan@connect.polyu.hk (Ameer Hamza Khan), shuaili@ieee.org (Shuai Li)

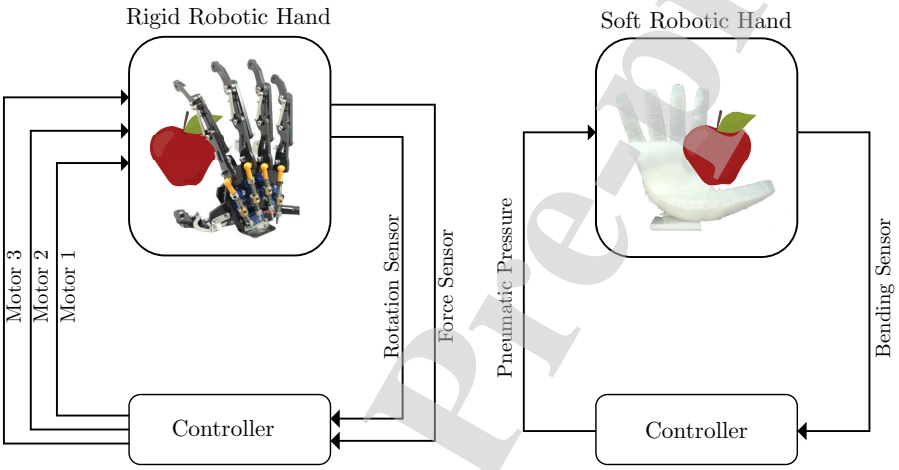
errors. Theoretical analysis is presented to prove the convergence and stability of the proposed controller. Extensive experimental and comparison results also prove the superiority of the proposed controller over other algorithms.

Keywords: Soft robotics, Modeling, Inverse dynamics.

1. Introduction

Soft robots, i.e., the development of robotic systems using soft materials, have been gaining research attention in recent years. The significant features distinguishing soft robots from traditional rigid robots are flexibility, compliance, adaptability, and inherent safety in human interactions. Soft robots can be used to solve several problems posed by rigid robotic systems. A heavy mechanical structure, a limited degree of freedom, potential structural damage, or control circuitry malfunctions are a few of the problems posed by rigid robots that can be solved by using soft robots. Fig. 1 illustrates the differences between rigid and soft robots. The most common types of soft robots use pneumatic pressurization as the actuation source. The body of the robot consists of several inflatable chambers, which expand when the air pressure is increased. The actuation produced by a soft robot depends on the pattern of inflatable chambers. Soft linear actuators [1, 2], and bending soft actuators [3, 4] are popular types of soft robots that are used to produce linear and bending motions, respectively. These can be considered the soft counterparts of linear and servo motors and are being used to replace them in applications requiring safe interactions with delicate objects, especially in rehabilitation applications where soft robots can safely interact with patients performing recovery exercises. Additionally, soft robots are also being used in industrial automation and have the potential to be used for developing autonomous robots for smart homes [5, 6, 7, 8]. The stringent requirement of safety and robustness in a household environment making soft robots an ideal candidate for such applications.

Despite their potential advantages, developing an accurate control algorithms for dynamic manipulation of soft robots is very important if they are to



Pros:
∗ Well-defined finite-DOF mathematical model.

Cons:
∗ Complicated force sensing mechanisms.
∗ Multiple motors inputs for a single finger.
∗ Rigid links can damage the item by excessive force in case of a sensor error.

Pros:
∗ Simple sensing mechanism.
∗ Single-input single-output control topology.
∗ Inherent safety because of flexible fingers.
∗ Simple Pneumatic actuation for bending a finger.

Cons:
∗ Complicated infinite-DOF matematical model.

Figure 1: The contrast between the sensing, actuation, and control mechanisms for rigid and soft robots. The example illustrates the handling of delicate objects required from automation robots. The figure shows the advantages and disadvantages of both robots and makes a case for soft robots in case of home automation.

compete with rigid robots in real-world applications. Soft robots have an infinite degree of freedom [9, 10, 11] due to their flexible bodies, making it impossible to formulate an accurate analytical mathematical model of the soft robot. The lack of an analytical mathematical model poses challenges to precise motion control and the optimization of dynamic responses [12, 13, 14]. To avoid this modeling challenge, most works on soft robotic control have focused on model-free control algorithms. The most commonly used model-free control algorithm is PID [15] and its variants, such as Fuzzy PID. However, the PID algorithm requires the tuning of the controller parameters, which is a labor-intensive task and can possibly result in poor performance due to the subjective judgment of the observer, who is tuning the controller parameters. PID controllers [16] are also non-adaptive to system variations, i.e., their parameters are tuned initially and then remain constant. Since soft robots undergo the inevitable wear and tear and minor structural damage due to their flexible structure, their behavior continually changes, and pre-tuned PIDs cannot adapt to the variations in the system. This non-adaptability of the controller parameters can result in unstable system behavior [17, 18]. Vikas et al. [19] formulated the soft robot motion problem as a graph traversal problem, but their proposed scheme can only be used for the coarse-grained control of soft robot motions. All of the abovementioned techniques are model-free and have a generic formulation for all robotic systems. They do not take into account the specific structure of the soft robot and produce low-accuracy results.

The model-based control of soft robots using the Finite Element Method (FEM) has also been studied. Faure et al. [20] and other recent works [21, 22, 23, 24] proposed a control strategy inspired by the locomotion of aquatic life and used FEM to model the robotic system. This technique can potentially produce highly accurate responses but requires detailed and accurate knowledge of the mechanical properties of the fabrication material. Furthermore, the high computational cost of FEM means that graphics processors (GPUs) are required for their implementation, making it impossible to achieve accurate, fine-grained control on embedded processors in real-time. Reymundo et al. [25] presented

a control technique based on statistical modeling. They used linear regression to model the system's input-output relations, but their controller explicitly assumes that the robot model is linear. This assumption is true only for a small subrange of the system's output but does not hold for the entire range. Several works [26, 27, 28, 29, 30, 31, 32, 9, 33, 34] have proposed a model-based control approach to optimize the dynamic response of soft robotic systems but require a priori knowledge and a mathematical model of the soft robot.

These challenges motivated us to explore the approximate analytical modeling of soft robotic systems. The approximate mathematical model can then be used in combination with the output feedback loop to compensate for inaccuracies in the model, and achieve highly accurate control of soft robots. Since most soft robots are pneumatically actuated, we used Lagrangian mechanics [35] to formulate the approximate mathematical model of a soft actuator. The approximate model is then used to formulate a closed-loop inverse dynamics controller. Inverse dynamics have been intensively studied [36, 37, 38, 39, 40, 41, 42] in control literature and we propose to use it owing to its simple formulation and intuitive explanation. We considered the specific case of a soft bending actuator for modeling, control, and experimentation. In the experimentation section, we presented the results for both an open-loop and a closed-loop inverse dynamics controller. We compared them with a PID controller, based on their response to step inputs. The results of our experiment show that the presented controller can provide better performance than model-free control algorithms such as PID.

This paper demonstrates the use of an approximate analytical model for controlling the motions of a soft robot. The major objectives of this paper are:

- To demonstrate the formulation of an approximate analytical model for a soft robot using principles of Lagrangian mechanics. The method for deriving the Lagrangian of the soft robot and the assumptions used to simplify the derivation process is explained.
- To formulate a closed-loop inverse dynamics controller based on an approximate analytical model. The convergence of the proposed controller

is proved using a Lyapunov function. It is shown that the proposed closed-loop system is asymptotically stable.

The main contributions of this paper are summarized as below:

- Soft robots are inherently difficult to model accurately, preventing model-sensitive approaches to be applied for their control. Due to this reason, the inverse dynamics method has never been reported for the control of soft robots, and whether it is possible to be applied remains an unsolved problem. This work gives a positive answer and presents the first inverse dynamics controller for soft robots.
- A data-driven approach is employed to identify the parameters of soft robots, and the derived data-driven model is applied for inverse dynamics control. This work provides a model on extending powerful nonlinear control tools to address soft robot control problems.
- Extensive experiments verify the effectiveness of the proposed control solution.

The remainder of the paper is organized as follows: Section 2 presents the architecture of the system used for modelling, controlling and experimenting. Section 3 presents mathematical modeling using Lagrangian mechanics, system identification, and controller formulation. Section 4 describes the experimental methodology, platform, and results. Section 5 will conclude the paper.

2. System Architecture

In this section, the system architecture used in modelling and control algorithm formulation will be explained.

2.1. Soft Actuator

A pneumatically actuated soft bending actuator [3] is one of the most popular soft actuators, which has been widely adopted because of its simple and efficient

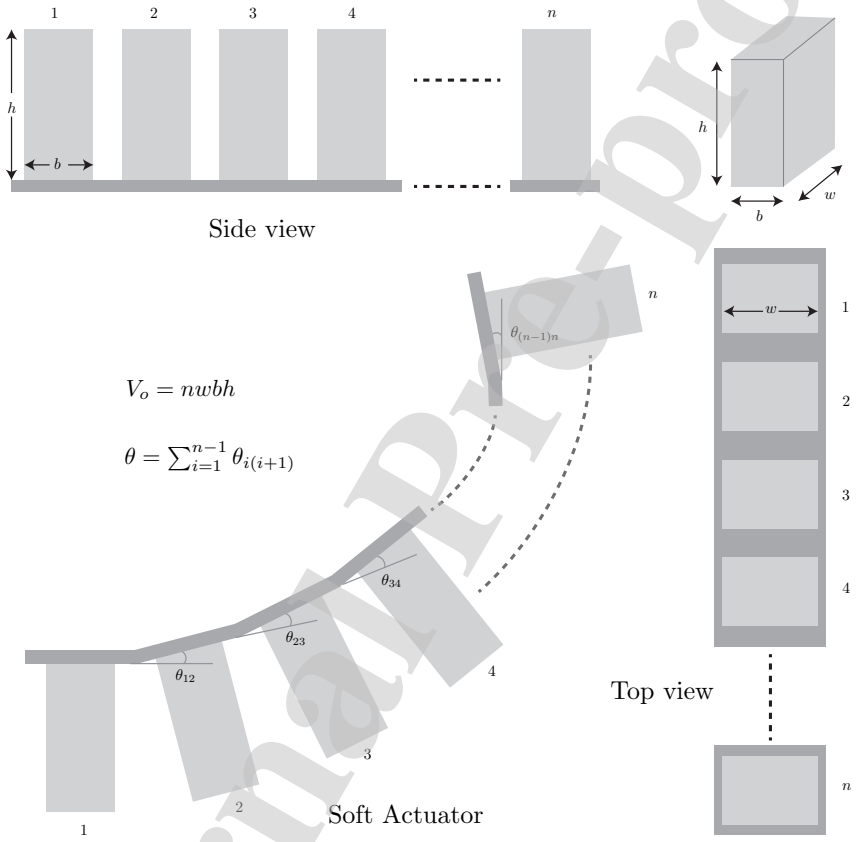


Figure 2: Design of the soft bending actuator used in the experiments. Top left: Side view of the actuator with n chambers. Top right: Dimensions of an individual chamber. Bottom left: Inflated actuator showing the calculation of the bending angle for an n chamber bending actuator. Bottom right: Top view of the soft bending actuator.

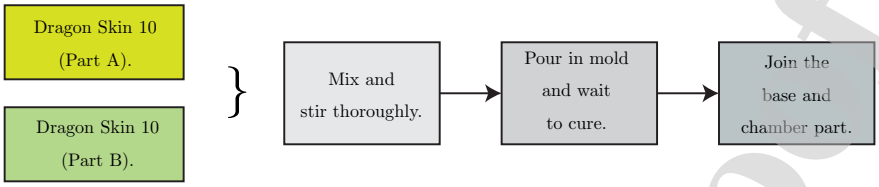
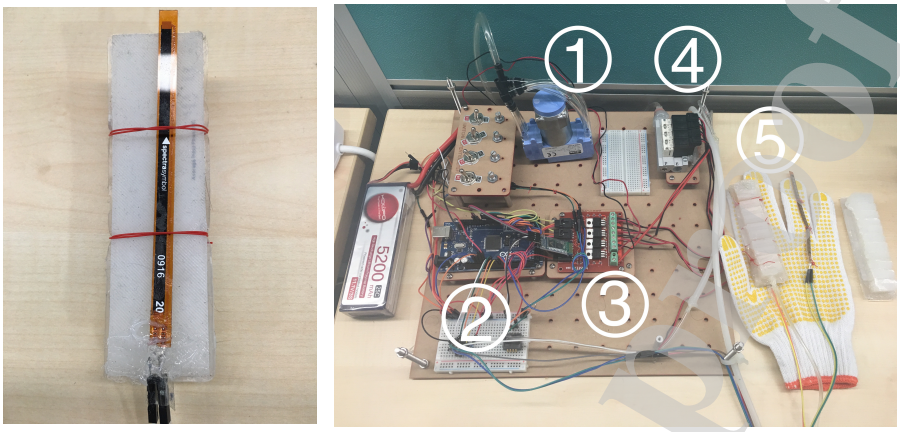


Figure 3: Flowchart of the process for fabricating the soft bending actuator used in the experimental platform.

design and ease of fabrication. The most commonly used design of soft bending actuator consists of a series of inflatable chambers, as shown in Fig. 2. All of the chambers are connected to each other using a thin channel in the base of the actuator. The actuation principle is based on the simultaneous inflation of all of the chambers, using an air pump. It is necessary to make the base of the actuator stiffer (shown in dark color in Fig. 2) than the top chambers to create a bending motion. The difference in stiffness causes the actuator to bend when inflated with pressurized air and can be achieved by embedding a thick sheet of paper in the base of the actuator. The bending angle of the soft actuator is directly proportional to the internal air pressure [43, 44, 1, 45]. The method of controlling the internal air pressure is explained in the next subsection.

Fig. 3 shows the flowchart of the process of fabricating the soft bending actuator. The 3D design of the molds is open source and available in a soft robotics toolkit [46]. The molds were printed using a 3D printer. Dragon Skin 10 from Smooth-On Inc. [47] was used as an elastomer for fabricating the soft bending actuator. The fabrication process is briefly described as follows: Dragon Skin 10 consists of two separate liquid mixtures called Part A and Part B. The process of curing the elastomer starts when Part A and Part B are mixed together (in a 1:1 ratio by weight). We created a mixture of elastomers by thoroughly mixing both parts, and poured the resulting mixture in 3D printed molds. The molds were set aside to cure in open air, and the process took around 6-8 hours before the cured elastomer could be removed from the molds.

Soft robots require different sensing mechanisms from that of a rigid robots.



(a) (b)

Figure 4: Components of the experimental platform. (4a) shows the sensing mechanism of the soft bending actuator used in the experiment. White part: bending actuator; dark part: bending sensor. (4b) shows the experimental platform: containing (1) an air pump, (2) a Mega 2560 microcontroller, (3) MOSFET switches, (4) valves, (5) bending sensor and actuator.

Several novel sensing mechanisms have been developed for soft robots [48, 49]. To sense the bending angle, we used a flex sensor from FlexiForce. The sensor consists of several layers of thin resistive film. The resistance of these films varies with their bending angle. The sensor is essentially a bend-dependent resistor. By measuring the change in resistance at different bending angles, a mapping was generated between the resistance value and the corresponding bending angle of the sensor. The sensor was attached to the back side of a soft bending actuator, so that the bending angles remained exactly same, as shown in Fig. 4a. Since a Flexiforce sensor provides analog voltage, we used ADC pin in our microcontroller to convert it into a digital value for further processing.

For generalized mathematical derivations, a soft bending actuator with n chambers each having width w, height h, and breadth b, as shown in Fig. 2, is considered. When this bending actuator is inflated through air, it will bend to create an angle θ. For the purpose of experimentation, a soft actuator with $n = 11$, i.e. 11 chambers, is fabricated.

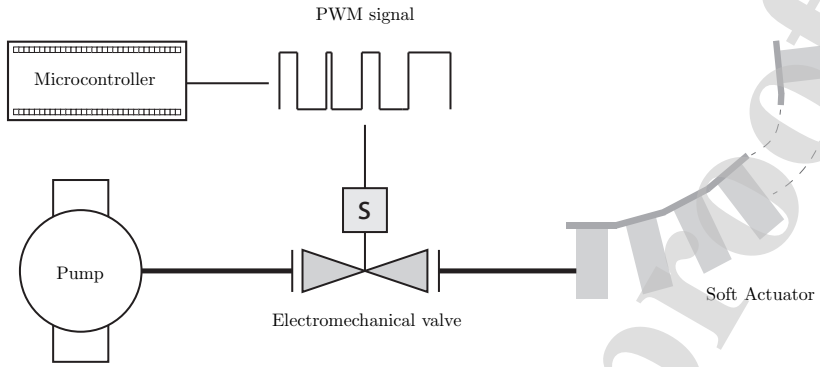


Figure 5: Mechanism to control the air pressure to the soft bending actuator. Electromechanical valves are controlled using the PWM signal of a specified duty cycle.

2.2. Actuation Mechanism

Apart from the fabrication and sensing of a soft bending actuator. The bending of the soft actuator is directly proportional to the internal air pressure [43, 44, 1]. To contol the internal air pressure, we used electromechanical valves. These valves can be controlled by using a microcontroller through MOSFETS switches. A 12V DC pump was attached to the the electromechanical valves. The pressure control mechanism is shown in Fig. 5. By switching the valves repeatedly and controlling their opening and closing times, the air pressure at output of the valves can be controlled as shown by [44]. The opening time of the valves was controlled using a PWM signal of 30 Hz, according to valve specifications. The air pressure at output of valve is a monotonically increasing function of valve's opening time i.e. duty cycle (**D**) of PWM signal.

2.3. Bending Control

The bending angle control system mainly consist of two parts; Microcontroller and and the MOSFET switches. The purpose of these parts is to control the valve opening time according the PWM duty cycle (**D**). The controlling mechanism is shown in Fig. 5. We used Arduino Mega 2560 as our microcontroller, since it has a built-in Analog to the Digital Converter (ADC) for reading sensor values and to a PWM generator to control the switching of valves. The

microcontroller cannot be used directly to control the switching of valves because the ATMega2560 I/O pins operate at 5V, whereas valves operate at 24V. Furthermore, the pins can only provide a maximum current of 40mA, which is insufficient to drive the valves. To solve this problem, we added a layer of MOSFET switches between the microcontroller and the valves. The MOSFETs were connected to a 24V DC battery. Since MOSFET switches can operate at an input voltage of 5V and on negligible current, their input was directly connected to the microcontroller pins and the output was connected with valves. By changing the duty cycle of PWM through the microcontroller, the valves can be opened and closed for specific amounts of time and the air pressure at output of the valves can be regulated. The experimental platform that was constructed is shown in Fig. 4b.

3. Controller formulation

This section presents details of the modeling process using Lagrangian mechanics and a formulation of the inverse dynamic controller along with proof of its convergence.

3.1. System Modeling

As explained in Section 1, soft robotic systems have a soft continually deformable structure, which makes it impossible to formulate an accurate closed-form mathematical However, in this section we will develop an approximate model of the soft bending actuator based on Lagrangian mechanics.

In Lagrangian mechanics, the dynamics of the system can be modeled using the energetics of the system. The Lagrangian $\mathbb{L}$ of a system is mathematically defined as:

$$\mathbb{L} = \mathbb{T} - \mathbb{V}, \tag{1}$$

where $\mathbb{T}$ is the kinetic energy present in the system whereas $\mathbb{V}$ is the potential energy contained in the system. The dynamics of the system can be derived

using Lagrange's equations of the first kind [35]

$$\frac{\partial \mathbb{L}}{\partial \mathbf{r_k}} - \frac{d}{dt}\frac{\partial \mathbb{L}}{\partial \dot{\mathbf{r}}_\mathbf{k}} + \sum \lambda_i \frac{\partial f_i}{\partial \mathbf{r_k}} = 0, \tag{2}$$

where $\mathbf{r_k}$ represent variables present in the system, i.e. θ in our case, and f_is are the functions modeling constraints of the system. Now we will demonstrate the process of deriving relations for potential energy V and kinetic energy T for the soft bending actuator. We then use those relations to calculate the dynamics of the actuator using (1) and (2).

Potential Energy: There are mainly two types of potential energies present in an actuated soft bending robot: elastic potential energy or strain energy and gravitational potential energy. The gravitational potential energy gained after actuation is given by $\mathbb{V}_g = mg\Delta h$, where m is the mass of the actuator, g is the acceleration due to gravity and Δh is the difference in the height of the center of gravity between the actuated and unactuated actuator. Since in the case of the soft bending actuator (Fig. 2), the variation in the height of system is negligible, its gravitational potential energy can safely be ignored. Elastic potential energy $\mathbb{V}_\epsilon$ is the main contributor to the total potential energy of an actuated soft bending robot and is given by following relation:

$$\mathbb{V}_\epsilon = \frac{1}{2} V_\epsilon E \epsilon^2, \tag{3}$$

where V_ϵ is the volumetric strain of actuated bending robot, E is Young's modulus and ϵ is the material strain. Young's modulus E is the ratio between stress and strain. It is a mechanical property of the material and for strain below 50% this ratio remains constant. By using the same approach as [26], we can argue that the strain produced in the actuator is directly related to the bending angle of the actuator, i.e., $\epsilon = k_1\theta$. This is even intuitively correct, because the greater the bending angle of the soft bending actuator, the greater is the elastic potential energy stored in it. By making use of all of these observations, we can get following relation for the elastic potential energy of the actuator.

$$\mathbb{V}_\epsilon = \frac{1}{2} V E k_1^2 \theta^2 \quad \rightarrow \quad \mathbb{V}_\epsilon = k_2 \theta^2. \tag{4}$$

As it has already been argued that the gravitational potential energy is negligible as compared to the elastic potential energy, the total potential energy $\mathbb{V}$ of the system is:

$$\mathbb{V} = \mathbb{V}_\epsilon + \mathbb{V}_g = k_2\theta^2. \tag{5}$$

By following the approach of [26], we define a generalized force τ that is being produced inside the soft bending actuator by differentiating the total potential energy $\mathbb{V}$ with respect to the bending angle θ, i.e., $\tau = \partial\mathbb{V}/\partial\theta$. Therefore

$$\tau = 2k_2\theta = 2\frac{k_2}{k_1}\epsilon, \tag{6}$$

where the last equality follows from the previously explained relation, $\epsilon = k_1\theta$. This generalized force will be used in modeling the dynamics of the system using a Lagrangian equation. The strain ϵ produced in the soft robot is directly proportional to the internal air pressure [43, 1]. The air pressure is supplied to the soft robot through the electromechanical valves. Since the dynamics of the pump and valve is much faster than motion of soft robot itself, therefore the air pressure inside the soft robot is directly proportional to the opening time of the valves. The opening time of valves is proportional to the duty cycle ($\mathbf{D}$) of the controlling PWM signal [44] i.e. $\epsilon \propto P_{internal} \propto T_{valve} \propto \mathbf{D}$, where $P_{internal}$ and T_{valve} denotes the internal pressure of soft robot and opening time of the valve respectively. The proportional relationship between strain and duty cycle ($\mathbf{D}$) can be assumed linear for simplicity. The feedback control will try to minimize the effect of unmodeled nonlinearities. Therefore, the relationship becomes $\epsilon = k_3\mathbf{D}$. Substituting this in (6) results in following relation for generalized force:

$$\tau = 2\frac{k_2k_3}{k_1}\mathbf{D} \quad \rightarrow \quad \tau = k_4\mathbf{D}. \tag{7}$$

This equation reveals the relationship between generalized force and the duty cycle of the valves and will be helpful in modeling dynamics of soft bending actuator.

Kinetic Energy: There are two types of Kinetic energy present in a system:

linear kinetic energy and rotational kinetic energy. Since a soft bending actuator can only perform bending motions, only rotational kinetic energy energy $\mathbb{T}_r$ exists and we can safely ignore linear kinetic energy $\mathbb{T}_l$. The rotational kinetic energy is related to the derivative of the bending angle with respect to time. According to the theory of rotational dynamics, the average rotational kinetic energy is related to the bending angle as $\mathbb{T}_r = \frac{1}{2}mr^2\dot{\theta}^2$, where m is the mass of actuator and r is the distance from the end of the actuator to its center of mass. Due to its symmetrical structure, the center of the mass of the soft bending actuator is located at its middle. Thus, the total kinetic energy $\mathbb{T}$ of the system is:

$$\mathbb{T} = \mathbb{T}_l + \mathbb{T}_r = \frac{1}{2}mr^2\dot{\theta}^2 \quad \rightarrow \quad \mathbb{T} = k_5\dot{\theta}^2. \tag{8}$$

Since the relationship for both kinetic energy $\mathbb{T}$ and potential energy $\mathbb{V}$ have been derived, can write the Lagrangian of the system using (1), as follows:

$$\mathbb{L} = k_5\dot{\theta}^2 - k_2\theta^2. \tag{9}$$

Now (2) is used to model the dynamics of the soft bending actuator using Lagrangian $\mathbb{L}$, as follows:

$$\frac{\partial \mathbb{L}}{\partial \theta} - \frac{d}{dt}\frac{\partial \mathbb{L}}{\partial \dot{\theta}} + \tau + b'\dot{\theta} = 0. \tag{10}$$

The $b'\dot{\theta}$ term is introduced in the above equation to model non-conservative frictional forces present in system, b' is a damping coefficient. By replacing the value of $\mathbb{L}$ and τ in (10), we get following model:

$$-2k_2\theta - 2k_5\ddot{\theta} + k_4\mathbf{D} + b'\dot{\theta} = 0, \tag{11}$$

which can be further simplified into the standard form of a second order system, as follows:

$$\ddot{\theta} = a_1\theta + a_2\dot{\theta} + b\mathbf{D}. \tag{12}$$

This equation captures the dynamic relationship between duty cycle $\mathbf{D}$ and bending angle θ and shows that the dynamics can be represented as a second order system. The parameters set $\{a_1, a_2, b\}$ needs to be determined experimentally.

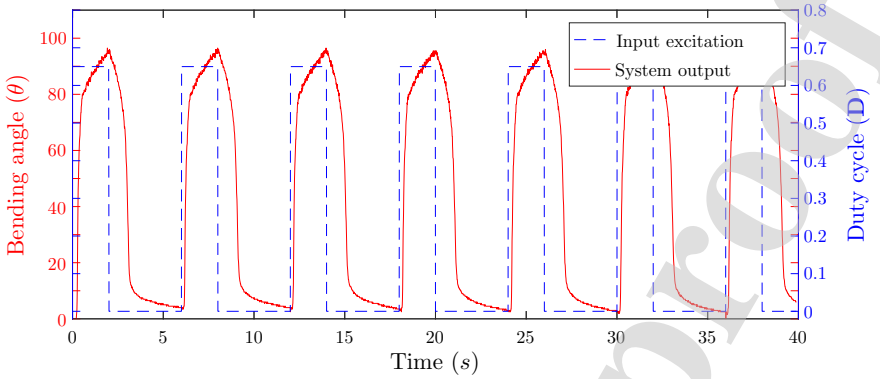


Figure 6: Input excitation and system output plots for identifying model (12) parameters. Blue curve show the input excitation signal, i.e. input duty cycle **D**. Red curve shows the response of the system corresponding to the given excitation.

3.2. Inverse Dynamics Controller Design

Now we will present the design of an inverse dynamics based controllers based on the actuator model given in (12). The main task here is to formulate a strategy to control the duty cycle D of the PWM signal, i.e., writing **D** in terms of a dynamic relation of the bending angle θ. First, we will present a method to experimentally identify the system parameters $\{a_1, a_2, b\}$; then, we will describe the controller formulation based on closed loop inverse dynamics.

To identify the system parameters, we excited the system using a periodic step signal, as shown in Fig. 6, and recorded the output response. We used the MATLAB system identification toolbox [50] to identify the system parameters and calculated following parameter values: $a_1 = -1.945\times10^4$, $a_2 = -1.579\times10^4$ and $b = 3.816 \times 10^6$.

Now we will derive a control law for controlling the bending angle (θ) of the soft bending actuator by inverting the system model (12) and solving the equation for (**D**):

$$\mathbf{D} = \frac{1}{b}(\ddot{\theta} + a_1\theta + a_2\dot{\theta}). \tag{13}$$

From this equation a feed-forward open loop controller can be formulated similar to [44]. If the reference angle of the soft bending actuator is represented by θ_r

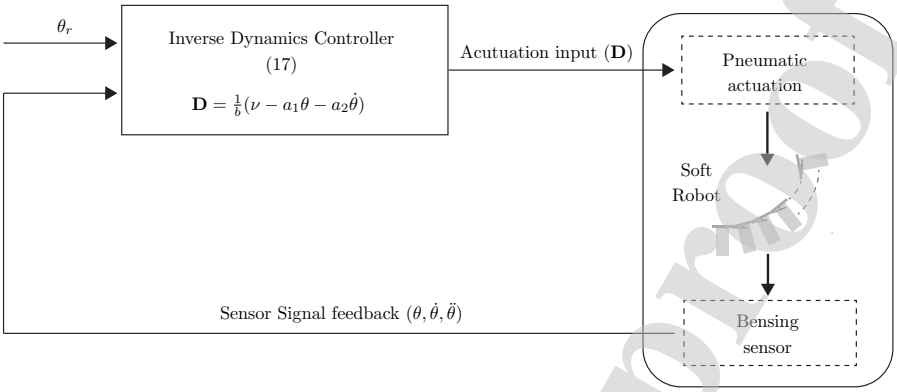


Figure 7: A schematic diagram of the soft robotic platform used in this paper. The diagram illustrates how the actuator and the inverse dynamics controller connectes in a feedback configuration.

then we have following relation from the definition of $\ddot{\theta}$

$$\ddot{\theta}_r = \frac{\theta_r - 2\theta[n] + \theta[n-1]}{\Delta t^2}. \tag{14}$$

where Δt is the sampling time period and $\theta[n]$ is the current value of the bending angle, while $\theta[n-1]$ represents the past value. Similarly, we have following relation:

$$\dot{\theta}_r = \frac{\theta_r - \theta[n]}{\Delta t}. \tag{15}$$

Replacing these values in (13) results in the following relation for the feed-forward open loop controller

$$\mathbf{D} = \frac{1}{b}(\ddot{\theta}_r + a_1\theta_r + a_2\dot{\theta}_r). \tag{16}$$

An important point to note is that the steady state response is given by $\mathbf{D} = a_1\theta_r/b$. Thus, a_1 and b are main factor controlling steady state response and resultantly steady state error. Since open loop feed-forward systems usually exhibit a constant steady state error, during the experiments we tried adjusting these parameters to improve the steady state performance of the open loop controller. Details of this process are provided in Section 4.

Now we will design a closed loop controller to compensate for variations and uncertainties in parameters $\{a_1, a_2, b\}$. Based on (12), a closed loop inverse dynamic based controller can be formulated that is similar to one proposed in [36], as:

$$\mathbf{D} = \frac{1}{b}(\nu - a_1\theta - a_2\dot{\theta}), \tag{17}$$

where $\nu = \kappa_p\tilde{\theta} + \kappa_d\dot{\tilde{\theta}} + \ddot{\theta}_r$ and $\tilde{\theta} = \theta_r - \theta$ is the output error. The schematic diagram of the controller is shown in the Figure 7. To analyze the error dynamics of the proposed controller, substitute $\mathbf{D}$ from (17) to (12). The error dynamics of the system are governed by the following differential equation:

$$\ddot{\tilde{\theta}} + \kappa_p\tilde{\theta} + \kappa_d\dot{\tilde{\theta}} = 0. \tag{18}$$

For this error dynamics equation we will now show that error $\tilde{X} \to 0$ as $t \to \infty$, i.e., the closed loop controller is asymptotically stable.

In order to analyze the stability of the proposed controller, we consider the following Lyapunov candidate function:

$$V(\tilde{\theta}, \dot{\tilde{\theta}}) = \frac{1}{2}\kappa_p\tilde{\theta}^2 + \frac{1}{2}\dot{\tilde{\theta}}^2. \tag{19}$$

We take the first-order derivative of this function and replace $\ddot{\tilde{\theta}}$ from (18):

$$\begin{aligned} \dot{V}(\tilde{\theta}, \dot{\tilde{\theta}}) &= \kappa_p\tilde{\theta}\dot{\tilde{\theta}} + \dot{\tilde{\theta}}\ddot{\tilde{\theta}}, \\ &= \kappa_p\tilde{\theta}\dot{\tilde{\theta}} + \dot{\tilde{\theta}}(-\kappa_p\tilde{\theta} - \kappa_d\dot{\tilde{\theta}}), \\ &= -\kappa_d\dot{\tilde{\theta}}^2 \leq 0. \end{aligned} \tag{20}$$

The above inequality holds as long as $\kappa_p, \kappa_d \geq 0$. Thus, the closed loop control system is asymptotically stable and the output error will converge to zero.

4. Experiments & Discussion

In this section we will present the experimental platform used to verify the efficacy of the presented controller. The experiment methodology to get the response of the soft bending actuator to the proposed controller will be presented.

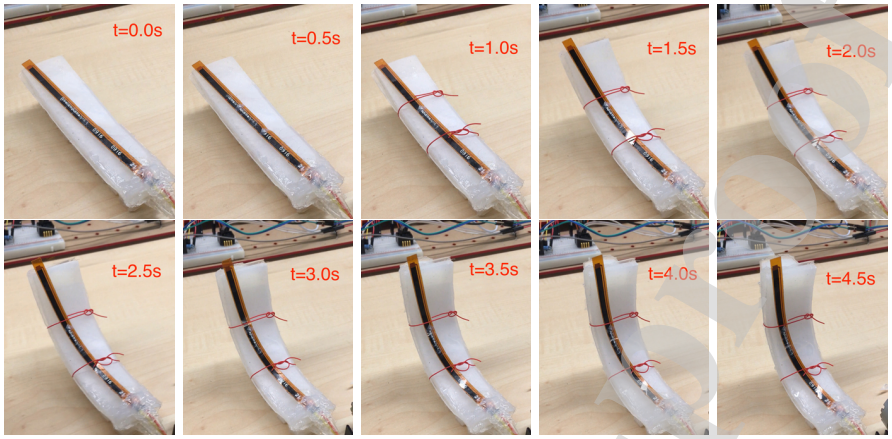


Figure 8: An example showing bending of the soft actuator with time as air is pumped into it. At $t = 0s$: the soft actuator is rest. From $t = 0.5s$ to $1s$: the soft actuator is inflated and starts to move from rest. From $t = 1.5s$ to $2.5s$: the soft actuator has been inflated to a visible angle and keeps bending further. At $t = 3.0s$: the soft actuator arrived at the desired angle. From $t = 3.5s$ to $4.0s$, the soft actuator overshoot slightly around the desired angle. At $4.5s$, the soft actuator returns back to the desired angle.

A comparison with a model-free benchmark controller; PID controller is also provided to demonstrate the effectiveness of the proposed controller. PID controllers are the most commonly used model-free controllers for controlling soft robots. The results of the comparison show that our proposed model-based controller can provide a better performance than PID, while avoiding the hassle of parameter tuning, which is necessary in case of PID. This is achieved since the controller based on the mathematical model of the actuator, thus accounting for the specific structure of the actuator.

4.1. Experimental Platform

In this section we will explain the experimental methodology and setup to verify the efficacy of the proposed controller and its comparison with the benchmark PID controller. As already explained in 2, our system consisted of a soft bending actuator along with pneumatics and electronics to control the bending angle of soft actuator. The major components of the experimental platform

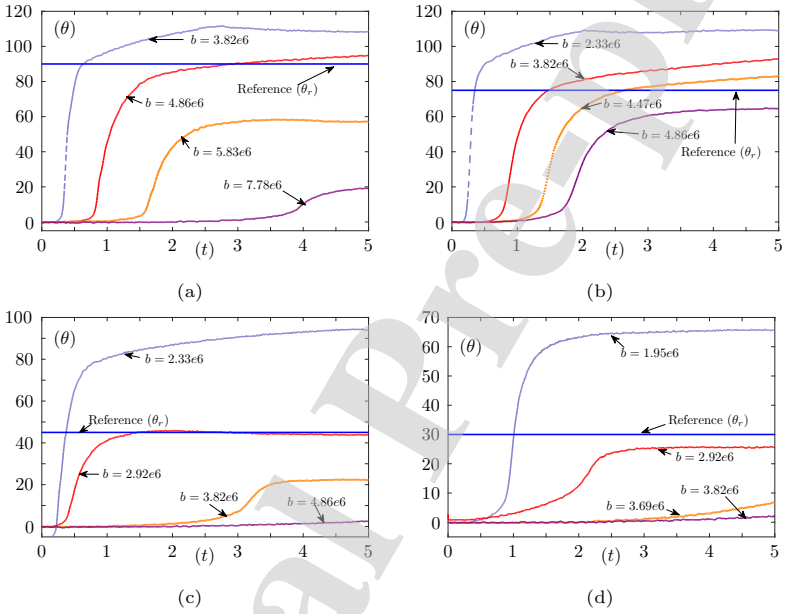


Figure 9: Dynamic response of a system using the feed-forward open loop controller of (16). Due to absence of a feedback loop all responses show large steady state error. Since steady state response depend on ratio $-a_1/b$, therefore experiments were conducted with variation of b. (9a) shows response of system when reference angle $\theta_r = 90^o$. (9b), (9c), and (9d) correspond to $\theta_r = 75^o$, $\theta_r = 45^o$, and $\theta_r = 30^o$, respectively.

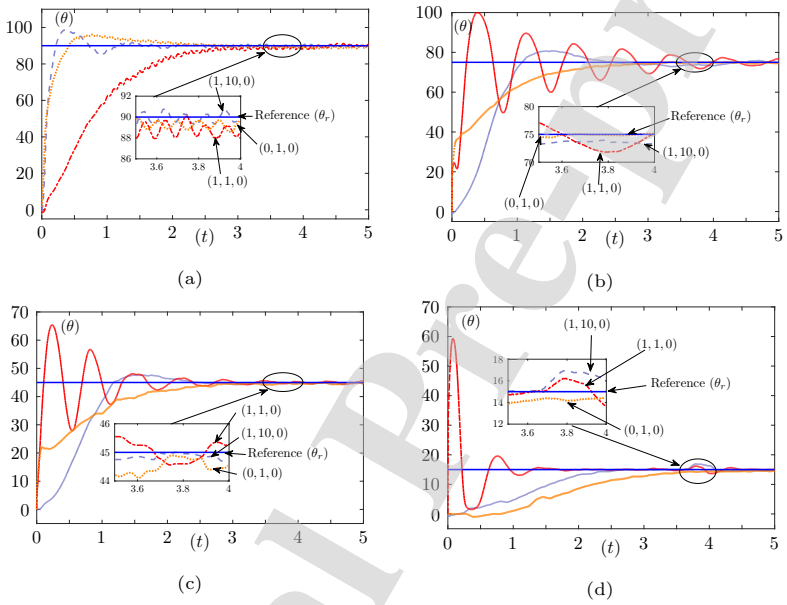


Figure 10: Dynamic response of a system using the PID controller of (21). Ordered pairs in these plots represents value of PID controller parameter (K_p , K_i , K_d). The steady state error converges to zero due to the presence of an output feedback loop. Experiments were conducted using parameters of different values to observe the effect on the system's response. (10a) shows the response of the system when the reference angle $\theta_r = 90^o$. (10b), (10c) and (10d), correspond to $\theta_r = 75^o$, $\theta_r = 45^o$, and $\theta_r = 15^o$, respectively. PID controller converges in all cases.

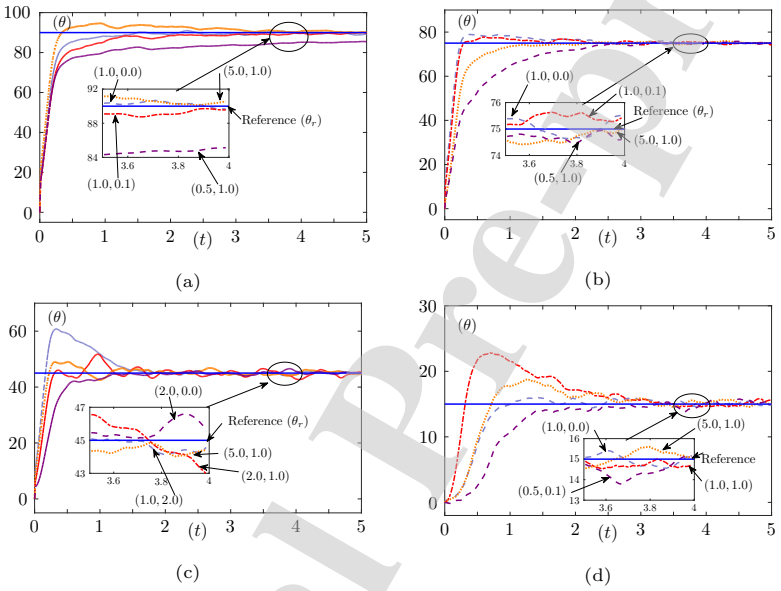


Figure 11: Dynamic response of a system using the closed loop inverse dynamics controller of (17). Ordered pairs in these plots represents value of inverse dynamics controller parameter $(\kappa_p\ ,\ \kappa_d)$. The steady state error converges to zero as it has already been proven that the closed loop system is asymptotically stable. Experiments were conducted using different values of controller parameters κ_p and κ_d. (11a) shows the response of the system when the reference angle $\theta_r = 90^o$. (11b), (11c), and (11d) correspond to $\theta_r = 75^o$, $\theta_r = 45^o$, and $\theta_r = 15^o$, respectively.

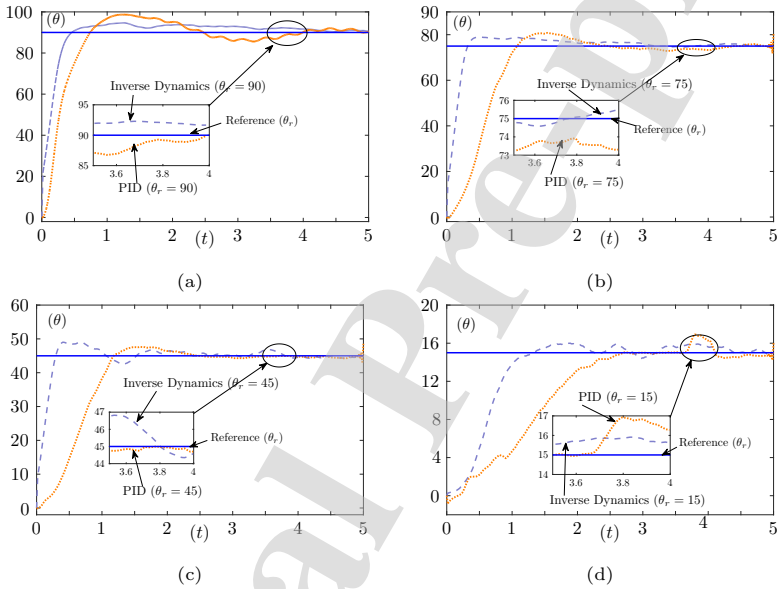


Figure 12: Comparison of the best observed performances of the closed loop inverse dynamic controller against the PID controller. (12a) shows a comparison of performance when the reference angle $\theta_r = 90^o$. Similarly, (12b), (12c), and (12d) shows a comparison for $\theta_r = 75^o$, $\theta_r = 45^o$, and $\theta_r = 15^o$, respectively.

consisted of a DC air pump, pneumatic valves, MOSFET switches and an ATMega2560 microcontroller as shown in Fig. 4b. The soft robot and sensing elements used are shown in Fig. 4a. Fig. 8 shows the snapshots of a typical experiment to determine the response of the actuator to a step input signal.

To determine the accuracy of the proposed controller, and their comparison, we excited the system with a step input signal and recorded the transient response of the system. Mean square error between step signal and the dynamic response is measured as the performance metric for each experiment. The first set of experiments consisted of evaluating the performance of open loop inverse dynamics controller of (16). The system was excited with step signal of different amplitudes i.e. $\theta_r \in \{30^o, 45^o, 75^o, 90^o\}$. These values were chosen because the typical operating range of the bending actuator ranges from 0^o to 90^o. Choosing these values covers the operational range and thus provides a systematic way to evaluate the performance of the controllers. The second set of experiments consisted of measuring the performance of benchmark PID controller. These results are used as benchmark to evaluate the performance of the proposed closed loop inverse dynamics controller. Similar to previous case, the system was operated with step signal of different amplitudes and actuator's tracking performance is recorded. The final set of experiments consisted of evaluating the performance of closed loop inverse dynamics controller of (17). Similar to other cases, the system was excited with different step signals. The results of experimental trials are presented and discussed in next section.

4.2. Results & Discussion

We conducted several sets of experiments to compare the performance of the proposed controller with other control techniques. Here, three types of results will be presented: the first are those based on open loop inverse dynamics, as given in (16)); the second are those based on the use of a PID controller; and the third are those based on the use of closed loop inverse dynamics, as given in (17). The results of the experiment show that the third method produces the best results mainly because it combines the system model of method one and

the feedback loop of method two.

4.2.1. Benchmark Controller

We also performed experiments with a PID controller. A PID controller provides a baseline for evaluating the performance of the proposed controller. The following formulation for discrete-time PID was used:

$$D[n] = K_p e[n] + K_i \sum_{i=1}^{n} e[i]\Delta t_i + K_d \frac{\Delta e_n}{\Delta t_n}, \tag{21}$$

where n is the count of current sample number, $D[n]$ is the duty cycle of the valve switches, $e[n] = \theta_r - \theta$ and K_p, K_i, and K_d are the control parameters for the PID controller. The PID controller was tried for different values of parameters and reference angles θ_r. Fig. 10 shows the results of the experiments. Each figure represents the performance of the soft bending actuator for a constant θ_r and different sets of PID parameter values. It is most important point to note here that the steady state output error eventually converges to zero and that the system is able to accurately track the input reference signal. It can also be seen that by changing the parameter values, it is possible to adjust and optimize the dynamic response of the system.

4.2.2. Open Loop Inverse Dynamics

The first set of experiments was performed to evaluate the performance of the feed-forward open loop controller. The system was given a constant reference angle θ_r and valves were controlled using open loop controller of (16). The experiments were repeated for different values of θ_r and parameters. Since the steady state response depends on having a ratio of $-a_1/b$, therefore only b was varied in the experiments, while $\{a_1, a_2\}$ remained constant. The results are shown in Fig. 9. It can be seen that the system suffers from high values of output error, and that in most cases the steady state response was unable to accurately track the reference signal θ_r. This is the result of the absence of a feedback loop, with the controller failing to adjust itself to compensate for variations in the parameters of the system.

4.2.3. Closed Loop Inverse Dynamics

We now analyze the performance of the closed loop inverse dynamic controller as proposed in (17). Similar to other methods, we run the system with different sets of parameter values and with reference angle θ_r. The results of the experiment are shown in Fig. 11. Each figure represents the system's response for a constant reference angle θ_r and different values of parameters. Since κ_p and κ_d are the main design parameters in a closed loop controller, we present the effect of variations in them. We have already proven the asymptotical convergence of a closed loop control system; therefore, similar to PID the steady state output error of the system eventually converges to zero. It can be seen that, similar to PID, the dynamic response of this controller can be tuned and optimized, by adjusting the parameter values.

We will now compare the performance of the presented controllers. It is evident from Fig. 9 that the feed-forward open loop controller produces a high steady state output error showing a poor performance, whereas Figs. 10 and 11 show that the PID controller and closed loop inverse dynamics controllers eventually converge to the given reference angle θ_r. Therefore, for the purpose of this comparison, only closed loop inverse dynamic and PID controllers are considered. Both controllers are given the same value of θ_r, and the best responses are shown in Fig. 12. Each figure represents a constant value of θ_r for different controllers. It can be seen that the performance of the closed loop inverse dynamics controller is better than that of the PID controller in all cases. Its exhibits low rise time and small steady state error as compared to PID.

5. Conclusion

In this paper we presented an approximate model of a soft bending actuator and used it to design an inverse dynamics closed loop controller. We presented both an open loop and closed loop design for the controller and provided an experimental comparison of their performance with a PID controller as the benchmark. The results of our experiment show that promising progress has

been made in developing a model based dynamic control of a soft robotic system, by developing an approximate model of the soft robotic system and then compensating for variations in the model using output feedback. Before ending this section as well as this paper, it is worthy mentioning that this is the first time that an inverse dynamics control approach has been achieved for soft robots.

A potential future research direction includes developing an adaptive controller to automatically account for unknown parameters and unmodeled dynamics of the soft robots. The adaptive controller starts with an ideal mathematical model and adaptively updates its parameters based on the feedback from sensing and control signals. Such a formulation will remove the need to accurately estimate the robot's model, increasing the reliability and efficiency of the development process.

References

[1] B. Tondu, Modelling of the mckibben artificial muscle: A review, Journal of Intelligent Material Systems and Structures 23 (2012) 225–253. doi:10.1177/1045389X11435435.
URL http://jim.sagepub.com/content/23/3/225.abstract

[2] M. Doumit, A. Fahim, M. Munro, Analytical modeling and experimental validation of the braided pneumatic muscle, IEEE transactions on robotics 25 (6) (2009) 1282–1291.

[3] K. Galloway, P. Polygerinos, C. Walsh, R. Wood, Mechanically programmable bend radius for fiber-reinforced soft actuators, in: Advanced Robotics (ICAR), 2013 16th International Conference on, IEEE, 2013, pp. 1–6.

[4] P. Polygerinos, Z. Wang, J. T. Overvelde, K. C. Galloway, R. J. Wood, K. Bertoldi, C. J. Walsh, Modeling of soft fiber-reinforced bending actuators, IEEE Transactions on Robotics 31 (3) (2015) 778–789.

[5] S. Kadry, Diagnostics and Prognostics of Engineering Systems: Methods and Techniques: Methods and Techniques, IGI Global, 2012.

[6] S. Kadry, G. Alferov, A. Kondratyuk, V. Kurochkin, S. Zhao, Modeling the motion of a space manipulation robot using position control, in: AIP Conference Proceedings, Vol. 2116, AIP Publishing LLC, 2019, p. 080005.

[7] Y. Zhang, S. Wang, G. Ji, P. Phillips, Fruit classification using computer vision and feedforward neural network, Journal of Food Engineering 143 (2014) 167–177.

[8] Y. Zhang, L.-N. Wu, S.-H. Wang, Survey on development of expert system, Jisuanji Gongcheng yu Yingyong(Computer Engineering and Applications) 46 (19) (2010).

[9] C. Della Santina, R. K. Katzschmann, A. Bicchi, D. Rus, Dynamic control of soft robots interacting with the environment (2018).

[10] A. A. Jaoude, K. El-Tawil, S. Kadry, H. Noura, M. Ouladsine, Analytic prognostic model for a dynamic system, International Review of Automatic Control 3 (6) (2010) 568–577.

[11] S. Kadry, G. Alferov, G. Ivanov, A. Sharlay, Stabilization of the program motion of control object with elastically connected elements, in: AIP Conference Proceedings, Vol. 2040, AIP Publishing LLC, 2018, p. 150014.

[12] Y. Zhang, P. Agarwal, V. Bhatnagar, S. Balochian, J. Yan, Swarm intelligence and its applications, The Scientific World Journal 2013 (2013).

[13] Y. Zhang, Y. Jun, G. Wei, L. Wu, Find multi-objective paths in stochastic networks via chaotic immune pso, Expert Systems with Applications 37 (3) (2010) 1911–1919.

[14] L. Cheng, W. Liu, C. Yang, T. Huang, Z.-G. Hou, M. Tan, A neural-network-based controller for piezoelectric-actuated stick–slip devices, IEEE Transactions on Industrial Electronics 65 (3) (2017) 2598–2607.

[15] G. Gerboni, A. Diodato, G. Ciuti, M. Cianchetti, A. Menciassi, Feedback control of soft robot actuators via commercial flex bend sensors, IEEE/ASME Transactions on Mechatronics (2017).

[16] K. Ang, G. Chong, Y. Li, Pid control system analysis, design, and technology, IEEE transactions on Control Systems Technology 13 (4) (2005) 559–576.

[17] S. Ling, H. Wang, P. X. Liu, Adaptive fuzzy dynamic surface control of flexible-joint robot systems with input saturation, IEEE/CAA Journal of Automatica Sinica 6 (1) (2019) 97–107.

[18] C. Yang, G. Peng, Y. Li, R. Cui, L. Cheng, Z. Li, Neural networks enhanced adaptive admittance control of optimized robot–environment interaction, IEEE transactions on cybernetics 49 (7) (2018) 2568–2579.

[19] V. Vikas, P. Grover, B. Trimmer, Model-free control framework for multi-limb soft robots, in: Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on, IEEE, 2015, pp. 1111–1116.

[20] M. Calisti, M. Giorelli, G. Levy, An octopus-bioinspired solution to movement and manipulation for soft robots, Bioinspiration & biomimetics 6 (3) (2011) 036002.

[21] T. Li, K. Nakajima, M. Kuba, T. Gutnick, B. Hochner, R. Pfeifer, From the octopus to soft robots control: an octopus inspired behavior control architecture for soft robots, Vie et milieu 61 (4) (2011) 211–217.

[22] C. Laschi, M. Cianchetti, B. Mazzolai, L. Margheri, M. Follador, P. Dario, Soft robot arm inspired by the octopus, Advanced Robotics 26 (7) (2012) 709–727.

[23] A. Marchese, C. Onal, D. Rus, Autonomous soft robotic fish capable of escape maneuvers using fluidic elastomer actuators, Soft Robotics 1 (1) (2014) 75–87.

[24] M. Shang, X. Luo, Z. Liu, J. Chen, Y. Yuan, M. Zhou, Randomized latent factor model for high-dimensional and sparse matrices from industrial applications, IEEE/CAA Journal of Automatica Sinica 6 (1) (2018) 131–141.

[25] A. Reymundo, E. Muñoz, M. Navarro, E. Vela, H. Krebs, Hand rehabilitation using soft-robotics, in: Biomedical Robotics and Biomechatronics (BioRob), 2016 6th IEEE International Conference on, IEEE, 2016, pp. 698–703.

[26] A. Marchese, R. Tedrake, D. Rus, Dynamics and trajectory optimization for a soft spatial fluidic elastomer manipulator, The International Journal of Robotics Research 35 (8) (2016) 1000–1019.

[27] F. Renda, M. Giorelli, M. Calisti, M. Cianchetti, C. Laschi, Dynamic model of a multibending soft robot arm driven by cables, IEEE Transactions on Robotics 30 (5) (2014) 1109–1122.

[28] I. Gravagne, C. Rahn, I. Walker, Large deflection dynamics and control for planar continuum robots, IEEE/ASME transactions on mechatronics 8 (2) (2003) 299–307.

[29] Y. Yekutieli, R. Sagiv-Zohar, R. Aharonov, Y. Engel, B. Hochner, T. Flash, Dynamic model of the octopus arm. i. biomechanics of the octopus reaching movement, Journal of neurophysiology 94 (2) (2005) 1443–1458.

[30] M. Luo, E. H. Skorina, W. Tao, F. Chen, S. Ozel, Y. Sun, C. D. Onal, Toward modular soft robotics: Proprioceptive curvature sensing and sliding-mode control of soft bidirectional bending modules, Soft robotics 4 (2) (2017) 117–125.

[31] T. G. Thuruthel, E. Falotico, F. Renda, C. Laschi, Learning dynamic models for open loop predictive control of soft robotic manipulators, Bioinspiration & biomimetics 12 (6) (2017) 066003.

[32] C. Della Santina, R. K. Katzschmann, A. Bicchi, D. Rus, Dynamic control of soft robots interacting with the environment (2018).

[33] V. Oguntosin, S. J. Nasuto, Y. Hayashi, Embedded fuzzy logic controller for positive and negative pressure control in pneumatic soft robots, in: Computer Modelling & Simulation (UKSim), 2017 UKSim-AMSS 19th International Conference on, IEEE, 2017, pp. 63–68.

[34] H. Wang, B. Yang, Y. Liu, W. Chen, X. Liang, R. Pfeifer, Visual servoing of soft robot manipulator in constrained environments with an adaptive controller, IEEE/ASME Transactions on Mechatronics 22 (1) (2017) 41–50.

[35] T. Shirafuji, Lagrangian mechanics of massless particles with spin, Progress of Theoretical Physics 70 (1) (1983) 18–35.

[36] F. Saunders, B. A. Trimmer, J. Rife, Modeling locomotion of a soft-bodied arthropod using inverse dynamics, Bioinspiration & biomimetics 6 (1) (2010) 016001.

[37] A. Herzog, N. Rotella, S. Mason, F. Grimminger, S. Schaal, L. Righetti, Momentum control with hierarchical inverse dynamics on a torque-controlled humanoid, Autonomous Robots 40 (3) (2016) 473–491.

[38] L. V. Santana, A. S. Brandao, M. Sarcinelli-Filho, R. Carelli, A trajectory tracking and 3d positioning controller for the ar. drone quadrotor, in: Unmanned Aircraft Systems (ICUAS), 2014 International Conference on, IEEE, 2014, pp. 756–767.

[39] C. Pizzolato, M. Reggiani, L. Modenese, D. Lloyd, Real-time inverse kinematics and inverse dynamics for lower limb applications using opensim, Computer methods in biomechanics and biomedical engineering 20 (4) (2017) 436–445.

[40] D. Kappler, F. Meier, N. Ratliff, S. Schaal, A new data source for inverse dynamics learning, in: Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on, IEEE, 2017, pp. 4723–4730.

[41] H. Kang, C. Liu, Y.-B. Jia, Inverse dynamics and energy optimal trajectories for a wheeled mobile robot, International Journal of Mechanical Sciences 134 (2017) 576–588.

[42] H. Wang, P. X. Liu, X. Zhao, X. Liu, Adaptive fuzzy finite-time control of nonlinear systems with actuator faults, IEEE transactions on cybernetics (2019).

[43] B. Tondu, P. Lopez, Modeling and control of mckibben artificial muscle robot actuators, IEEE control systems 20 (2) (2000) 15–38.

[44] E. H. Skorina, M. Luo, W. Tao, F. Chen, J. Fu, C. D. Onal, Adapting to flexibility: Model reference adaptive control of soft bending actuators, IEEE Robotics and Automation Letters 2 (2) (2017) 964–970.

[45] X. Luo, M. Zhou, Y. Xia, Q. Zhu, An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems, IEEE Transactions on Industrial Informatics 10 (2) (2014) 1273–1284.

[46] D. Holland, E. Park, P. Polygerinos, G. Bennett, C. Walsh, The soft robotics toolkit: Shared resources for research and design, Soft Robotics 1 (3) (2014) 224–230.

[47] Smooth-on inc, https://www.smooth-on.com/, accessed: 2017-07-15.

[48] Y.-L. Park, C. Majidi, R. Kramer, P. Bérard, R. Wood, Hyperelastic pressure sensing with a liquid-embedded elastomer, Journal of Micromechanics and Microengineering 20 (12) (2010) 125029.

[49] W. Felt, K. Chin, C. Remy, Contraction sensing with smart braid mckibben muscles, IEEE/ASME Transactions on Mechatronics 21 (3) (2016) 1201–1209.

[50] L. Ljung, MATLAB: System Identification Toolbox: User's Guide Version 4, The Mathworks, 1995.

Ameer Hamza Khan: Formal analysis, Investigation, Validation, Writing
Shuai Li: Conceptualization, Methodology, Resources, Supervision

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: