DEPARTMENT OF MECHANICAL ENGINEERING

TECHNISCHE UNIVERSITÄT MÜNCHEN

Thesis for obtaining the academic degree
Master of Science (M.Sc.)
at the Department of Mechanical Engineering at
Technical University of Munich

Modeling and control of hysteretic nonlinearities in robotic catheters using deep learning

Yao Zhang

DEPARTMENT OF MECHANICAL ENGINEERING

TECHNISCHE UNIVERSITÄT MÜNCHEN

Thesis for obtaining the academic degree
Master of Science (M.Sc.)
at the Department of Mechanical Engineering at
Technical University of Munich

Modeling and control of hysteretic nonlinearities in robotic catheters using deep learning

Author: Yao Zhang, B.Sc.

Supervisor: Prof. Dr.-Ing. Ralph Kennel
Chair of Electrical Drive Systems and Power Electronics

Co-supervisor: Prof. Dr. Ir. Emmanuel Vander Poorten
Department of Mechanical Engineering
Robot-Assisted Surgery group (RAS)
Katholieke Universiteit Leuven, Leuven, Belgium

Advisor: Haotian Xie, M.Sc.

Submission Date: 06 April, 2021

I confirm that this Thesis for obtaining the academic degree
Master of Science (M.Sc.)
at the Department of Mechanical Engineering at
Technical University of Munich is my own work and I have documented all sources and material used.

München, 06 April, 2021 Yao Zhang

Acknowledgments

First, I want to thank Prof. Ralph Kennel and Haotian Xie at Technische Universität München for being my supervisors at TUM and their professional, scientific advice for this thesis.

I would like to thank my master thesis co-supervisor Prof. Emmanuel Vander Poorten for allowing me to conduct research at Katholieke Universiteit Leuven, Leuven, Belgium. Under his supervision, I was able to finish this thesis and successfully carried out my first publication. Throughout the weekly meeting with him, I was growing from a freshman to a trained early-stage researcher in the research world. He could always point out the right direction to me whenever I was confused during the research. I was deeply impressed by his rigorous research spirit and passion for research. It is a great pleasure for me to have him as my co-supervisor.

I would like to thank Mr. Di Wu as my daily advisor. Mr. Wu always provided me with timely help and greatly helped for the experiment of this thesis. I would also like to thank Dr. Mouloud Ourak, Dr. Kenan Niu at the Robot-Assisted Surgery Group of KU Leuven for providing me with professional guidance, scientific suggestions of this thesis. I feel very lucky to have encountered them during my master thesis research at KU Leuven and also look forward to working with them as my colleagues in my future research career.

Finally, I want to say that nobody is more important to me than my family. I must express my thankful heart to my parents Runwei Zhang and Fangying Wang for their continuous encouragement and self-giving love throughout my master's study. It is their support that allows me to grow up and gives me infinite courage to face challenges. I would also like to give my heartfelt thanks to my girlfriend Weiwei Wang. It was her love that relieved me of the fatigue of traveling between Leuven and Munich. It was also her love that raised me up when I got weary and suffered some setbacks. Thanks to my parents and my girlfriend again for making this thesis and this unique, wonderful experience possible.

Yao Zhang

München, Bayern
April, 2021

Abstract

Comparing to open surgery, Minimally Invasive Surgery (MIS) has less operative trauma, faster recovery time, and reduced pain. Percutaneous Coronary Intervention (PCI) is one of the MIS procedures, which is commonly used for treating Coronary Artery Disease (CAD). During a PCI procedure, catheters are steered to recanalize the coronary arteries and open up the occlusion. However, the positioning precision of the catheter tip is negatively affected by hysteresis. Improper motion and potential puncture of the vessels can be fatal.

To avoid unnecessary tissue damage due to the imprecise positioning, the governing hysteresis needs to be understood and compensated for. This work investigates the feasibility to model and compensate for the hysteresis of catheters with a Long Short-Term Memory (LSTM) network. To access the value of the LSTM approach, a Deadband Rate-Dependent Prandtl-Ishlinskii (DRDPI) model and a Support Vector Regression (SVR) model were established. Hysteresis data is collected from a bench-top Pneumatic Artificial Muscle (PAM) actuated catheter. Afterwards, the DRDPI, the SVR, and the LSTM models are validated on this experimental setup.

The LSTM was tested using four groups of test data sets containing diverse patterns. The results demonstrated that the LSTM is able to predict the tip bending angle with sub-degree precision. The LSTM outperformed the SVR model and the DRDPI model by 60.1% and 36.0%, in arbitrarily varying signals, respectively.

Furthermore, a Control Long Short-Term Memory (c-LSTM) model and an Inverse Deadband Rate-Dependent Prandtl-Ishlinskii (IDRDPI) model were proposed and validated. To compensate the hysteresis, four different types of trajectories were tested on these proposed models. The results demonstrated that the c-LSTM model outperformed the IDRDPI model by at least 75% and is able to predict the catheter distal response merely based on proximal input without including sensory feedback. The proposed c-LSTM makes it possible to steer catheters with high precision in free space, which enables a few clinical applications e.g. steering catheters with high accuracy during endovascular interventions. While demonstrated upon hysteretic PAMs, the proposed method is believed to be fairly general and also applicable to other drive methods such as cable-based or SMA-actuation which are also known to exhibit non-negligible friction and are commonly used to drive robotic catheters.

Contents

Acknowledgments **iii**

Abstract **iv**

1 Introduction **1**

2 The State of the Art **5**
- 2.1 Literature research method 5
- 2.2 Literature review 6
- 2.3 Drawbacks of the prior art 10

3 Experimental setup **11**
- 3.1 Description of the PAM actuated catheter 11
  - 3.1.1 Pneumatic Artificial Muscle 11
  - 3.1.2 Bending segment 12
- 3.2 Overview on the entire experimental setup 14

4 Experimental data acquisition and characterization **17**
- 4.1 Experimental data acquisition 17
  - 4.1.1 Input data type selection 17
  - 4.1.2 Data acquisition GUI 17
- 4.2 Experimental data characterization 19

5 Deadband Rate-Dependent Prandtl-Ishlinskii model **22**
- 5.1 Overview of the DRDPI model 22
- 5.2 Threshold and deadband functions of the DRDPI model 24
- 5.3 Hysteresis modeling with the DRDPI model 25
  - 5.3.1 Unknown parameter identification 25
  - 5.3.2 Training of the proposed DRDPI model 27
- 5.4 Modeling performance of the DRDPI model 30
  - 5.4.1 Definition of metrics 30
  - 5.4.2 Performance evaluation of the DRDPI model 30
  - 5.4.3 Discussion of the DRDPI model performance 33

6 Support Vector Regression model **35**
- 6.1 Overview of the SVR model 35
- 6.2 Hyperparameters of the SVR model 36

6.3 Hysteresis modeling with the SVR model 37
6.4 Modeling performance of the SVR model 38
6.4.1 Performance evaluation of the SVR model 38
6.4.2 Discussion of the SVR model performance 42

7 Long Short-Term Memory model 43
7.1 Overview of the LSTM model 43
7.2 Construction of the LSTM model 45
7.2.1 Data preprocessing for the LSTM model 45
7.2.2 Hyperparameters of the LSTM model 46
7.3 Hysteresis modeling with the LSTM model 48
7.4 Modeling performance of the LSTM model 50
7.4.1 Performance evaluation of the LSTM model 50
7.4.2 Discussion of the LSTM model performance 53

8 Comparison of modeling performance between proposed models 54

9 Inverse Deadband Rate-Dependent Prandtl-Ishlinskii model 60
9.1 Overview of the Inverse DRDPI model 60
9.1.1 Proposed unknown functions of the IDRDPI model 60
9.1.2 Preliminary test of the IDRDPI model 62
9.2 Optimization of the Inverse DRDPI model 62
9.3 Performance of the Inverse DRDPI model for position control 65
9.3.1 Performance evaluation of the IDRDPI model 65
9.3.2 Discussion of the IDRDPI model performance 70

10 Control Long Short-Term Memory model 73
10.1 Structure of the c-LSTM model 73
10.2 Hyperparameters of the c-LSTM model 75
10.3 Position control with the c-LSTM model 76
10.3.1 Training process of the c-LSTM model 76
10.3.2 Position control performance of the c-LSTM model 76
10.3.3 Discussion of the c-LSTM model performance 82

11 Comparison of control performance between proposed models 84

12 Discussion and conclusion 91
12.1 Conclusion 91
12.2 Future work 92

List of Figures 93

List of Tables 96

Bibliography 99

1 Introduction

Coronary Artery Disease (CAD) is one of the most prevalent forms of heart disease. It is one of the leading causes of death worldwide. It is the leading cause of death in the United States in both men and women. One-third of deaths over 35 years old have been attributed to CAD [1]. CAD happens when the arteries that supply blood to the heart muscle become narrow and close. Atherosclerosis is responsible for occlusion of the coronary. The buildup of cholesterol and other material on the inner walls is called plaque (see Figure 1.1).

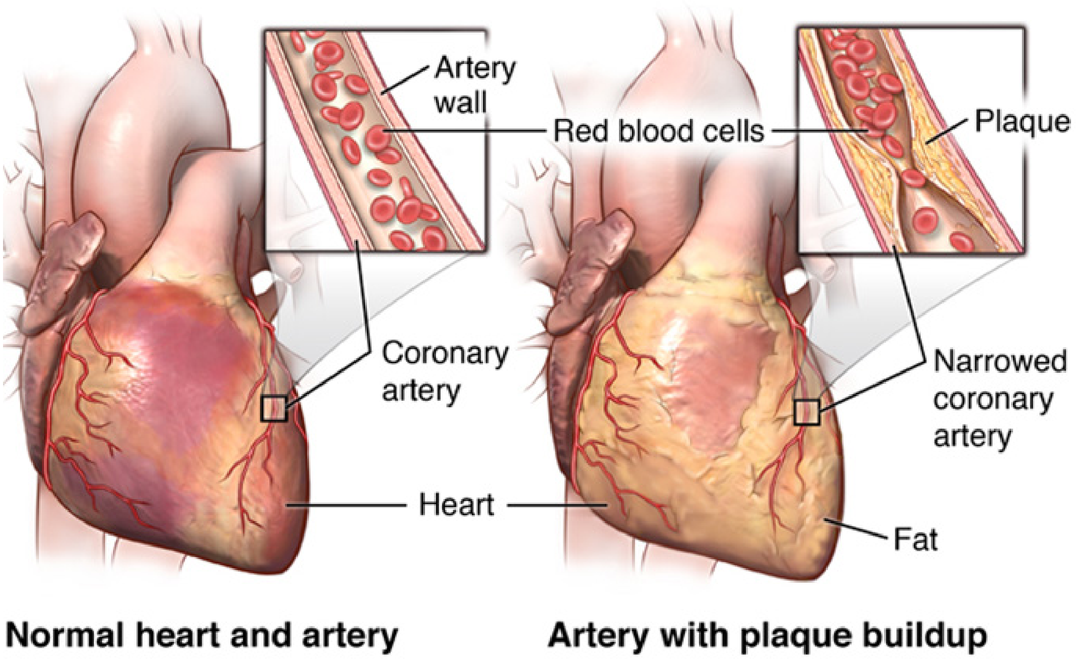

Figure 1.1: Coronary Artery Disease (CAD) [2]

The coronary arteries supply blood, oxygen and nutrients to the heart muscle. A buildup of plaque can narrow these arteries, decreasing blood flow and oxygen supply to the heart muscle. If left untreated, the reduced blood flow may cause chest pain (angina), shortness of breath, or other coronary artery disease signs and symptoms. A complete blockage can cause a heart attack, even sudden death [3].

To treat CAD patients in a minimal invasive fashion, treatment with guide wires and catheters, a so-called Percutaneous Coronary Intervention (PCI), can be adopted. Comparing to open surgery, MIS has less operative trauma, faster recovery time, and reduced pain. Especially patients with a high risk for perioperative complications may benefit from PCI [4]. During PCI, the catheters access the vasculature through the femoral or radial artery and are moved up to the coronaries to recanalize the occlusion. The procedure of PCI can be summarised as follows (Figure 1.2) [5]:

- a catheter is inserted into the blood vessel either at the groin or at the arms;
- using a special type of X-ray called fluoroscopy, the catheter is moved in the blood vessels surrounding the heart where the coronary artery is narrowed;
- when the tip is in place, a balloon tip covered with a stent is inflated;
- the balloon tip compresses the plaque and expands the stent;
- once the plaque is compressed and the stent is in place, the balloon is deflated and withdrawn.
- the stent stays in the artery, holding it open (Figure 1.3).

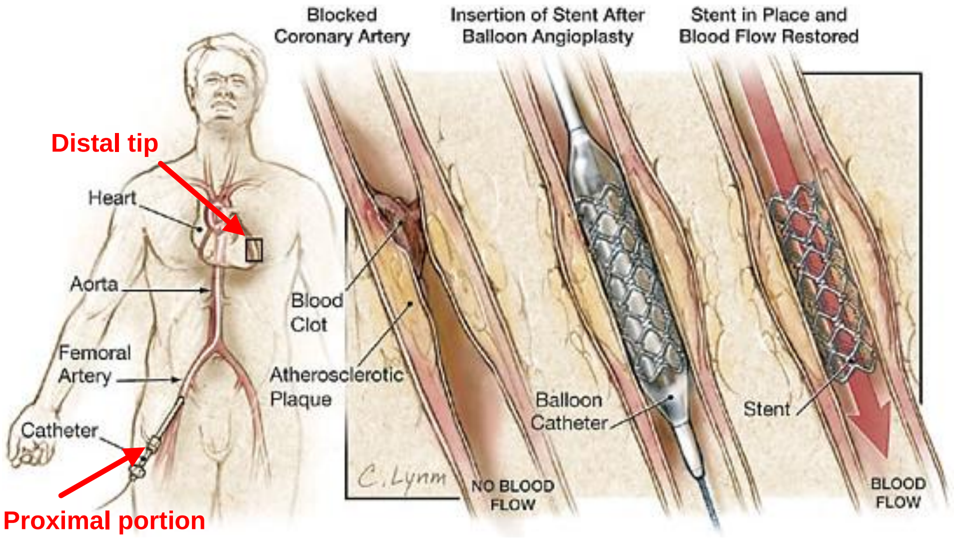

Figure 1.2: Percutaneous Coronary Intervention (PCI) Procedure and Recanalization of the occlusion [6]

Due to the tortuosity of the aorta, the fragile and deformable nature of the vessels as well as heartbeat, good maneuverability and controllability of the catheters are imperative. Robotic catheters, which can offer improved ergonomics and improved dexterity compared to the conventional manual catheters have been developed [8]. However, the positioning precision of the catheter remains an important challenge.

The reason why catheters are hard to control accurately include: external effects, such as blood flow and unpredictable contacts with the vessel wall. Another aspect is internal namely

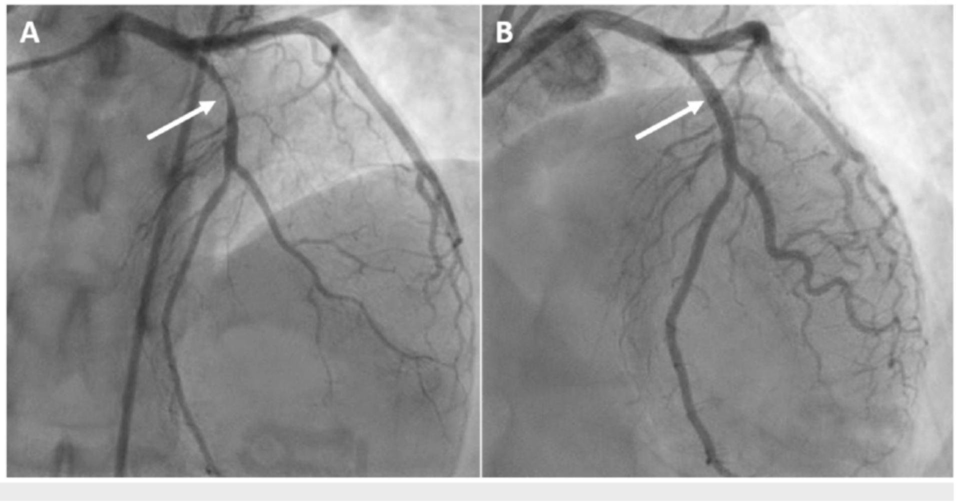

Figure 1.3: X-Ray imaging of Recanalization. **A.** Before the PCI Recanalization. **B.** After the PCI Recanalization [7].

the nonlinear behaviour in the drive system which is largely affected by hysteresis. This thesis mainly focuses on the internal hysteresis of the catheters, which is to be understood and compensated for. In this thesis, Long Short-Term Memory (LSTM), a deep learning method is established to model the hysteresis of a Pneumatic Artificial Muscle (PAM) actuated robotic catheter. In addition, to compare with the proposed LSTM model, another learning-based Support Vector Regression (SVR) model and an analytical model Deadband Rate-Dependent Prandtl-Ishlinskii (DRDPI) were also established. After the modeling process, two models namely Inverse Deadband Rate-Dependent Prandtl-Ishlinskii (IDRDPI) and Control Long Short-Term Memory (c-LSTM), are also established. The ability of these methods to compensate the hysteresis and achieve accurate position control of the robotic catheter system is investigated. The main goal of this thesis is:

- improving the positioning accuracy of robotic catheters;

and sub-goals are:

- collecting bending angle/pressure data form an experimental setup including a catheter segment;
- modeling the hysteresis in a PAM-driven catheter based on the LSTM model;
- modeling the hysteresis in a PAM-driven catheter based on the SVR model;
- modeling the hysteresis in a PAM-driven catheter based on the DRDPI model;
- comparing the modeling performance between the LSTM, SVR, and DRDPI model;
- validating the position control performance of the IDRDPI and c-LSTM model on the compensation of the hysteresis of the PAM-driven catheter.

Part of this work was supported by the ATLAS project. ATLAS received funding from the European Union’s Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 813782.

2 The State of the Art

2.1 Literature research method

In recent years, Minimally Invasive Surgery (MIS) has grown rapidly because it offers smaller scars, faster recovery time, and reduced pain. During MIS procedure, in particular with treating for Coronary Artery Disease (CAD) through PCI, catheters are very important tools. Limited positioning accuracy is well-known problem. Before engaging in activities to improve positioning ability, it is important to get an overview of prior work that addressed this problem. To obtain such overview a high-level survey of the state-of-the-art was conducted first.

The search process was programmatically generated by using the Scopus® Application Programming Interface (API). Before the search, a search matrix was generated, which is presented in Figure 2.1. The matrix consists of four columns that are combined with special operators. The first column named object represents the character of the hysteresis. In this column, four objects are given, namely: hysteresis, which this thesis mainly focused on; backlash, a very common and simply model for hysteresis modeling; dead-zone, which is a widely world phenomenon and will be discussed later in this thesis; non-linear, which is a common denomination capturing all these phenomena. The symbol * means a potential extension of non-linear, e.g., to non-linearity. This symbol expands the searching process and will also used in the following columns.

The second 'method' column, indicates the main purpose of the search. In this column, four items are listed: model* represents the potential methods to model hysteresis. These methods could be considered for adoption in the thesis. The second item compensat* collects words that describe compensate and its potential extensions. The reason why this item is added here, is because hysteresis should be compensated to reach a precise control. The last two items, namely identifi* and control*, search for words that describe the identification process of the modeling and control to compensate hysteresis.

The third column lists relevant intervention type where similar medical devices are used. Seven intervention types were identified - cardic, endovascular*, endoscop*, urteroscop*, colonoscop*, percutaneous*, laparoscop*.

The fourth column captures the relevant medical devices where hysteresis can be found. In this column, three devices are listed - catheter*, *scope, robot*. This thesis focuses on hysteresis in the catheter, however, solution for closely related systems may also be useful for this work.

To connect four columns, the logic AND operator is used. Between each element of a column, a logic OR is adopted. After preparing this search matrix, it is set up with the API of Scopus®. Search queries are programmatically implemented and the research results are retrieved and checked for duplicates via the scopus API. The list of the research results are saved as a .csv file and manually evaluated by the author.

There are in total 566 results carried out based on the search matrix in Figure 2.1, after manually evaluated and selected, all relevant literature will be introduced in the following section.

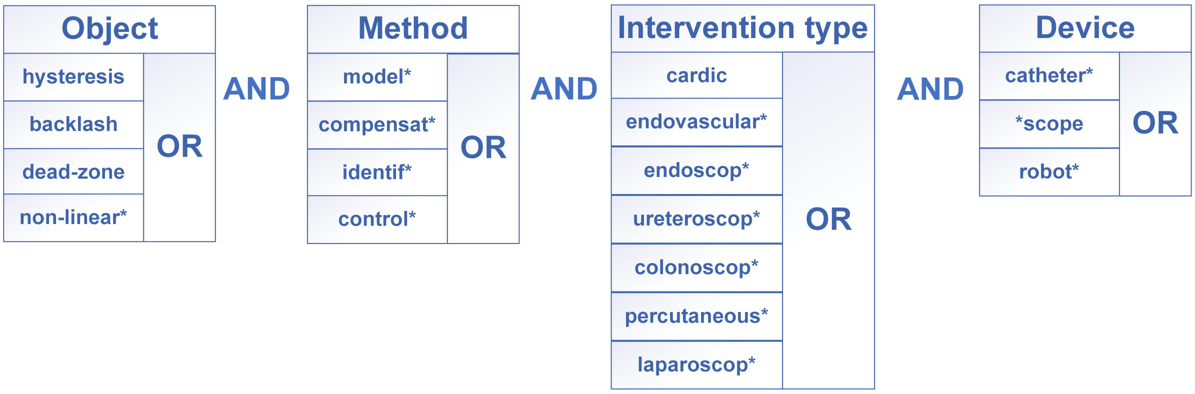

Figure 2.1: Search matrix of the conducted literature review in this thesis

2.2 Literature review

To effectively treat CAD disease, the PCI procedure is commonly used [1]. There are many possibilities of flexible tools for this type of treatment, such as steerable needles [9], endoscopes [10], and catheters [11]. During PCI catheters are steered to recanalize the occlusion. The coronary arteries supply blood, oxygen, and nutrients to heart muscle, which is very important for our body [4]. However, because of the tortuosity of the aorta, the deformable fragile nature of the vessels as well as the heartbeat, good maneuverability, and controllability of the catheters are imperative [3]. Robotic catheters, which can improve ergonomics and higher-dexterity compared to conventional manual catheters, are gradually gaining attention to address these challenges [8].

Robotic catheters can be operated based on various working principles, such as Cable-Driven (CD) [12], Tendon-Sheath Mechanism (TSM) [13], and McKibben muscles [14]. Cable-Driven technology is the most popular driving principle for robotic catheters. The cables are routed through the entire length of the catheters, but experience high levels of friction with their guiding tubes. Because of this friction, CD robotic catheters face difficulty to reach a large bandwidth. This hinders the development of CD technology when good responsiveness is desirable. To solve this problem, from the 1950s, Pneumatic Artificial Muscles (PAMs) became popular for many application. They have a high operation bandwidth, easy fabrication, and low-cost [15]. Besides, PAMs show good promise for using in medical applications e.g., used in interventional instruments, in which high accuracy is required [16].

Besides challenges in the design of catheter robotics [17], precise control is also challenging regardless whether actuated by CD or PAMs. Hysteresis focus a fusion source of imprecision

Table 2.1: Methods to minimize hysteresis in flexible surgical robots from previous literature

Authors and publication year	Device	Actuation/Transmission Principle	Methods
Bardou *et al.*, 2012 [18]	Endoscope	Cable-Driven (CD)	Feedback from Electromagnetic (EM) sensor
Roesthuis *et al.*, 2013 [19]	Flexible Surgical Instrument	CD	Feedback from FBG sensor
Reilink *et al.*, 2013 [20]	Endoscope	CD	Locating catheter tip using imaging modalities
Cabras *et al.*, 2017 [21]	Endoscope	CD	Locating catheter tip using imaging modalities
Baek *et al.*, 2020 [22]	Multi-arm Flexible Surgical Robot	Tendon-sheath mechanism (TSM)	Computer vision + kinematic model
Do *et al.*, 2014 [23]	Endoscope	TSM	Bouc–Wen model
Wang *et al.*, 2020 [24]	Ureteroscopy	CD	Coleman-Hodgdon model
Sun *et al.*, 2013 [25]	Flexible Surgical Robot	CD	Tendon Elongation model
Omisore *et al.*, 2018 [26]	Cardiovascular Catheter	CD	Adaptive Backlash Model
Legrand *et al.*, 2020 [27]	Fetoscope	Mckibben muscle	Generalized Prandtl-Ishlinskii Model
Xu *et al.*, 2017 [28]	Serpentine Manipulator	TSM	Machine learning (regression)
Porto *et al.*, 2019 [29]	Endoscope	CD	Machine learning + Inverse kinematic Model

of catheter positioning. Hysteresis establishes a complex non-linear and multi-valued relationship between the input signals and the response of the catheter distal tip (see Figure 2.2). This non-linear and multi-valued relationship leads to inaccurate control of catheters when navigation and positioning the tip of the catheter. Inaccurate positioning of the catheter's tip could induce thrombus or damage the vessel wall since the aorta and the vessel wall are fragile. To address this issue, several methods have been explored in the past (Table. 2.1).

Using the external sensors as feedback to implement a closed-loop control was presented in [18], [30]. In [18] and [30], Electromagnetic (EM) sensors are used to provide the feedback to a CD endoscope. However, mounting sensors at the distal tip of the catheters is challenging due to spatial restrictions and sterilization requirements [29]. Imaging techniques, e.g. ultrasound, especially A-mode ultrasound imaging [31] have great potential to be used to get real-time intraoperative data [32] and for control feedback [20], [21] and [33]. Furthermore, these imaging-based methods are also integrated with kinematic models [22].

Amongst others, analytic models are being explored to model the hysteresis. These models can be classified into two types: 1) operator-based (the models use operators to characterize the hysteresis) and 2) differential-based (the models use differential equations to model the hysteresis) [34]. Based on the literature review in this thesis, three analytical models are introduced by describing three commonly used models 1). Preisach, 2). Bouc-Wen and 3) Prandtl-Ishlinskii model.

1. Preisach model

   The Preisach model is one of the most popular models in the hysteresis modeling and compensation. This model is widely used to model smart materials. The Preisach model

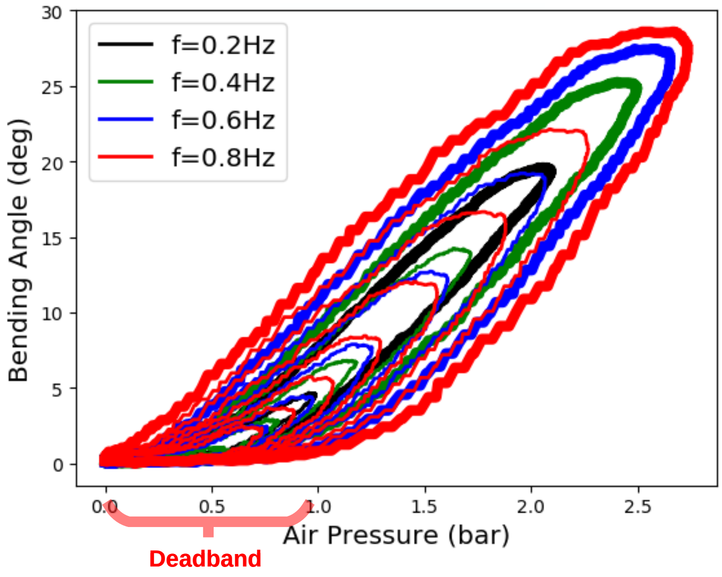

Figure 2.2: Hysteresis with deadband

shows good performance at deadband region of the hysteresis. Besides, it also can give good performance at lower bandwidth. Specially under the no-load condition. However, the accuracy of this model gradually decreases when pre-loading and the input frequency are increased [35].

2. Bouc-Wen model

   The mainly difference between the Preisach and the Bouc-Wen model is that the Preisach is an operator-based model but the Bouc-Wen model is a differential-based model [36]. The Bouc-Wen model is a rate-independent model, however, hysteresis is a rate-dependent phenomenon [37]. Besides, the Bouc-Wen model is not invertible [34]. To model the rate-dependent hysteresis, a linear Hammerstein model is combined with a Bouc-Wen model by Wang *et al.* [38]. On the other hand, to inverse the Bouc-Wen model, a Bouc-Wen least square support vector machine (LS-SVM) is adapted to identify or compensate the hysteresis by Rakotondrabe *et al.* [39].

3. Prandtl-Ishlinskii (PI) model

   Finally, an overview of some Prandtl-Ishlinskii (PI) models is given in Figure 2.3. In general, PI models can be categorized based on two criteria. Whether the model uses a generalized play operator with envelop function or deadband operator or not, PI models can be divided into generalized and non-generalized PI models

   - rate-independent PI model

     In [41], a rate-independent Prandtl-Ishlinskii (PI) model was used to characterize the hysteresis of piezoceramic actuators that operate at low frequency.

   - generalized rate-independent PI model

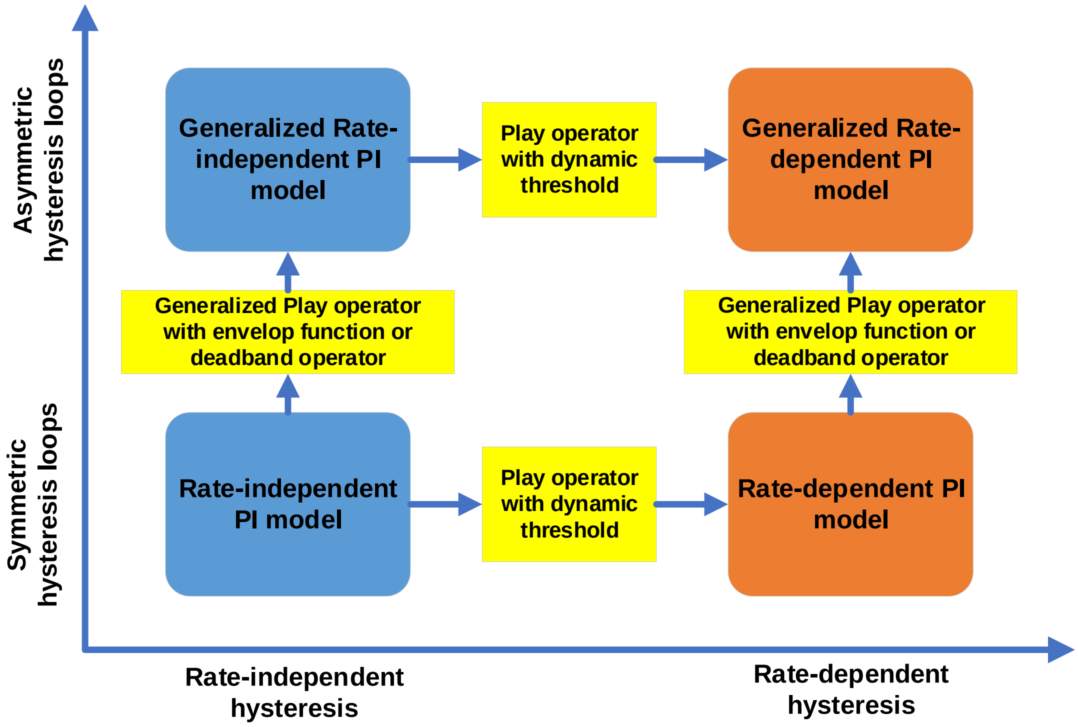

Figure 2.3: The different implementations of Prandtl-Ishlinskii models [40]

To model asymmetric rate-independent hysteresis, a generalized play operator with envelop function was used in [42].

- rate-dependent PI model

  To model rate-dependent hysteresis, a play operator with dynamic threshold was used in [43]. In [44], this model was used to model the hysteresis nonlinearities of magnetostrictive actuators.

- generalized rate-dependent PI model

  To model the asymmetry and deadband hysteresis, a generalized rate-dependent PI model was used in the [42] with envelop function and deadband operators.

In recent years, deep learning techniques have gained many interests because they can avoid complicate identificated process by simply training an artificial neural network [45], [46], [47], and [48]. Up to now, some prior works also works attempted to use machine learning or deep learning to model and compensate for hysteresis. In [28], a regression method to learn the inverse kinematics model of a serpentine surgical manipulator was employed. Besides, [29] used machine learning methods to achieve accurate position control of a flexible surgical robot. Both above-mentioned works dealt with cable-driven robots, however, they adopted conventional machine learning methods rather than taking benefit of recent advances in deep learning. This thesis explores the feasibility value to employ deep learning methods to model and compensate for hysteresis. Besides, the analytic models and machine learning models will also be established to allow a fair assessment of the value of deep learning methods for this application.

2.3 Drawbacks of the prior art

In the following some restrictions of current hysteresis modeling and compensation methods are described. In [18], [30], external sensors such as Electromagnetic (EM) sensors were implemented to achieve closed-loop control. However, mounting sensors at the tip of the catheter is challenging due to the size of the sensors and sterilization requirements [29]. Imaging techniques could also help for control feedback [20]. These techniques are used in [20] and [21]. Cabras *et al.* used imaging-based methods for modeling and compensation of the hysteresis and integrated these with kinematic models [22]. Compared to sensors technique, the imaging technique do not need to mount the sensors on the tip of the catheters. However, the training process of the imaging technique is complicated because of the huge computational cost. Besides, the imaging support device is an additional requirement for the imaging acquisition.

In [35], the Preisach model is demonstrated. This model shows good performance at deadband region of the hysteresis. Nevertheless, the accuracy of the model decreases when the pre-loading and the input frequency increases. Another model named Bouc-Wen model, which also combined with a linear Hammerstein model, is introduced in [36]. However, the Bouc-Wen model can not be used directly to compensate for the hysteresis unless least square support vector machine is added to this model, since this model is cannot be inversed.

A rate-independent PI model is introduced to model the hysteresis of piezoceramic [41]. However, this model is limited because the hysteresis is rate dependent. Furthermore, if the hysteresis pattern is asymmetric, the model will show a bad performance. Despite a generalized version of rate-independent model that is proposed in [42], the rate-dependent problem is still not solved. A rate-dependent PI model is described in [43]. This model can solve the rate dependent hysteresis problem. Furthermore, a generalized version of rate-dependent PI model is introduced in [42]. The Prandtl-Ishlinskii (PI) model is composed of a lot of mathematical formulations and need a long training time for the identification process. Given that it is arguable that the most sophisticated method described in literature, it will be used as a reference in this work.

3 Experimental setup

3.1 Description of the PAM actuated catheter

3.1.1 Pneumatic Artificial Muscle

From 1950s, the use of Pneumatic Artificial Muscle (PAMs) became more popular because of its advantages: high bandwidth, easy fabrication and low cost [15]. During the medical intervention, a precise control is required, PAMs could show good performance [16]. Due to the advantages of PAMs, this thesis will focus on catheter with embedded with PAMs. Such catheters could operate at higher bandwidth, and greater precision provided that its hysteretic behaviour were compensated adequately. The structure of the PAM can be divided into five parts (shown in Figure 3.1):

1. non-inflatable tube;
2. ferrule;
3. balloon;
4. braid;
5. steer wire.

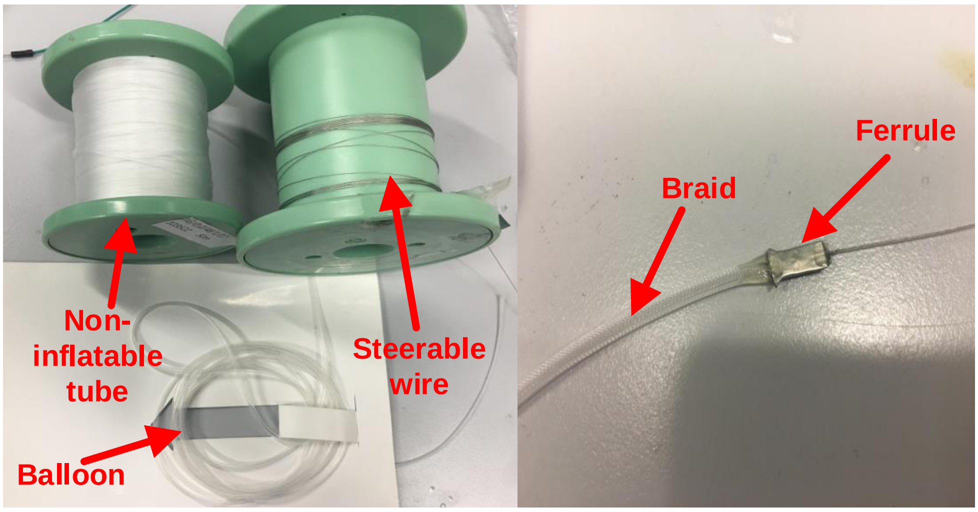

Figure 3.1: Five parts of PAM

The Non-inflatable tube is a plastic tube, which is used to supply the compressed air to the artificial muscle. Because this tube is non-inflatable, the air pressure of the compressed air can be supplied accurately to the muscle. To ensure air leaks is avoided, the ferrule, placed at the tip of the artificial muscle, closes the tube hermetically. The balloon dilates and contracts when the air pressure rises. To prevent extreme dilation of the balloon, a braid is sleeved outside the balloon. The braid provides resistance to protect the balloon against over this dilation. The length of the artificial muscle shortens when the muscle is dilated. This generates a pulling force along the muscle axis. To pass this force to the bending segment of the catheter, a short steering wire is fixed off-center at the tip of the robotic catheters. The steerable wire is in-extensible and can pass on the force almost lossless to the bending segment.

3.1.2 Bending segment

The bending segment is a one degree-of-freedom (DOF) unidirectional catheter distal segment with an embedded PAM. This bending segment is fabricated out of Nitinol using metal laser cutting technology. In this thesis, a bending section with 4.4 mm diameter is used (shown in Figure 3.2). The slots lasered in the Nitinol tube increase the compliancy which allows larger curvatures and bending angles for this robotic catheter.

Figure 3.2: Bending segment

To measure the actual bending angle of the robotic catheter, a laser distance sensor was positioned next to the tip of the bending segment. A light-weight black plastic piece was mounted at the tip of the catheter. It can reflect the light from the laser distance sensor. The structure of this bending angle measurement as shown in Figure 3.3.

In Figure 3.3, a laser photoelectric sensor (OADM 12I6460/S35A, Baumer Group, Switzerland) is used to measure the distance s (unit: mm) from the catheter tip to the sensor with a sampling frequency of 250 Hz. The measured voltage output U (unit: V) from the laser

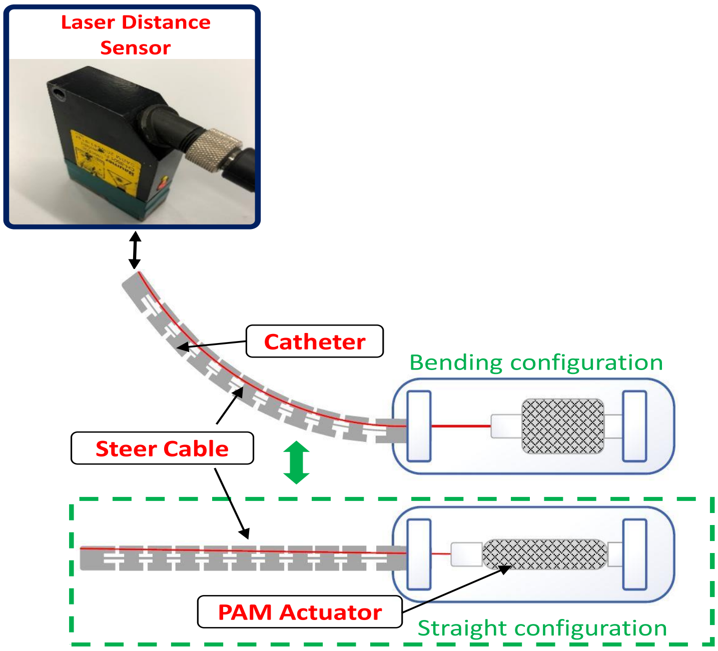

Figure 3.3: Bending angle measurement

distance sensor can be converted to the catheter tip bending angle based on the following relation:

$$s_i = mU_i + n \tag{3.1}$$

$$d_i^{'} = \max(s_1, s_2, \cdots, s_N) - s_i \tag{3.2}$$

$$\theta_i = \frac{2 \cdot d_i^{'} \cdot L}{H^2 + d_i^{'2}} \cdot \frac{180}{\pi} \tag{3.3}$$

In (3.1), subscript i indicates the i-th sample of the group and the N represents the number of samples in the data group. The measured voltage U from the laser photoelectric sensor is proportional to the distance s to the catheter tip, also the m = 4.05 (mm/V) and n = 29.36 (mm) in (3.1) was identified from the data collection process, which will be introduced in the next chapter. As for (3.2), the maximum distance between the sensor and the catheter tip is reached when the catheter is straight, while the minimum distance is reached when the maximum pressure is input into the catheter. The straight configuration is referenced to as zero displacement, then the displacement of catheter tip $d^{'}$ can be calculated by subtracting the measured distance **s** from the maximum distance $\max(s_1, s_2, \cdots, s_N)$ between the laser sensor and the catheter tip as expressed in (3.2).

However, the bending distance is not the parameter that we want to collect from the laser sensor. The bending angle of the catheter is the final parameter we want to collect in this thesis. Thus, a conversion between the bending distance and bending angle should be established. To address this problem, a constant curvature model assumption is proposed in [49]. This assumption is motivated by the fact that the catheter distal tip is lightweight, furthermore the targeted operation speed and variations thereof are low for safety reasons. Also at this point no external forces are assumed as the catheter works in free space (shown in Figure 3.4). Based on the proposed model, a geometrical conversion between the distance and the bending angle as given in 3.3 can be found. For the distal section used here we have: $L = 77$ mm and $H = 65$ mm.

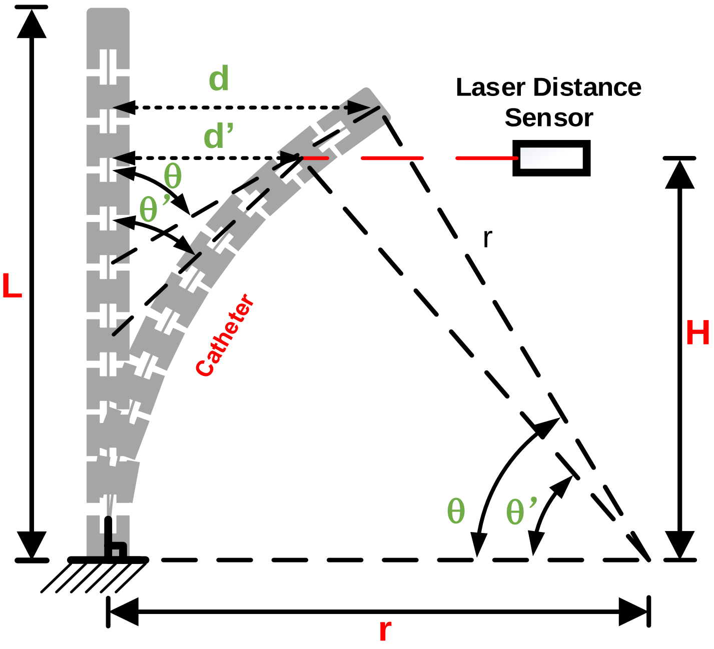

Figure 3.4: The catheter tip displacement $d^{'}$ captured by the laser distance sensor is converted to a bending angle θ using (3.1) - (3.3) based on a constant curvature model [49] assumption.

3.2 Overview on the entire experimental setup

The whole setup can be divided into two main parts, namely the hardware part and the software part as explained next:

- Hardware

  In order to accurately control the input pressure, the pressure is fed by an air supply through a proportional pressure valve (Festo, Germany) to the artificial muscle in cascade (shown in Figure 3.5). A proportional valve and a pressure sensor (21Yseries, Keller, Switzerland) are installed in series with the above-mentioned circuit to measure the pressure. The OADM laser distance sensor is used to capture the motion of the catheter tip. The obtained data will be collected and converted via (3.1) - (3.3).

- Software

  To accurately control the input pressure and obtain the output, a software for supporting the communication between the different modules is adapted. LabVIEW is a powerful system-design platform and development tool for a visual programming language from National Instruments (NI®). In this thesis, this software is used to support the experiment.

The obtained data will be collected and converted by the methods, which has been introduced in the last section. The characterization of the acquisition data will be introduced in the next chapter.

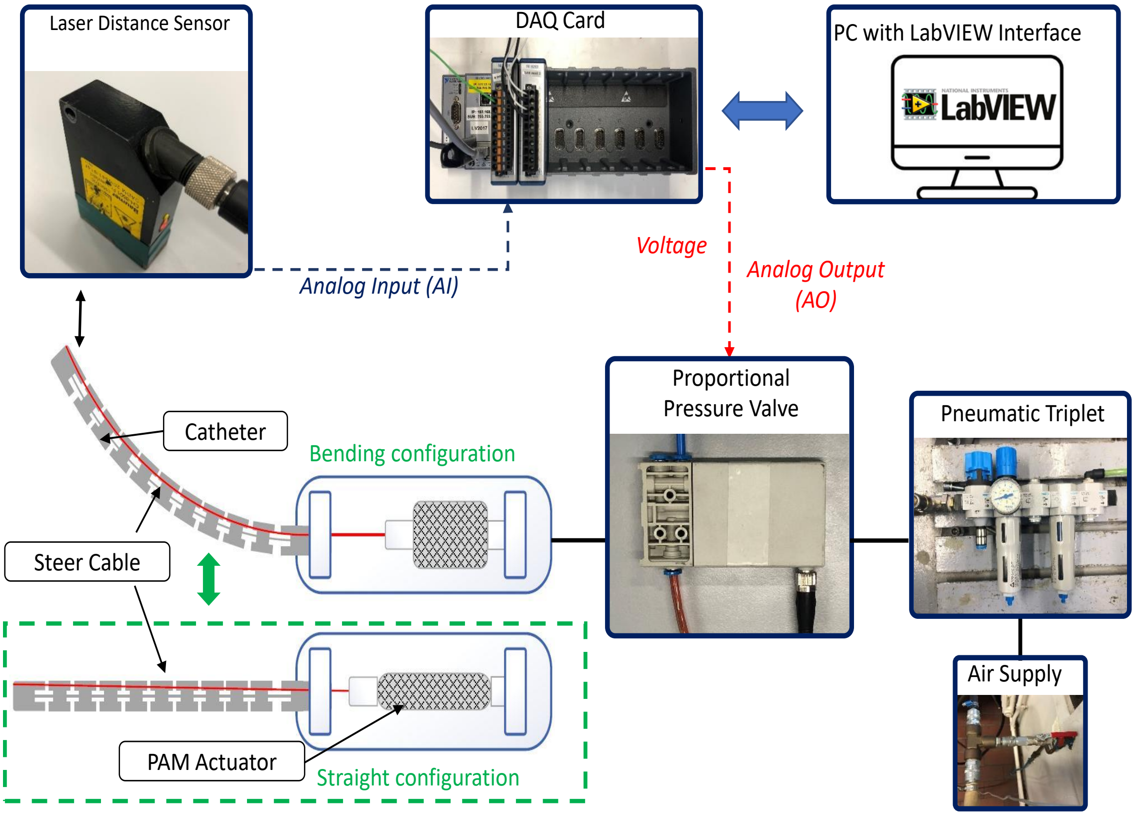

Figure 3.5: A PAM-driven catheter segment is controlled by a proportional pneumatic valve, which receives a command signal from a PC through a NI® CompactRIO system. The resulting catheter tip bending angle is captured by a laser sensor.

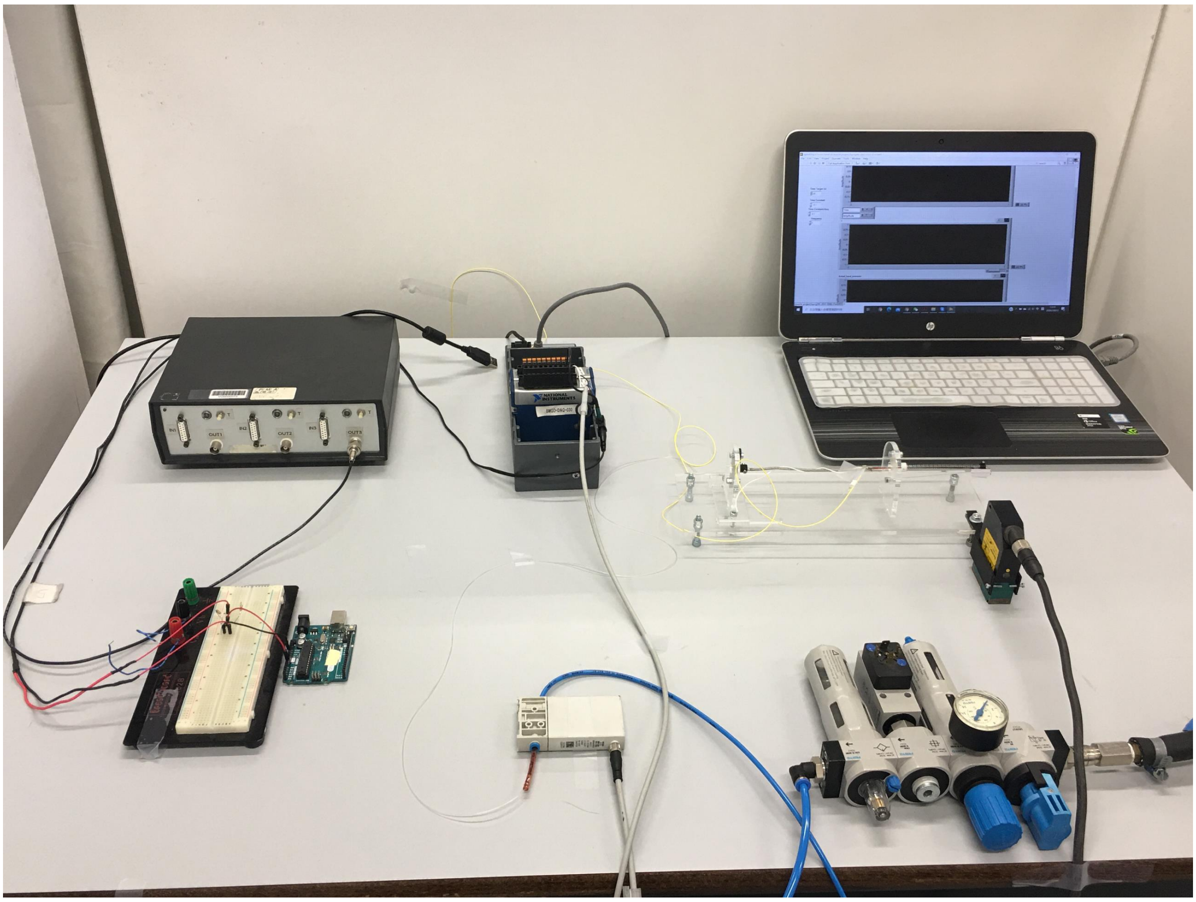

Figure 3.6: PAM-driven setup in the lab

4 Experimental data acquisition and characterization

4.1 Experimental data acquisition

4.1.1 Input data type selection

In this experimental setup, the input is the pressure and the output is the bending angle of the catheter. Based on the practical using situation, the movement of the catheter can be divided into data with a zero baseline and data with a non-zero baseline. The zero baseline data means that after an excitation of the catheter, movement will return to the initial position. The non-zero baseline data means that the catheter is bent to a specific position, but does not not move back to the initial position. It is essential to characterise these two typical types data of the catheter, which would be helpful for the modeling and compensation for the hysteresis.

In order to characterise movement of the catheter and provide sufficient training data, descending sinusoidal pressure commands with zero baseline, described in (4.1), and with non-zero baseline, described in (4.2), were sent to the setup to generate multi-loop hysteresis.

$$p_1(t) = Ae^{-\tau t}(\sin(2\pi f t - \frac{\pi}{2}) + 1) \qquad \text{[bar]} \tag{4.1}$$

$$p_2(t) = Ae^{-\tau t}(\sin(2\pi f t - \frac{\pi}{2})) + A \qquad \text{[bar]} \tag{4.2}$$

The amplitude A of both signals equals 1.5 bar to achieve a maximum amplitude of 3 bars. In (4.1) and (4.2), the variable f is the excitation frequency expected in Hz. Input frequencies up to 1 Hz are investigated in this experiment, since in many PCI applications 1.2 Hz is already sufficient [50]. The amplitude and the frequency of input data is determined explicitly for this experiment, because as mentioned before, the hysteresis depends on both the input frequency and the amplitude of the excitation signal [50]. Thus different excitation frequencies are included in the training data, namely the frequencies of the excitation signal are set to 0.2, 0.4, 0.6 and 0.8 Hz in the training data set. For the modeling of hysteresis, only one value τ = 0.15 for the time constant to generate multi-loop hysteresis is enough [51].

4.1.2 Data acquisition GUI

In order to accurately control the input pressure, LabVIEW® was used in this experiment, Figure 4.1 shows the front panel of the LabVIEW® setup in this experiment is demonstrated. On the left hand, four variables are listed in one column. The first item is the running time of the program, which defines the duration of the experiment. The second item determines the time constant τ, which also shows up in (4.1) and (4.2). In order to provide the possibility to explore the potential of this setup, the time-constant frequency is added at the third place. With this variable, a frequency changing signal can be generated. The last item is the frequency of the input pressure, in this thesis, 0.2 Hz, 0.4 Hz, 0.6 Hz and 0.8 Hz are used to generate the input pressure series.

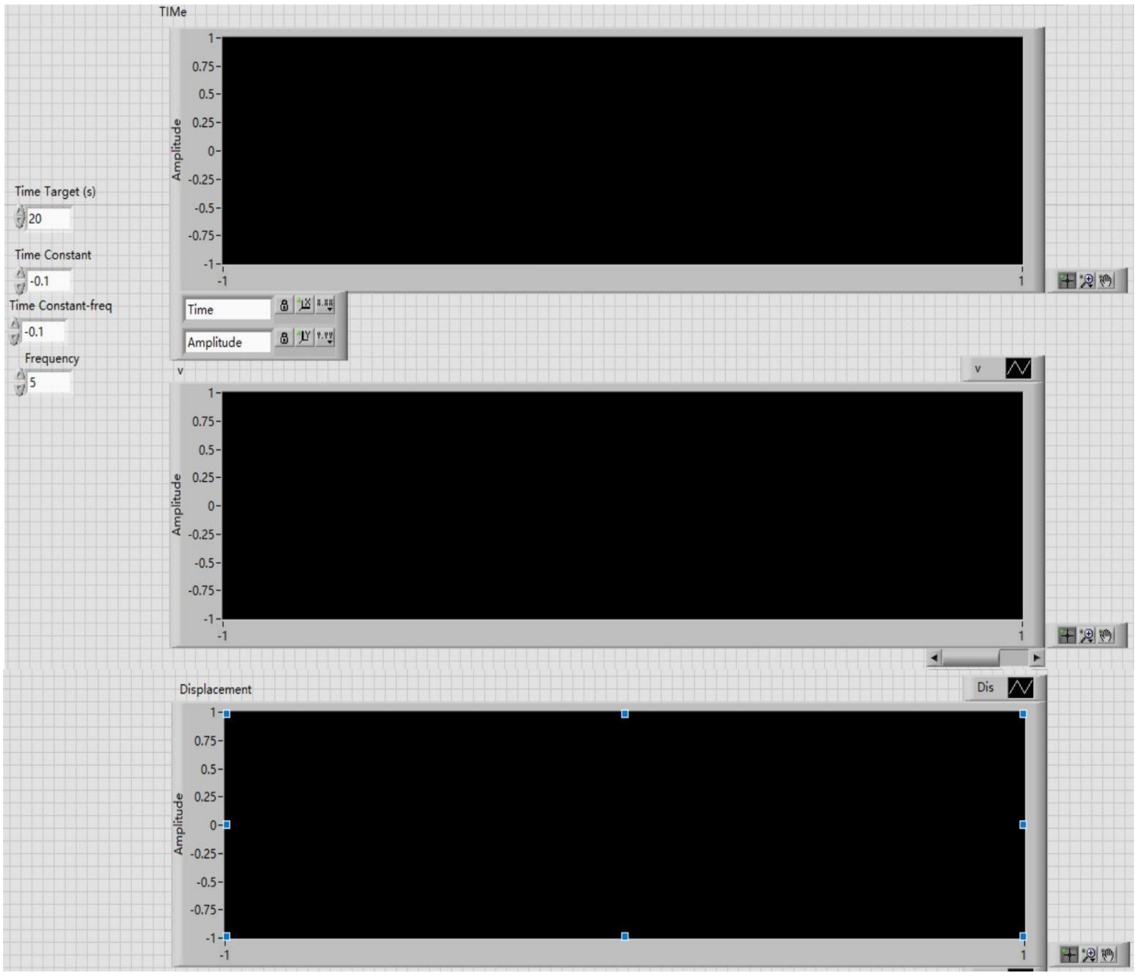
Time Target (s) 20
Time Constant -0.1
Time Constant-freq -0.1
Frequency 5

Figure 4.1: LabVIEW® Front panel

The three black displays show the duration of the experiment, the input pressure, and the displacement of the catheter tip, respectively. The transformation between the displacement and the bending angle in this setup follows (3.3). The data shown in the display can be exported with csv format.

One of the advantages of LabVIEW® is graphical programming interface. Each front panel has its block diagram, users can add components in the block diagram, and the corresponding components are shown on the front panel. Figure 4.2 shows the block diagram of the zero data generation, which represents the formula in (4.1). By connecting lines with each other

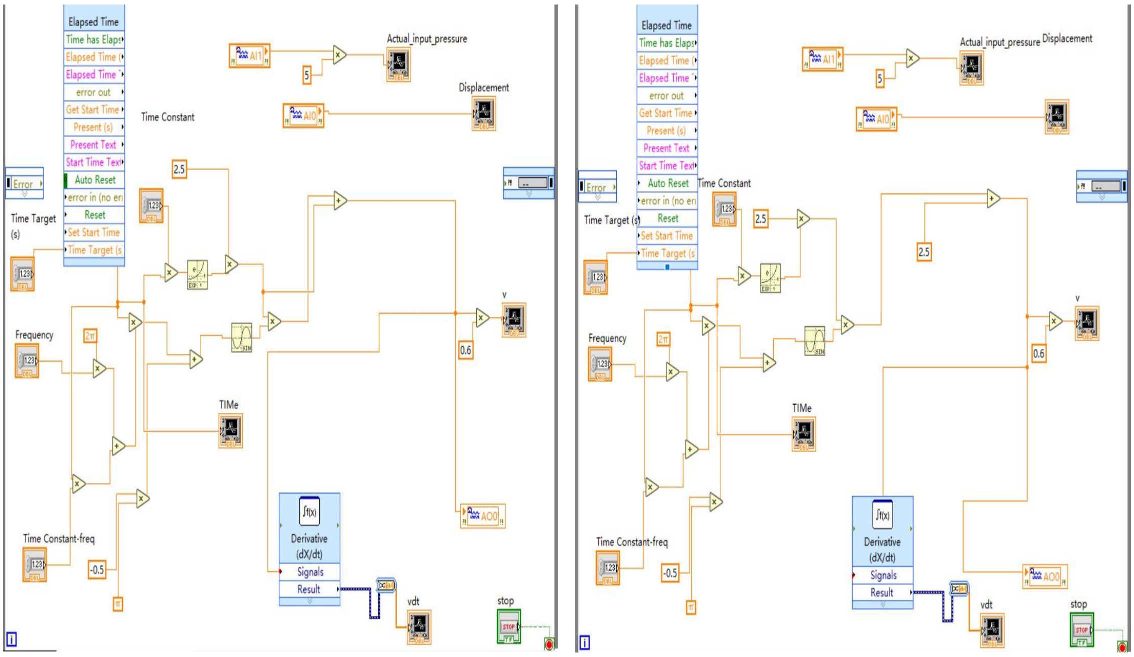

Figure 4.2: Block diagram of zero data and non-zero data

and combining different calculation operators a formula can be formed. The same coding logic of the non-zero baseline data generation is same with zero data baseline generation (shown in Figure 4.2).

The LabVIEW program provides not only the formulation of the expected formula, but also is important for the communication between the different modules in the experiment setup. The AO0 in the program is the interface of the generated input pressure signals. It sends control signals to the proportional pressure valve, which can accurately control the input pressure. On the other hand, the laser distance sensor can measure the catheter tip's bending distance. The measured data can be sent to the PC and shown in the front panel via the AI0 in the block diagram.

4.2 Experimental data characterization

The hysteresis pattern can be divided into two main types:

- Symmetric or asymmetric;
- Deadband or Non-deadband.

Based on these two types, four sub-types of hysteresis patterns can be distinguished, namely a symmetric non-deadband hysteresis pattern, asymmetric deadband, an asymmetric deadband, and finally the most complicated type: asymmetric non-deadband hysteresis pattern. The hysteresis types is determined by the physical character of the setup. Thus a characterization process of the hysteresis type for the experimental data is necessary.

As discussed in subsection 4.1.1, zero baseline type and non-zero type of data will be input into the experimental setup. Based on these two types input data, two types of output data will be collected. Figure 4.3 demonstrates the hysteresis pattern of the experimental data when the zero data is used. It is clear that the hysteresis pattern is asymmetric as well as and exhibits deadband. Because of the deadband part, which marked with red bracket, the hysteresis pattern is asymmetric. Moreover, the Figure 4.3 also shows the major and minor loops of the hysteresis type. The major loops under each frequency are drawn with wide lines and the minor loops are represented with thin lines. In this experiment, four frequencies as mentioned that 0.2 Hz (black). 0.4 Hz (green), 0.6 Hz (blue) and 0.8 Hz (red) input data are characterized in Figure 4.3. Also, with the increment of input frequency, the width of the major loops is also increasing.

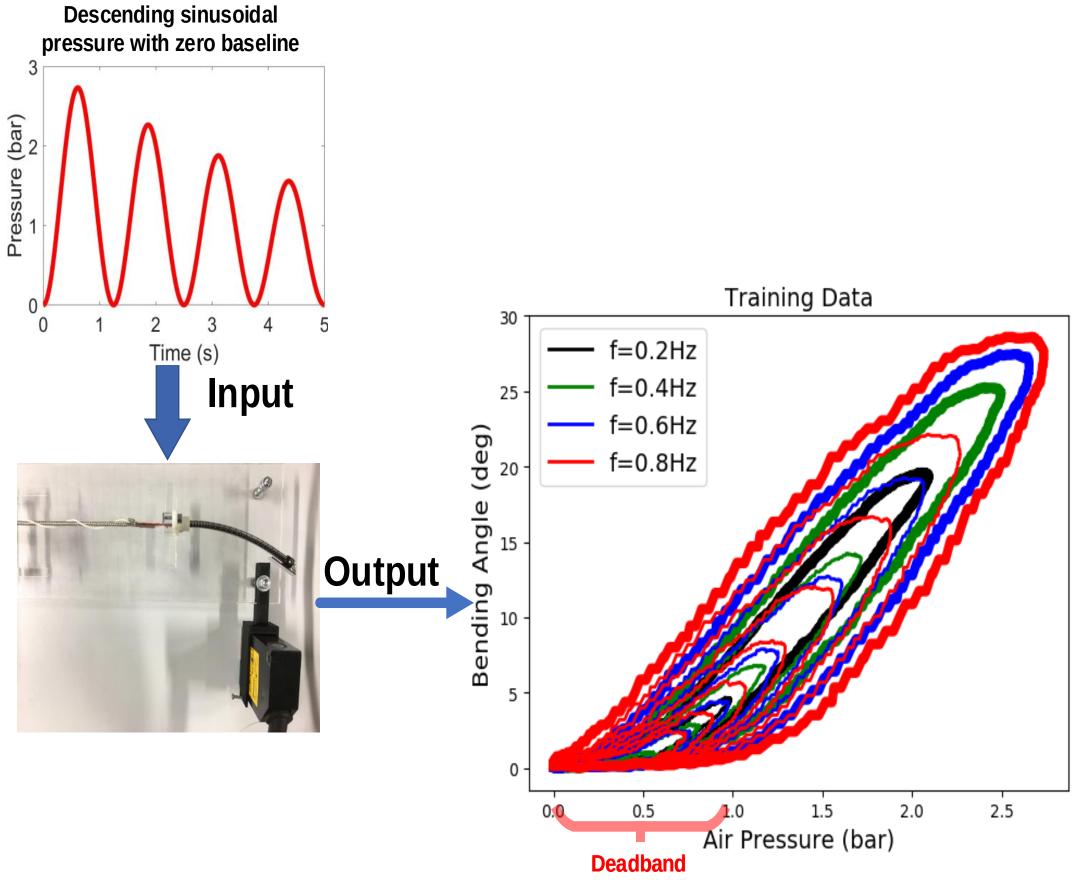

Figure 4.3: Asymmetric deadband hysteresis pattern (zero data)

Besides the zero baseline data input, non-zero data is also used to characterize the hysteresis pattern. Figure 4.4 shows the hysteresis pattern for the non-zero data. Also four frequencies, namely 0.2 Hz (black), 0.4 Hz (green), 0.6 Hz (blue), and 0.8 Hz (red) are shown. Similar to the zero baseline data hysteresis output, the width of the hysteresis pattern is also increasing with higher frequency. However, because the input data will not move back to the initial position, the major loops and the minor loops of each frequency also do not move back to the origin point. Moreover, the minor loops are gradually converged to the center of the hysteresis pattern. Also, the minor loops show no deadband because the catheter moves from a non-initial position, where no deadband is apparent.

To sum up, the hysteresis pattern of both zero and non-zero data is asymmetric as well as deadband-exhibiting. The range of the deadband is approximate from 0 to 1 bar. It is not affected by the input data frequency. Also, the width of the hysteresis loop depends on the input frequency. Based on this observed behaviour, further modeling and compensation methods will be introduced and investigated in the following chapters.

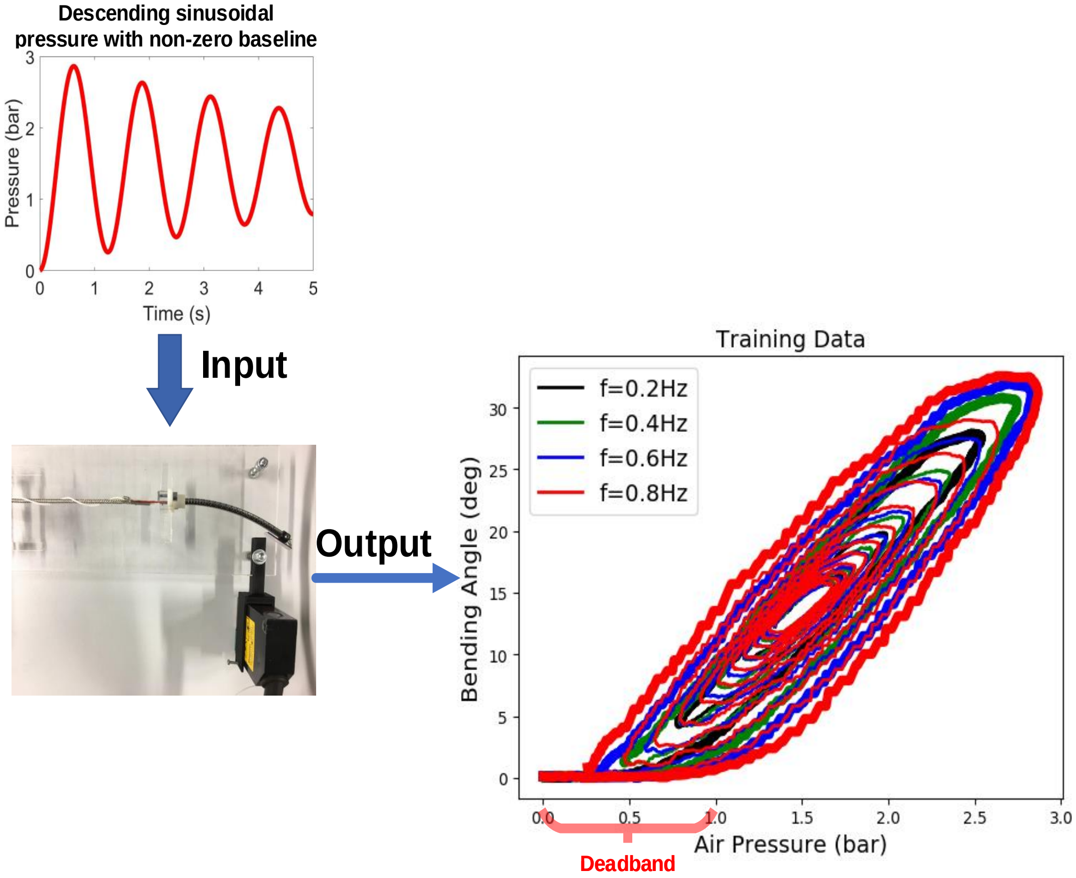

Figure 4.4: Asymmetric deadband hysteresis pattern (non-zero data)

5 Deadband Rate-Dependent Prandtl-Ishlinskii model

5.1 Overview of the DRDPI model

Throughout the literature review in section 2.2, the analytical model Prandtl-Ishlinskii (PI) is chosen. However, there are four different sub-types, namely: rate-independent Prandtl-Ishlinskii, generalized rate-independent Prandtl-Ishlinskii, rate-dependent Prandtl-Ishlinskii, and a generalized rate-dependent Prandtl-Ishlinskii model. Based on the characterization of hysteresis pattern in chapter 4, the hysteresis pattern of the steerable catheter was found to be rate-dependent with deadband.

In order to model this type hysteresis, a state-of-the-art analytic model called a Deadband Rate-Dependent Prandtl-Ishlinskii (DRDPI) model also used in [50] was established in this thesis. The DRDPI model is a sophisticated and practical model that takes into account the impact of input frequency on the pattern of the hysteresis. Moreover, deadband operators in this model allow it to model asymmetric hysteresis as well as deadband-exhibiting behaviour that appears at the bottom of the hysteresis loops (see Figure 4.3 and Figure 4.4). Therefore, the DRDPI model can in principle fully model the hysteresis originating from the entire PAM-driven catheter system. The output of the DRDPI model is as follows [52]:

$$y(t) = a_0 u(t) + \sum_{i=1}^{N} a_i \Phi_{r_i}(t) \tag{5.1}$$

where u is the input data and N indicates the number of play operators of the DRDPI model. a_0 and a_i are weighting constants of the input and of the play operators, respectively. These weighting parameters are identified by using optimization algorithms based on the measured data from the experimental setup. The play operators of the rate-dependent model are constructed using the input u and the dynamic threshold function r_i as follows [53]:

$$\Phi_{r_i} = max\left\{u(t_j) - r_i(\dot{u}(t_j)), min\left\{u(t_j) + r_i(\dot{u}(t_j)), \Phi_{r_i}(t_{j-1})\right\}\right\} \tag{5.2}$$

The dynamic threshold function $r_i(\dot{u}(t_j))$ in (5.2) must be strictly ascending to ensure it has an inverse model [43]. Therefore, a constraint is imposed on the dynamic threshold functions which is shown as follows [54]:

$$0 \leq r_1(\dot{u}(t)) \leq r_2(\dot{u}(t)) \leq \cdots \leq r_n(\dot{u}(t)) \tag{5.3}$$

Figure 5.1 illustrates the effect of the dynamic threshold function r_i on the output. With this dynamic threshold, a so-called Rate-Dependent Prandtl-Ishlinskii (RDPI) is obtained. The output of the RDPI model y is computed based on a linear summation of the system input and the effect of the dynamic threshold function r_i on the input. Figure 5.1 also demonstrates that the output changes when the input rate changes.

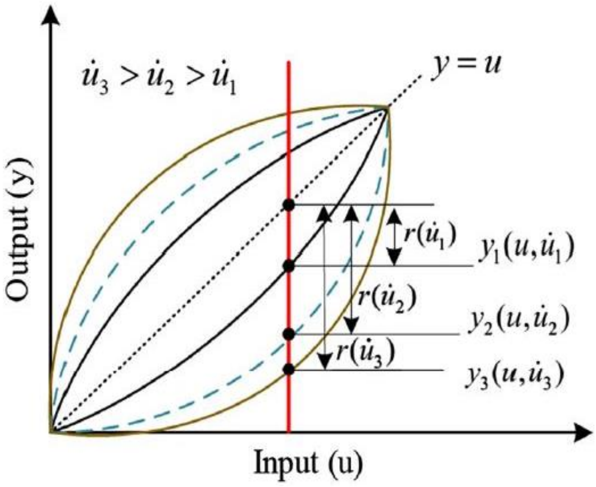

Figure 5.1: The effect of dynamic threshold function on the output [55]

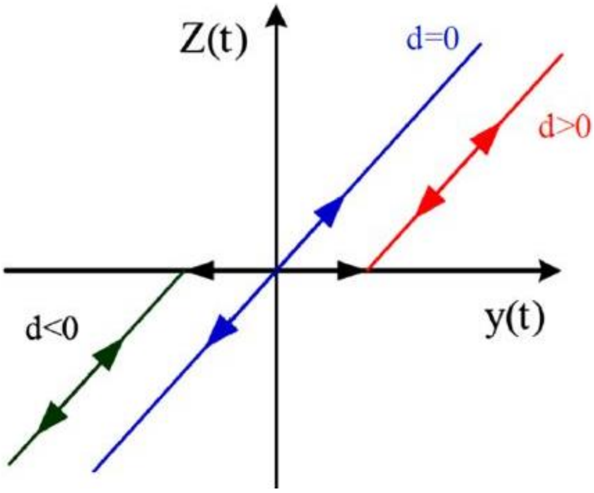

Figure 5.2: Threshold of deadband operators [55]

As summarized in chapter 4, the hysteresis pattern has deadband, which leads to this asymmetric character. The proposed RDPI model is not able to model this hysteresis pattern. Thus the deadband operator is subsequently introduced in the RDPI model. The deadband operator Z can be used to characterize asymmetry and deadband in hysteresis loops. The output of the proposed model is presented as [40]:

$$E(t) = Z(y(t)) = \sum_{j=-k}^{k} g_i \Lambda_{d_i}(y(t)) \tag{5.4}$$

Z is a summation of $2k + 1$ weighted deadband operators, g_i are weighting constants, and k is a positive integer. With deadband operators added, the RDPI model now becomes a Deadband Rate-Dependent Prandtl-Ishlinskii (DRDPI). The output of the Λ_{d_i} operator is calculated by [56]:

$$\Lambda_{d_i}(y) = \begin{cases} max(y - d_i, 0) & for \quad d_i > 0 \\ y & for \quad d_i = 0 \\ min(y - d_i, 0) & for \quad d_i < 0 \end{cases} \tag{5.5}$$

d_i are the thresholds of the deadband operators. The deadband operator is illustrated in Figure 5.2. Deadband operators change the symmetric output of the RDPI model to an asymmetric, which can phase the correspondence between the output of the DRDPI model and the measured data from the experimental setup.

5.2 Threshold and deadband functions of the DRDPI model

In section 5.1, important formulations of the DRDPI are introduced. However, some details of the function are still not clear. Thus in this section, the threshold and deadband functions of the DRDPI model will be further introduced. Because the hysteresis pattern depends on the input frequency, the proposed threshold function is formulated as follows [57]:

$$r_i(\dot{u}(t)) = \zeta i + \beta \left| u(\dot{t}) \right| \tag{5.6}$$

ζ and β are positive constants. The (5.6) ensures the strictly ascending of the dynamic threshold [52]. Also, i shows the number of play operators of the DRDPI model (i = 1, ... , N). In (5.1). ζ describes the rate-independent hysteresis effect and β considers the effect of the input data [52]. The weighting constants of the RDPI model in (5.1) are formulated as follows:

$$a_i = \alpha_0 e^{-\beta_0 i}, \quad for \quad i = 1, 2, ...N \tag{5.7}$$

By using such a relation in (5.7) for describing the weighting constants of the play operators, the number of parameters which need to be defined can be reduced from N to two. Despite the number of parameters is reduced remarkably, from the experiment results it shows that the performance of the model where the function (5.7) is used is not much different with the model where all parameters are used [50].

Another important element of the DRDPI model - the deadband operator - characterizes asymmetry and deadband effects in the hysteresis behavior of the PAM - driven actuator. In order to reduce the number of parameters, parameterized functions that are used to describe the thresholds and weighting constants of deadband operators also can be introduced. For the positive indices the following relations are adapted [50]:

$$d_i = d_p i, \quad for \quad i = 0, 1, ...k \tag{5.8}$$

Table 5.1: Unknown parameters of the proposed model

Model	Unknown parameters
Proposed DRDPI	$\alpha_0, \alpha_1, \alpha_2, \beta_0, \beta_1, \beta_2, d_p, d_n, \zeta, \beta, a_0, g_0$

and

$$g_i = -\alpha_1 e^{-\beta_1 i}, \quad for \quad i = 0, 1..., k \tag{5.9}$$

Besides the positive indices, thresholds and weighting functions for negative indices are considered as follows:

$$d_i = d_n i, \quad for \quad i = -k, -k+1, ... -1 \tag{5.10}$$

and

$$g_i = -\alpha_2 e^{-\beta_2 i}, \quad for \quad i = -k, -k+1, ... -1 \tag{5.11}$$

The positive and negative indices form the output of the deadband relation and also the threshold relation of the DRDPI (see Figure 5.2). In (5.7), (5.9), and (5.11), all of them use an exponential function as a distribution function of the weighting constants to provide a good agreement with the capability of the deadband operators and reduce the number of unknown parameters [52].

5.3 Hysteresis modeling with the DRDPI model

5.3.1 Unknown parameter identification

In section 5.1 and 5.2, formulations of the proposed DRDPI model are introduced. However, there are some unknown parameters that can not be defined. To sum up, all unknown parameters of the proposed DRDPI model are listed in Table. 5.1.

In order to identify the unknown parameters in the proposed model, the objective function is given as follows:

$$f(X) = \sum_{s=1}^{S} \sum_{i=1}^{N_s} \left(\frac{y^s(i) - y^s_{exp}(i)}{N_s} \right)^2 \tag{5.12}$$

In this function y^s is the output of the model and the y^s_{exp} is the measured output. S is the number of considered excitation frequencies, and N_s is the number of measured data samples of the *sth* excitation frequency.

The proposed DRDPI model is trained to model the hysteresis behaviour under different excitation frequencies. The training process will be introduced in the next section. The training process is based on (5.12), which represents the Mean Square Error (MSE) between the model output and the measured data. In order to minimize he MSE, a Genetic Algorithm (GA) is used to identify the unknown parameters in Table 5.1.

A Genetic Algorithm is a search heuristic that is inspired by Charles Darwin's theory of natural evolution. The GA reflects the process of natural selection where the fittest individuals are selected for reproduction in order to produce offspring of a next generation. By using this characterisation of the GA, unknown parameters of the proposed model can be optimized.

Firstly, the process of the natural selection starts with he selection of the fittest individuals from a population. Secondly, the fittest individuals produce offspring which inherit the characters of the parents and that will be added to the next generation. Based on the nature selection principle, if parents have better fitness, their next generation will be better than parents and have a better chance to survive. This process keeps on iterating, and at the end, a generation with the fittest individuals will be found.

During the GA process, five phases will be considered as follows:

- Initial population

  The GA process begins with a set of individuals which is called the initial population. In this population, each individual is a solution to the DRDPI model. In this thesis, the solution is the set of unknown parameters of the proposed DRDPI model. An individual is characterized by a set of parameters known as genes. Genes are joined into a string to form a chromosome, or in this situation called a solution.

- Fitness function

  The fitness function determines the ability of an individual to fit the environment. Also, the fitness function determines the ability of an individual to compete with other individuals. The fitness function of this thesis is given in (5.12). In order to evaluate the individual, a fitness score is given to each individual. The probability that an individual will be selected for reproduction is based on its fitness score.

- Selection

  After the fitness process, the GA will move to the selection phase. The idea of the selection phase is to select the fittest individuals and let them pass to the next generation. During the selection phase, the fitness score is the most important criteria for selecting. Two pairs of individuals are selected based on their fitness scores. Individuals with higher fitness scores will have more chance to be selected for reproduction.

- Crossover

  Crossover is the most significant phase in a Genetic Algorithm. For each pair of parents to be mated, the crossover point is different, in other words, the crossover point is chosen randomly.

- Mutation

  In some new formed generations, some of their genes can be subjected to a mutation with a low random probability. The mutation can lead to flip some genes. The mutation is important to the population because it can maintain diversity and prevent premature convergence.

The GA is implemented by using the Toolbox in MATLAB®. The process of the GA is illustrated in the Figure 5.3.

5.3.2 Training of the proposed DRDPI model

Hysteresis depends on both the input frequency and the amplitude of the excitation signal [50], different excitation frequencies are included in the training data. In this experiment, four frequencies of the excitation signal are set to 0.2 Hz, 0.4 Hz, 0.6 Hz, and 0.8 Hz in the training data set. The input data is generated based on (4.1) and (4.2). The time constant τ was chosen as $\tau = 0.15$ to generate multi-loop hysteresis. Two types of training data (bending angle - input pressure), namely zero and non-zero baseline data, featuring major and minor loops. The training data is shown in Figure 5.4.

In total 26798 samples were acquired in the training data set. It is noteworthy that the hysteresis does not only come from the PAM, but also originates from the whole system e.g., the relative movement between the steer cable and the NiTi tube during bending, the compliance of the pneumatic tubes, the compressibility of the air, the nonlinear behaviour of the applied material (Nitinol) and of the valves that were used.

Based on the GA Toolbox in MATLAB®, the identification process was performed on a CPU (Intel Core i7 CPU @ 2.80GHz with a RAM of 8GB), since there was no wide-spread library for a GPU-based training GA method in MATLAB®. Because of the large amount of training data, the whole identification process of the proposed DRDPI model took around 3.5 hours. The termination condition in the Toolbox was set at the instant where the average relative change in the best fitness function value was less than or equal to $1e-6$, or when the maximum iteration set by users of the GA was reached. In this thesis's training process, the training terminates when the average relative change in the best fitness function value was less than $1e-6$.

In order to visualize the training process, the evaluation of the training loss is shown in Figure 5.5. The x-axis represents the number of executed iterations of the training process and the y-axis represents the training loss. The unit of the training loss is deg^2, which is in line with the fitness function introduced in (5.12) - Mean Square Error. From the x-axis it can be concluded that more than 1000 iterations are performed during the training process. Moreover, a plateau can be observed, which means that this training process is sufficient.

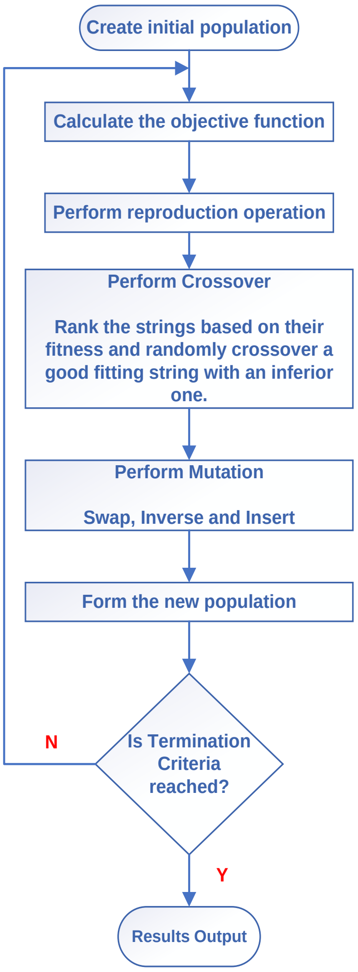

Figure 5.3: Genetic Algorithm process

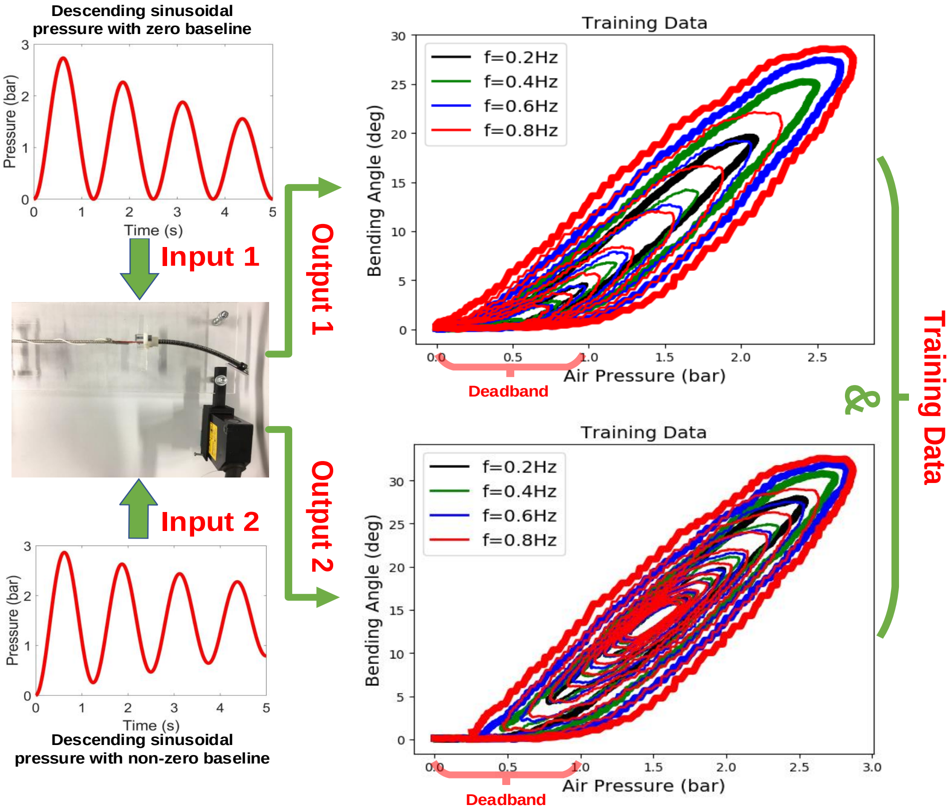

Figure 5.4: Training data collection

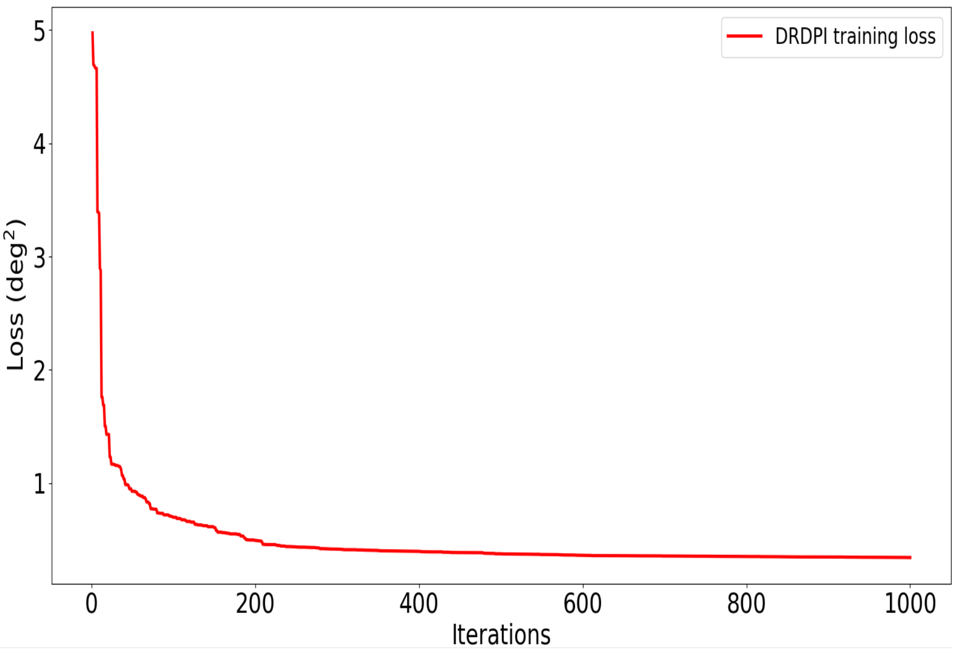

Figure 5.5: DRDPI training loss

Table 5.2: Identified parameters of the model

Parameter	α_0	α_1	α_2	β_0	β_1	β_2	d_p	d_n	ζ	β	a_0	g_0
Value	8.449	0.99114	1.3158	2.3639	1.645	1.8206	0.94854	0.24838	0.05143	0.15616	0.79797	5.9868

5.4 Modeling performance of the DRDPI model

5.4.1 Definition of metrics

In this section, the modeling performance of the proposed DRDPI model is identified. The training process of this DRDPI model with different initial values is performed five times and the best result is chosen. The unknown parameters are shown in Table. 5.2. After identification, the model should be tested with other data sets that are different from the training data sets. The performance of the model will be quantitatively evaluated with the help of three metrics: the Maximum Absolute Error (MAE), the Root Mean Square Error (RMSE), and the Normalized Root Mean Square Error (NRMSE). The MAE, computed following (5.13), measures the maximum absolute error between the output of the model and the ground truth among all samples:

$$MAE = \max\left\{\left|\hat{\theta}_i - \theta_i\right|\right\}, i = 1, 2, ..., N \tag{5.13}$$

where N is the number of sample points in each group of the test data. The RMSE following (5.14) calculates the square root of the square error between the output of the model and the ground truth:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(\hat{\theta}_i - \theta_i)^2}{N}}, i = 1, 2, ..., N \tag{5.14}$$

At last, the NRMSE will also be used to evaluate the performance of the model. The NRMSE relates the RMSE to the observed range of the variables, and it is defined as follows:

$$NRMSE = \frac{RMSE}{\theta_{max} - \theta_{min}} \tag{5.15}$$

5.4.2 Performance evaluation of the DRDPI model

The test data should be different from the training data of the DRDPI model for evaluation. The less the test data resembles the training data, the more the potential of the model is demonstrated. Thus four different types of test signals are described in detail as follows:

1. A descending sine wave following (4.1) but with different f = 0.7 Hz and τ = 0.12 from the training data was generated, while the amplitude A = 1.5 remains unchanged.

   The DRDPI model shows a good performance at the first four waves peak, however, since the fifth peak, the DRDPI model shows larger errors (see Figure 5.6). Because each peak of the model output is leveled off, the real behaviour is not followed very well

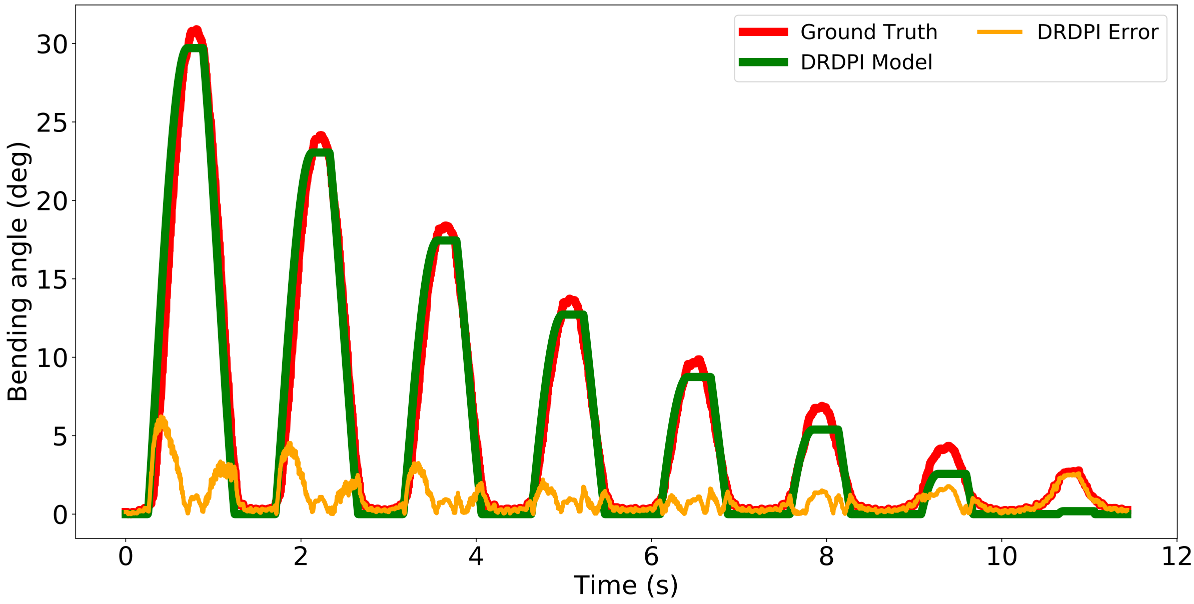

Figure 5.6: 1. Descending sine wave differing from training data (DRDPI)

there. Moreover, at the last wave the model performs the worst since it almost does not predicts an output while the peak of the ground truth is still around 5 degrees. The last wave causes a larger MAE equal to 6.2 degree.

2. In the training data sets, data with non-zero baseline was also used. Thus the DRDPI model could have the ability to test the data with non-zero baseline. To test the performance of the proposed model, a test signal called attenuated down-chirp sine wave with time-varying frequency following (5.16) with A = 0.9, B = 1.2, f = 0.7, τ = 0.1, and c = -0.1 was given by:

$$p(t) = Ae^{-\tau t}(\sin(2\pi(f+ct)t - \frac{\pi}{2}) + 1) + B \qquad [\text{bar}] \tag{5.16}$$

Figure 5.7 shows the prediction made by the DRDPI. At the first two waves, the model performs relative good. However, with decreasing frequency, the model starts to perform worse. The peak and the off-peak of the DRDPI's output can not fit the ground truth. Moreover, the increasing and the decreasing part of the output has a offset compared with the ground truth, which leads to a higher RMSE = 1.22 degree and NRMSE = 6.28%.

3. Since the ascending pattern is not considered in the training data, as third test data an ascending up-chirp sine wave was generated following (5.17) with A = 0.6, f = 0.3, τ = 0.05 and c = 0.15:

$$p(t) = Ae^{-\tau t}(\sin(2\pi(f+ct)t - \frac{\pi}{2}) + 1) \qquad [\text{bar}] \tag{5.17}$$

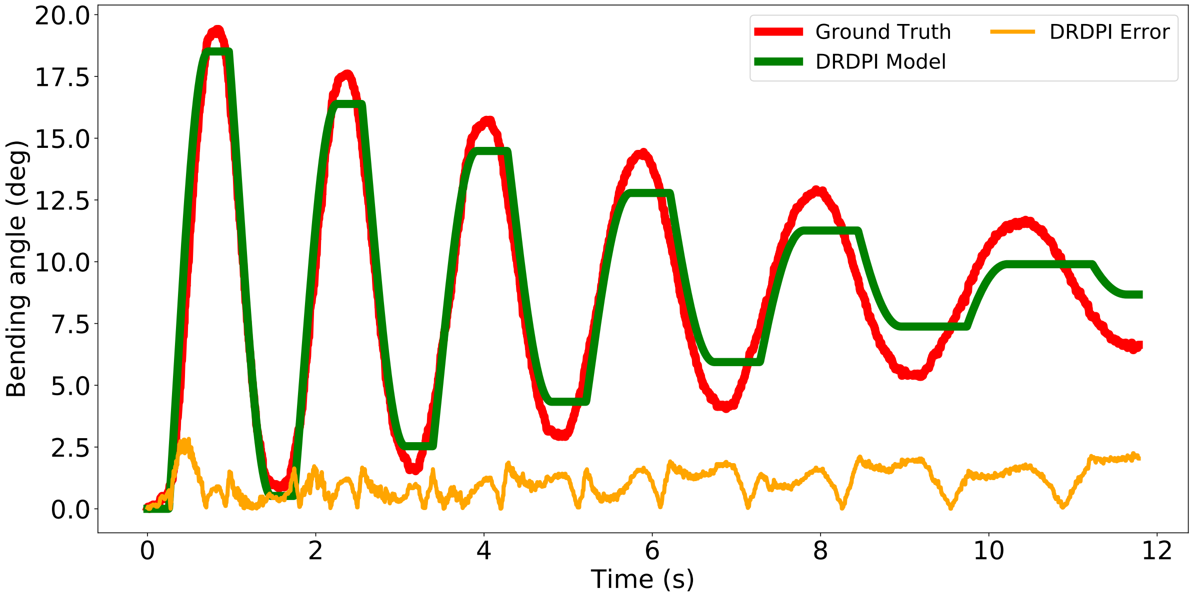

Figure 5.7: 2. Attenuated down-chirp sine wave with shifted baseline (DRDPI)

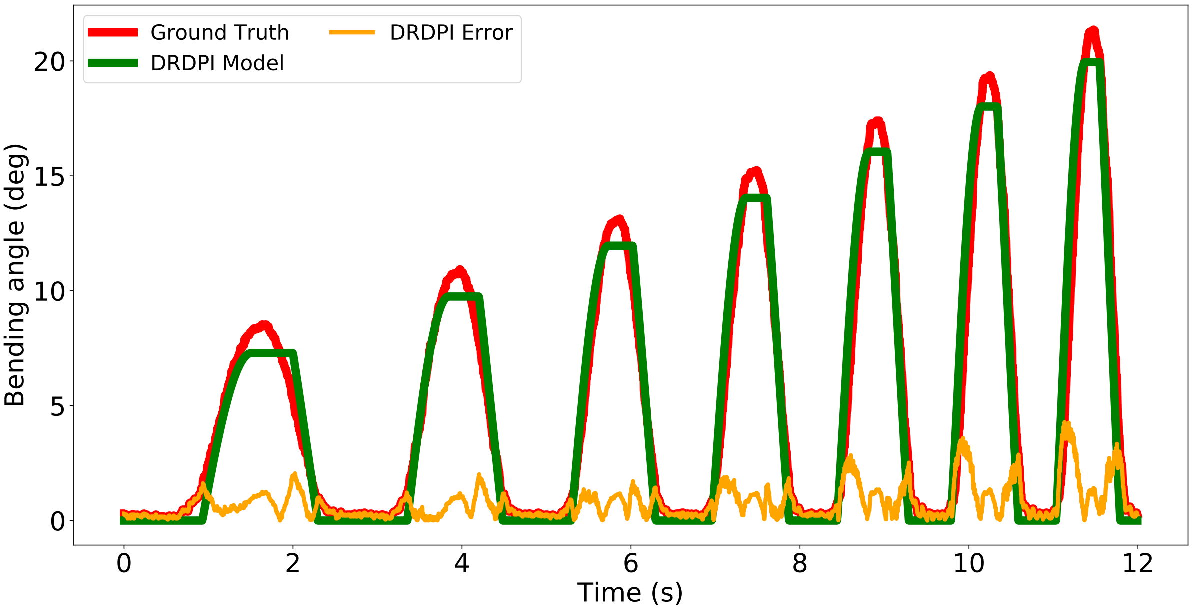

Figure 5.8: 3. Ascending up-chirp sine wave with zero baseline (DRDPI)

Based on Figure 5.8 it can be concluded that the proposed DRDPI model performs better at the lower frequencies. However, during the whole process, the output of the proposed output always has an offset compared to the ground truth. It is also noteworthy that the peak of the output is always a plateau, which could lead to a larger MAE = 4.3 degree in this type of the test data.

4. In order to explore the potential of the proposed model, an arbitrarily varying signal was generated to represent the most general commands that can take place in practice.

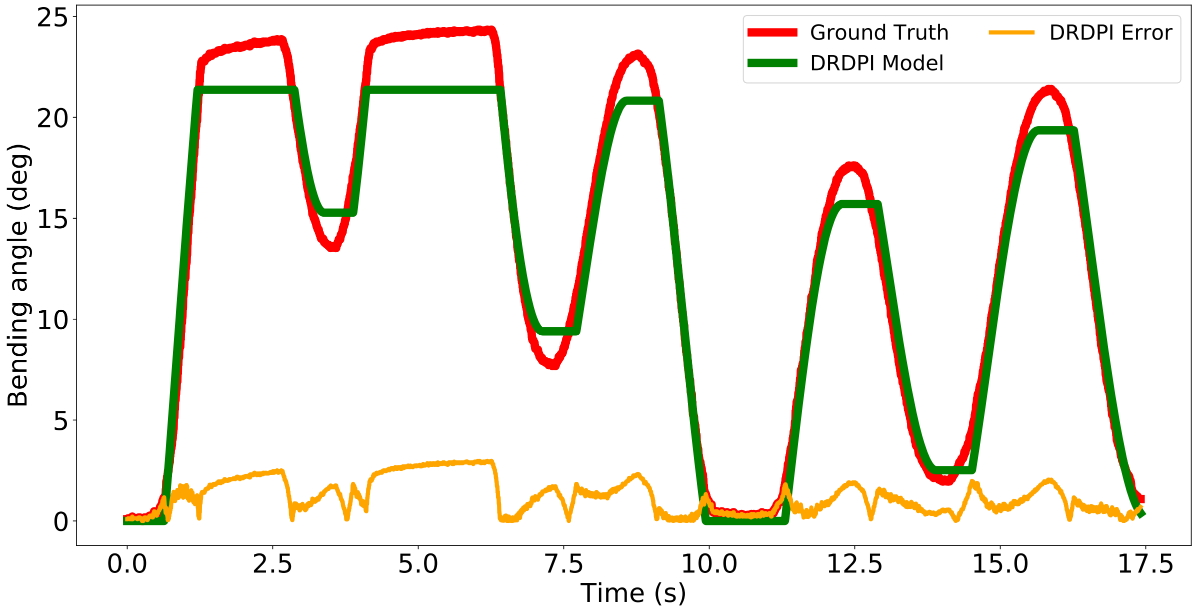

Figure 5.9: 4. Arbitrarily varying signal with plateaus (DRDPI)

Figure 5.9 demonstrates the output of the proposed model compared with the ground truth. During the increasing and decreasing period, the output fits the ground truth relative well. However, the peak and the valley of the DRDPI output are still plateau, thus the MAE = 3.0 degree and the NRMSE = 6.1%.

5.4.3 Discussion of the DRDPI model performance

The training data is based on (4.1) and (4.2), which provides descending sine wave with zero baseline and descending sine wave with non-zero baseline. The four test data provide insight in this generalization ability of the proposed DRDPI model. Except for the first test data with descending zero baseline, the other three test data has different characters from the training data.

In order to evaluate the modeling performance of the proposed DRDPI model, three metrics are used and visualized in the following histogram Figure 5.10. Among these four test signals, the first test signal has the biggest MAE with 6.2 degree. Follow that is the third test signal with 4.3 degree at MAE, and the last two test signals with almost same MAE. Besides the third test signal - Ascending up-chirp sine wave with zero baseline has the smallest RMSE with 1.1 degree, the other three signals have approximate equal sizes. The performance of the proposed model based on the last metric is relatively good with the highest NRMSE 6.28% and with the lowest 4.54%.

To sum up, the proposed Deadband Rate-Dependent Prandtl-Ishlinskii shows a good performance since it can basically follow the test signals. Even if those characters differ from the training data. This result shows that this model has a great potential for modeling the hysteresis in the PAM-driven robotic catheter.

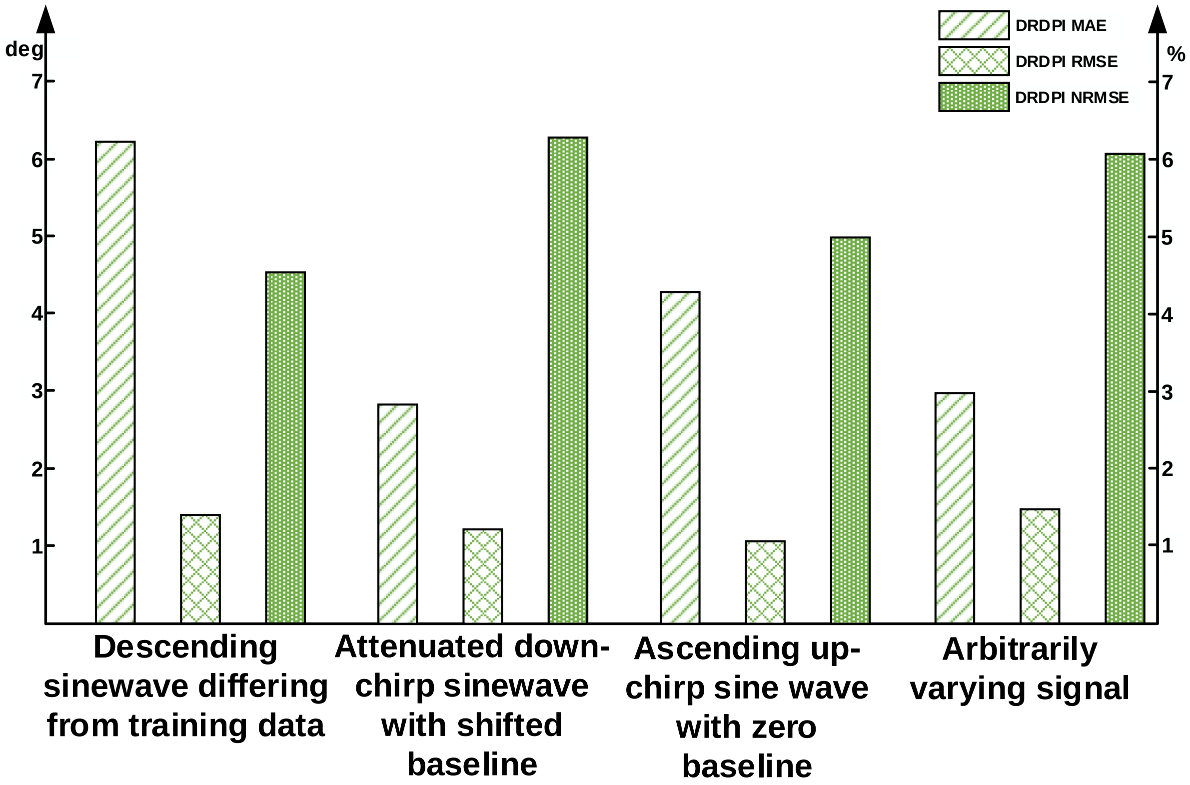

Figure 5.10: Three metrics evaluation of the modeling performance (DRDPI)

6 Support Vector Regression model

6.1 Overview of the SVR model

Support Vector Regression (SVR) is a model based on the statistical learning theory and a supervised learning model. SVR is superior in solving the problems of nonlinearity, which could be used to model the hysteresis [58]. Before moving further step into the Support Vector Regression (SVR) model, a Support Vector Machine (SVM) is briefly introduced. Generally, Support Vector Machines are considered to be a classification approach. They can easily handle multiple continuous and categorical variables. An SVM constructs a hyperplane in multidimensional space to separate different classes. The core idea of an SVM is to find a maximum marginal hyperplane that best divides the data set into classes. Normally, SVM is considered to be best fit for classification problem (classifier). In the following Figure 6.1 three key factors of the SVM are shown:

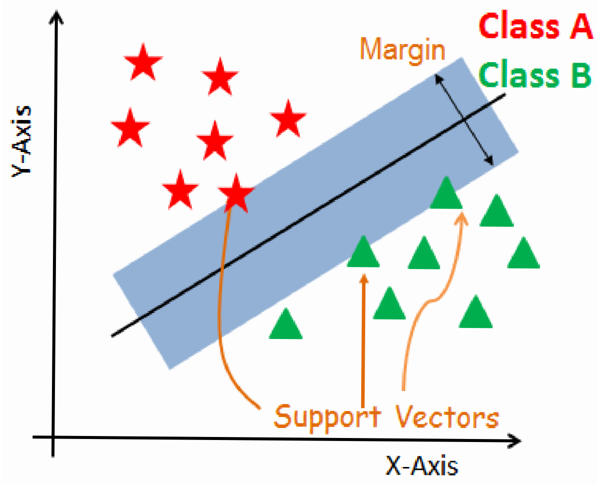

Figure 6.1: Support Vector Machine in 2D

- Support Vectors

  The Support vectors are the data points that are closest to the hyperplane. Figure 6.1 shows the support vectors with red arrow line. These points will define the separating line better by manually defined margins. These points are more relevant to the construction of the classifier.

- Hyperplane

The blue triangle in the figure is the projection of the hyperplane that we want search. A hyperplane is a decision plane which separates between a set of objects having different class memberships.

- Margin

  The width of the hyperplane is called the margin in the 2D using case. In other words, a margin is a gap between the two boundaries on the closest class points. This is calculated as the perpendicular distance from the line to support vectors or closest points. If the margin is larger in between the classes, then it is considered a good margin, a smaller margin is a bad margin.

SVM, the Support Vector Regression uses the same principles as the SVM. But it is still a bit different from SVM. SVR is an regression algorithm , so we can use SVR for working with time series data instead of the classification tasks of SVM.

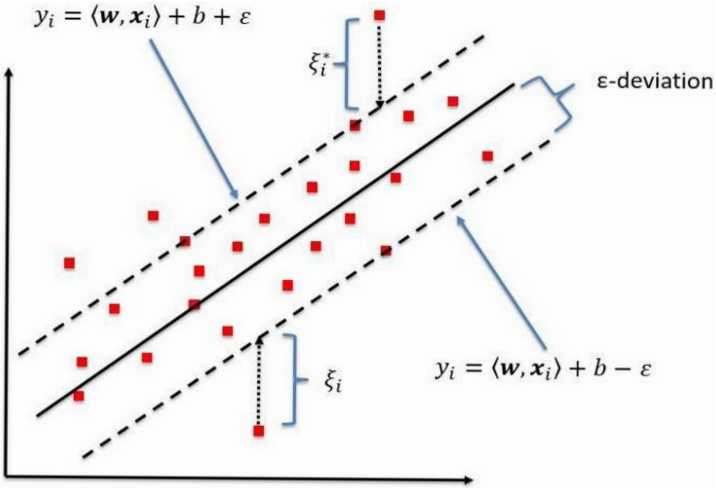

Figure 6.2: Support Vector Regression [59]

Figure 6.2 illustrates the working of a SVR. The main advantage of this algorithm is regression, thus the decision boundary, which is marked with two dash lines, is our margin of tolerance. We are going to take only those points who are within this boundary. To be more specific, SVR gives us more flexibility to define how much error is acceptable in the model and will find an appropriate line or hyperplane in higher dimensions to fit the data. In order to reach this goal, the hyperparameters of the SVR model are important for the model performance. The type of hyperparameters and the selection of them are going to be introduced in the following section.

6.2 Hyperparameters of the SVR model

In order to obtain a better SVR model, the tuning process of the hyperparameters should be applied. In this project, three hyperparameters of the proposed SVR model will be introduced:

Table 6.1: Hyperparameters of the SVR model

C	**Kernel**	**Kernel coefficient**	**Margin of tolerance**
10	**RBF**	**0.1**	**0.1**

- Kernel

  The main function of the kernel is to transform the given data set input data into the required form. In the SVR model, there are various types of function that can be chosen such as a linear, a polynomial, and a Radial Basis Function (RBF).

- Penalty parameter

  C is the penalty parameter, which represents misclassification or an error term. The misclassification or error term tells the SVR optimization how much error is bearable. A smaller value of C creates a small-margin hyperplane and a larger value of C creates a larger-margin hyperplane. The MAE generally decreases as C increases.

- γ

  A lower value of γ will loosely fit the training data set, whereas a higher value of gamma will exactly fit the training data set. Too large γ value such as 10 can cause over-fitting.

Based on the value of the hyperparameters, it is a essential problem that how to choose them in a scientific way. Thus here in the thesis, the K-Fold cross validation method was implemented. K-Fold cross validation is where a given data set is split into a K number of sections/folds where each fold is used as a testing set at some point. To obtain better hyperparameters, here $K = 10$ is adopted, which means the data set is split into 10 folds. In the first iteration, the first fold is used to test the model and the rest are used to train the model. In the second iteration, second fold is used as the testing set while the rest serve as the training set. This process is repeated until each fold of the 10 folds have been used as the testing set. After the cross validation. the hyperparameters of the SVR are chosen as follows: penalty C = 10, kernel = Radial Basis Function (RBF) kernel, kernel coefficient γ = 0.1, margin of tolerance ϵ = 0.1. Parameter C and γ were optimized using a grid search (C$\in$\{0.001,0.01,0.1,1,10\}, $\gamma\in$\{0.001,0.01,0.1,1\}) and a cross-validation method (k=10) as mentioned before.

6.3 Hysteresis modeling with the SVR model

In the previous section, the hyperparameters of the proposed SVR model are already described. In order to compare the proposed model with the DRDPI model proposed in the last chapter, the training data generation of the SVR is also based on the (4.1) and (4.2). The whole training data of the SVR is same with the DRDPI model (see Figure 5.4).

Similar to the training process of the DRDPI model, the training process of the SVR model also has a loss function. The unit of the training loss in DRDPI model is deg^2, thus the loss function of the SVR model is chosen as MSE function. Besides the training loss, in this machine learning-based model, the over-fitting problem should also be noticed. This problem is solved by also plotting the validation loss. Figure 6.3 shows the training loss and the validation loss of the SVR model.

The training loss of the proposed SVR model reaches a plateau only after around 20 iterations. The same holds for the validation loss. Moreover, the validation loss does not go up, which means no over-fitting problem in this training process. Through the local zoom in the Figure 6.3, the training loss and validation loss lines are different, but very similar. This also indicates that the training of the proposed SVR model is more efficient than the proposed DRDPI model. The training of the SVR model is performed on a CPU (Intel Core i7 CPU @ 2.80GHz with a RAM of 8GB). The training time of the SVR model took around 20 minutes.

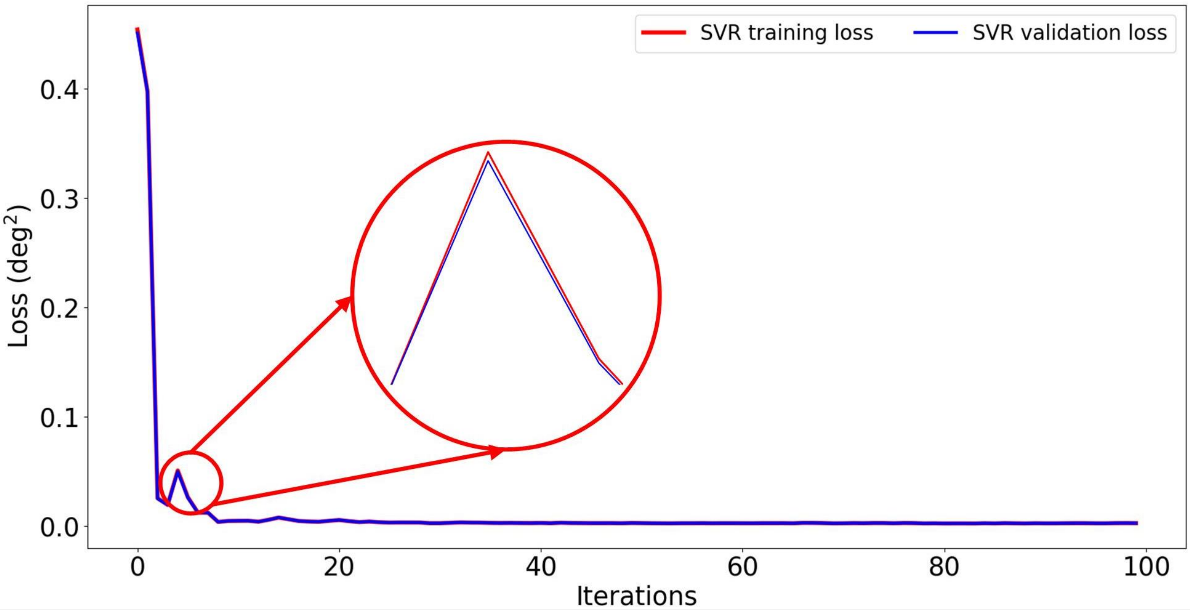

Figure 6.3: SVR training and validation loss

6.4 Modeling performance of the SVR model

6.4.1 Performance evaluation of the SVR model

In order to explore the performance of the SVR model, besides training data and validation data, test data is also needed to generate. The SVR model implemented in this thesis was based on the work of [58]. Since the training data of the SVR model is same as for the DRDPI model, the same test data is also used to test the SVR model. The four test data and its results are illustrated as follows:

1. A descending sine wave that is same with the first test data in section 5.4 is adopted.

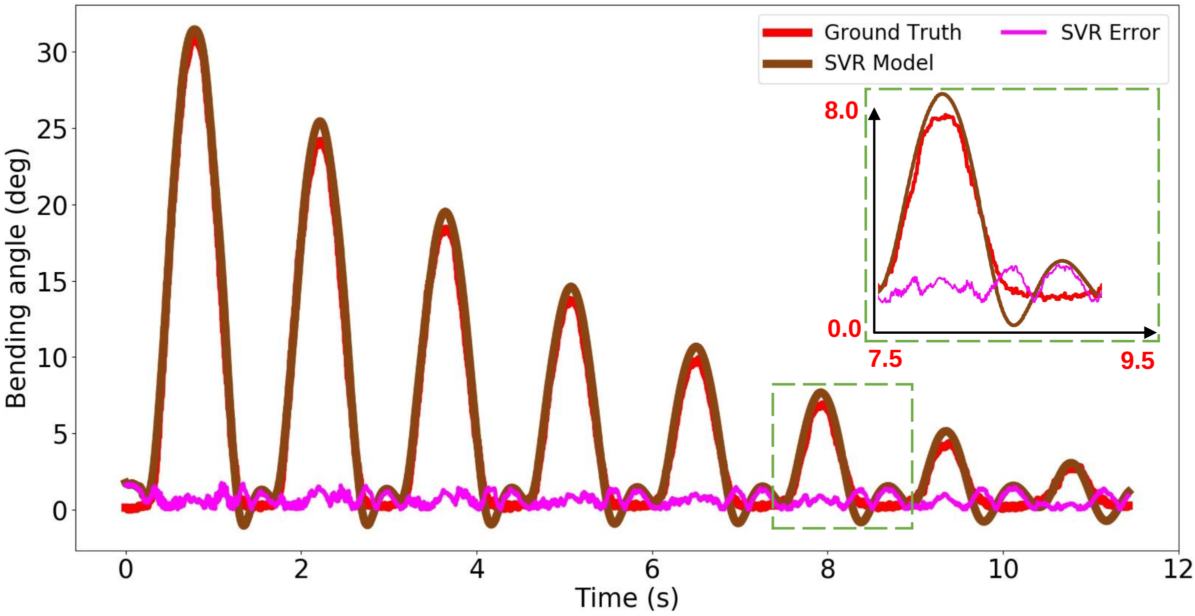

Figure 6.4: 1. Descending sine wave differing from training data (SVR)

Table 6.2: Three metrics of the first test data in SVR model

Modeling Methods	**Descending sine wave differing from training data**		
	RMSE (deg)	**MAE (deg)**	**NRMSE**
SVR	**0.758**	**1.833**	**2.45%**

At each peak of the test data, the proposed SVR model shows a good performance. It's output can basically fit the ground truth, which can lead to a relative lower RMSE = 0.8 degree. However, it is noteworthy that the output of the SVR model shows a disturbance at the beginning of the test data and the deadband of the ground truth. At the start, the output can not depart from zero but starts from around 2-3 degrees. At each deadband, it's performance shows a disturbance wave with small amplitude around 2 degree. Thus the MAE is lower than the 2 degree but equals to 1.8 degree.

2. The same Attenuated down-chirp sine wave with shifted baseline is also implemented here following (5.16) with A = 0.9, B = 1.2, f = 0.7, τ = 0.1, and c = -0.1.

   This test data differs at two points from the training data, namely non-zero baseline

Table 6.3: Three metrics of the second test data in SVR model

Modeling Methods	**Attenuated down-chirp sine wave with shifted baseline**		
	RMSE (deg)	**MAE (deg)**	**NRMSE**
SVR	**0.797**	**3.535**	**4.10%**

Table 6.4: Three metrics of the third test data in SVR model

Modeling Methods	**Ascending up-chirp sine wave with zero baseline**		
	RMSE (deg)	**MAE (deg)**	**NRMSE**
SVR	**0.760**	**1.687**	**3.56%**

and changing frequencies. The same problem from the first test data also appears here. At the beginning, the output is not able to start from zero. This leads to a larger MAE in this test data with a MAE = 3.5 degree. However, the output of the SVR follows the ground truth very well for changing frequencies. This leads to a smaller RMSE = 0.8 degrees.

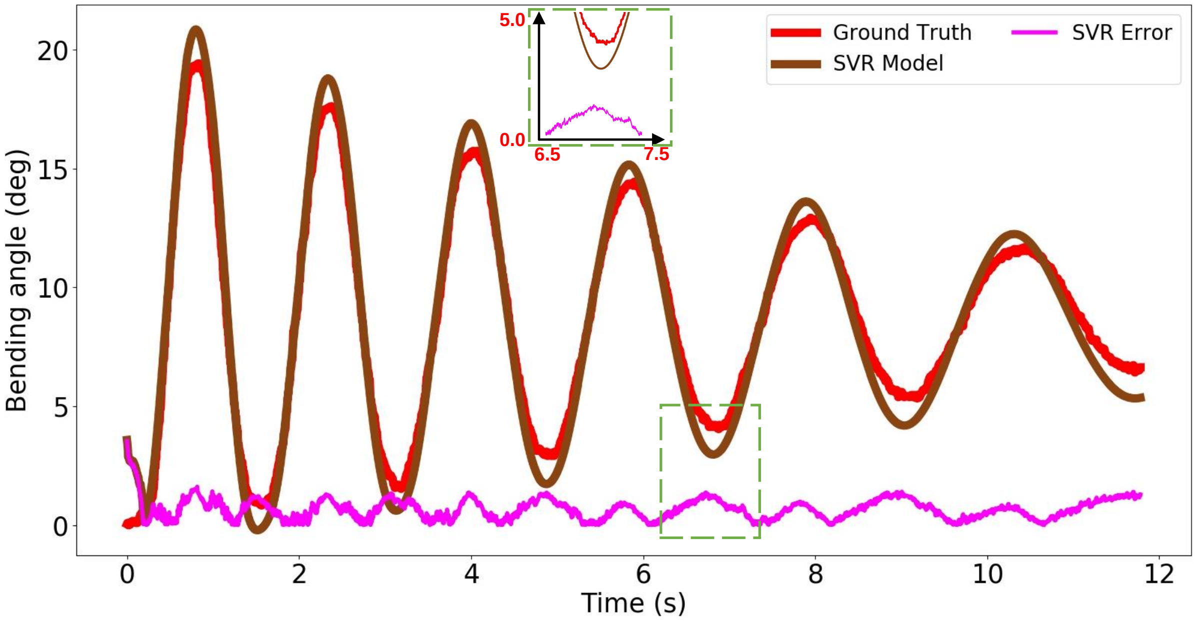

Figure 6.5: 2. Attenuated down-chirp sine wave with shifted baseline (SVR)

3. Same with the test data mentioned in the section 5.4, this test data called Ascending up-chirp sine wave with zero baseline. The generation of this test data following (5.17) with $A = 0.6$, $f = 0.3$, $\tau = 0.05$ and $c = 0.15$.

   Similar with the performance in the first test data, the proposed SVR model is not able to follow the trajectory of the ground truth at the beginning and the deadband parts. However, the resulting error stays consistently below 2 degrees. As visble in Figure 6.6. Thus the MAE equals to 1.7 degrees and the RMSE = 0.8 degrees.

4. In order to explore the potential of the proposed SVR model, an arbitrarily varying signal same with the last chapter was implemented.

   Despite that this type of test data shows plateaus, which is more practicable but

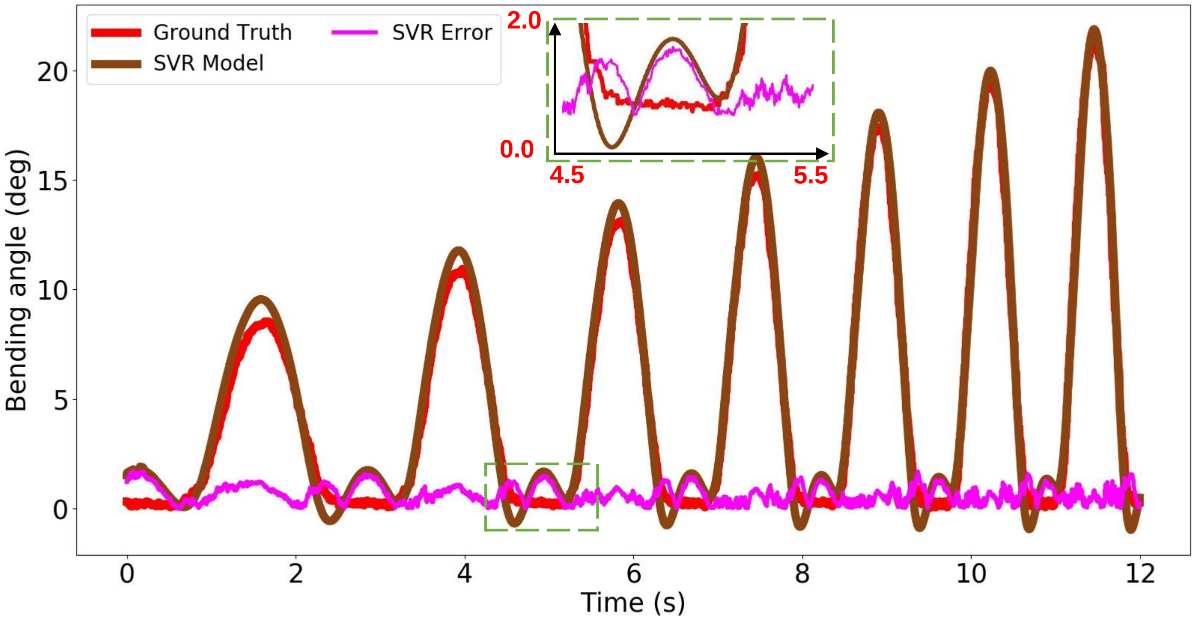

Figure 6.6: 3. Ascending up-chirp sine wave with zero baseline (SVR)

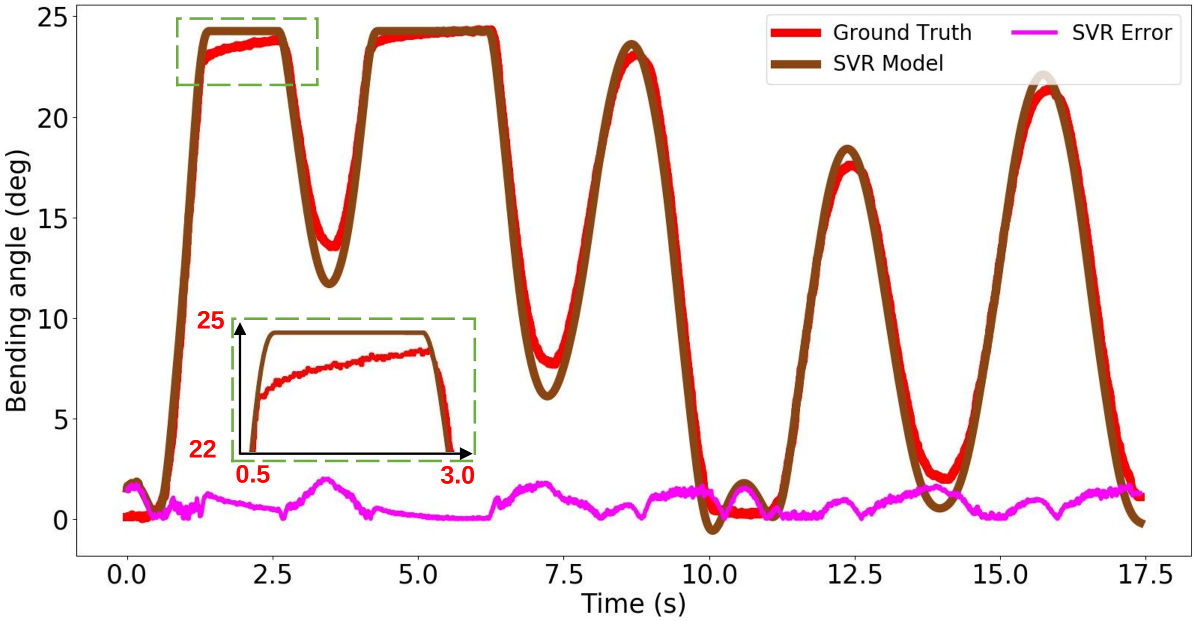

Figure 6.7: 4. Arbitrarily varying signal with plateaus (SVR)

Table 6.5: Three metrics of the fourth test data in SVR model

Modeling Methods	**Arbitrarily varying signal with plateaus**		
	RMSE (deg)	**MAE (deg)**	**NRMSE**
SVR	**0.920**	**2.010**	**3.78%**

challenging for the proposed model. The output of the model can not only follow the plateau trajectories well, but also performs well at each peak. This leads to an RMSE = 0.9 degrees and a NRMSE = 3.78%. Not only the SVR model has the worst performance at the beginning and the deadband, the proposed model but also performs worst at each wave valley. This leads to the MAE = 2.0 degrees.

6.4.2 Discussion of the SVR model performance

Table 6.2 - 6.5 summarize all three metrics of the four test data. From the Figure 6.4 - Figure 6.7 it can be concluded that the proposed SVR can provide higher accuracy, especially during the increasing, decreasing, and the plateau periods. To sum up, the proposed SVR model can model the hysteresis of this robotic PAM-driven catheter with sub-degree accuracy. However, the SVR model is not able to follow the trajectory at the beginning and the deadband of the ground truth. This could lead to lager MAE and NRMSE. Even if the SVR model has this disadvantage, the SVR model still outperforms than the DRDPI model in all three metrics.

7 Long Short-Term Memory model

7.1 Overview of the LSTM model

A Long Short-Term Memory (LSTM) is a deep learning-based methods, which is explicitly designed to avoid the long-term dependency problem. It happens that an LSTM was proposed as a tool to process sequential information and take historical information into account [60]. A system is defined to exhibit hysteresis if it has a sort of memory. This means that the output at a certain moment is not only determined by the corresponding input but also by the past inputs [34]. In a sense this can be considered as a long-term memory. Thus it is not illogical to consider the use of an LSTM to model hysteretic behavior.

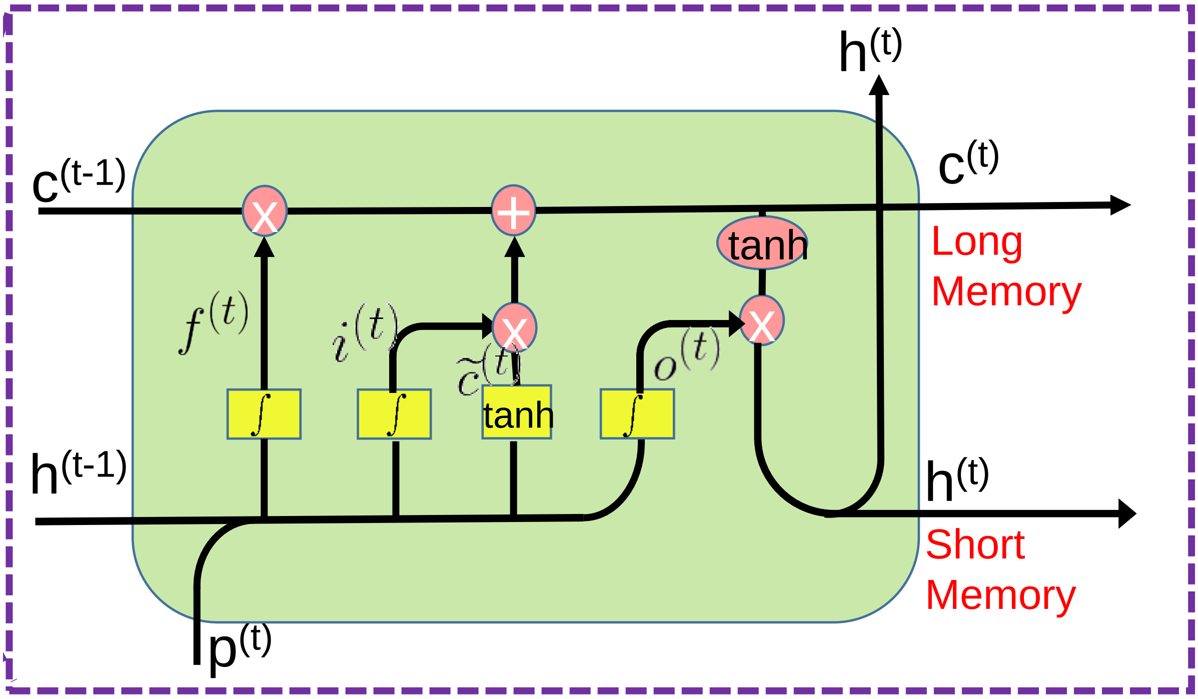

Figure 7.1: Long Short-Term Memory cell

Figure 7.1 shows an LSTM cell. The LSTM can be viewed as a stack of LSTM cells that contain a number of units in a LSTM cell. Each LSTM cell has an input gate, an output gate, and a forget gate to control the information flow. The information in a cell is processed as follow [60]:

$$f^{(t)} = \sigma(W_f p^{(t)} + U_f h^{(t-1)} + b_f) \tag{7.1}$$

$$i^{(t)} = \sigma(W_i p^{(t)} + U_i h^{(t-1)} + b_i) \tag{7.2}$$

$$\widetilde{c}^{(t)} = \tanh(W_g p^{(t)} + U_g h^{(t-1)} + b_g) \tag{7.3}$$

$$c^{(t)} = f^{(t)} * c^{(t-1)} + i^{(t)} * \widetilde{c}^{(t)} \tag{7.4}$$

$$o^{(t)} = \sigma(W_o p^{(t)} + U_o h^{(t-1)} + b_o) \tag{7.5}$$

$$h^{(t)} = \tanh(c^{(t)}) * o^{(t)} \tag{7.6}$$

In (7.1), $p^{(t)}$ is the input pressure to the PAM-driven robotic catheter experimental setup at time t. The variable σ is the sigmoid function with an output range of [0,1] (see Figure 7.2) given by:

$$S(x) = \frac{1}{1+e^{-x}} \tag{7.7}$$

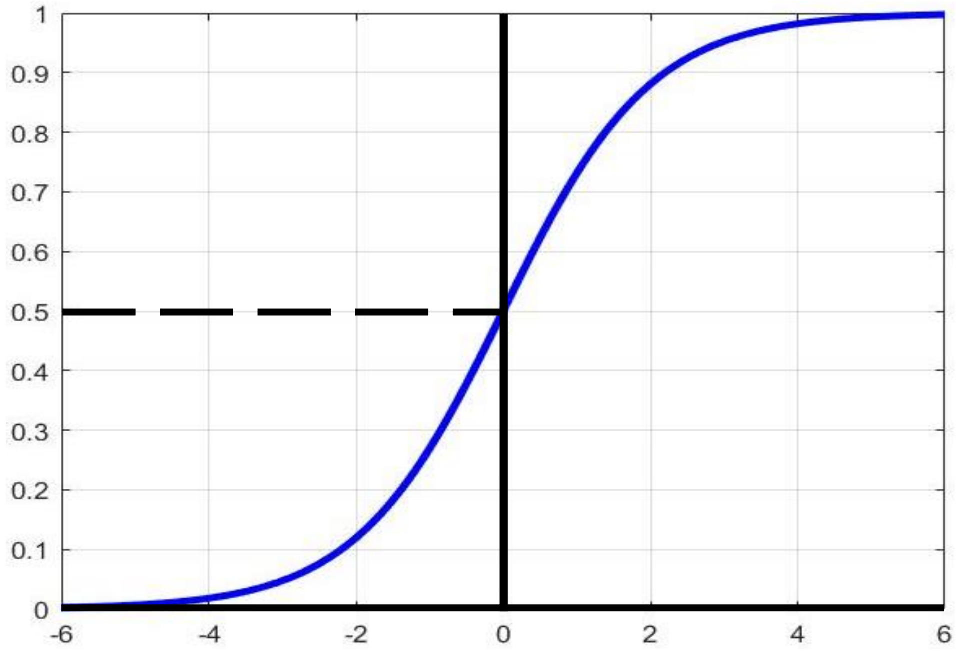

Figure 7.2: Sigmoid function

It regulates the percentage of information that flows through each gate. Another activation function is the tanh (see (7.8)), whose outputs range from [-1,1] (see Figure 7.3):

$$\frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{7.8}$$

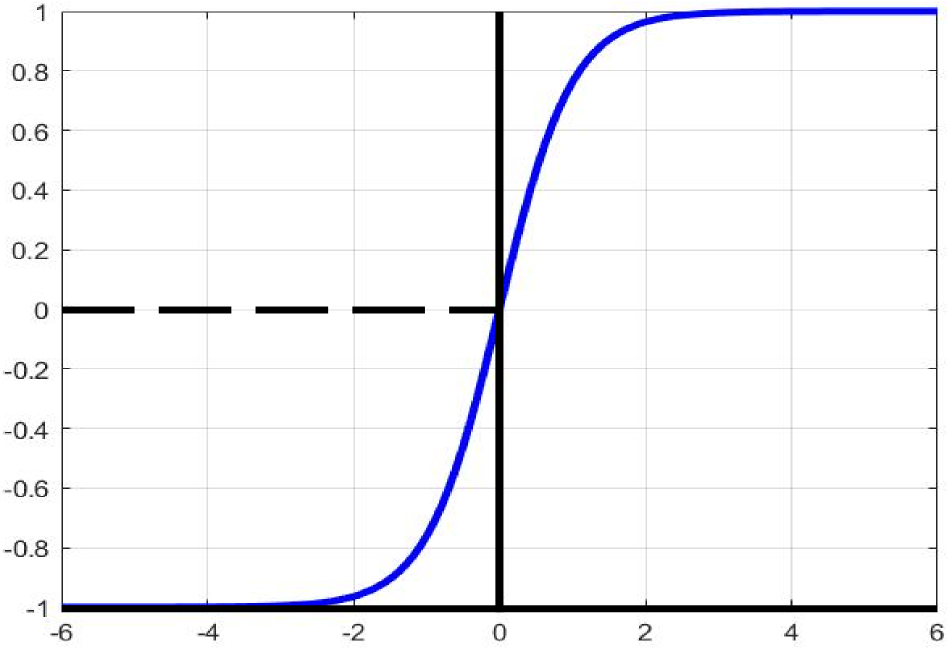

Figure 7.3: Tanh function

The forget gate f determines which information from in the previous cell state $c^{(t-1)}$ can be preserved. The input gate i, which goes through sigmoid function and the input modulation gate $\widetilde{c}$, which goes through tanh function, are combined to update the cell state (see Figure 7.1). A new cell state $c^{(t)}$ is created afterwards. The cell state c stores the selective information, which is considered as long-term memory in the LSTM, can be established. The output gate o works as a filter. It will decide which part of information from the new cell state $c^{(t)}$ can be output and transferred to the hidden state $h^{(t)}$. The matrix, U, W, and b, with different subscripts, represent the weights and biases in each gate. The dependencies will be tracked based on the gates between the elements in the input sequence.

7.2 Construction of the LSTM model

7.2.1 Data preprocessing for the LSTM model

As introduced in the last section, the output at a certain moment is not only determined by the corresponding input but also by the past inputs. Because of the character of the proposed LSTM model, the input of the LSTM model should not be one dimension but multi-dimensional. The historic input should be considered in each input matrix. Figure 7.4 demonstrates the data preprocessing method of the input data.

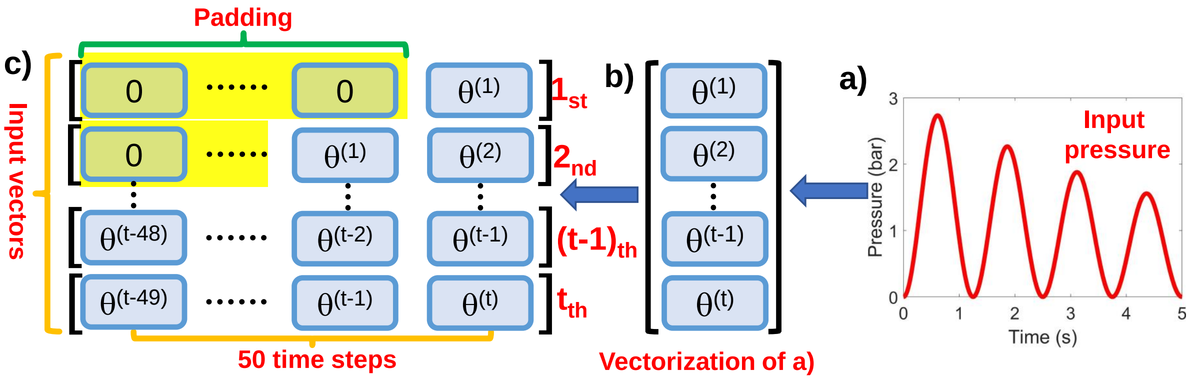

Figure 7.4: Input data preprocessing

a) The input data of the LSTM model is the air pressure along with the time. In other words, the input data is a time series data with many data points. The input data needs to be processed before it is sent to the LSTM model.

b) The whole input data can be viewed as a matrix with only one column but multi rows. As shown in the Figure 7.4, the input data is a vector ranged with time.

c) After vectorization of the input data, the input data will be split into several input vectors based on the size of the time window. The size of the time window determines the historic input pressure that the LSTM model will consider. However, at the initialization, there is no historic input data when the amount of the selected time points is smaller than the size of the time window. Therefore a padding method is proposed to solve this problem. Zero elements are used to fill in the gap during the initialization. According to a previous pilot study [61] and [62], a window size equal to 50 was shown to have a good performance while maintaining a low computational cost. Figure 7.4 (c) shows the padding and splitting process for a time window of 50 samples.

7.2.2 Hyperparameters of the LSTM model

Similat to the proposed SVR model, the LSTM model is also a learning-based model. Therefore the hyperparameters of the LSTM are also essential for the performance of the model. The hyperparameters of the LSTM model are introduced as follows:

- Number of hidden layers

  An LSTM can be viewed as a stack of LSTM cells that contain a number of units in an LSTM cell. Therefore the layers of the LSTM should be defined by the user. In this thesis, 2 hidden layers are chosen for the proposed LSTM model.

- The number of neurons per cell

  As mentioned before, the LSTM contains a number of units in an LSTM cell. The number of neurons need also be defined, and the number of neurons is better with the power of 2. Therefore in this experiment 64 neurons in each LSTM cell are adopted.

- The activation function

  As introduced in the last section, the activation functions of the LSTM are a sigmoid (see Figure 7.2) and a tanh (see Figure 7.3).

- Optimizer

  Optimisers play a very crucial role to increase the accuracy of the model. In the LSTM there are many possibilities for choice of optimizer.

  1. SGD: Stochastic Gradient Descent

     The SGD optimizer is also known as an incremental gradient descent the tries to find minimum or maximum error via iteration. However, it is not certain to converge to a local minimum even if the objective function is not convex or pseudo convex. This is one of the main issues it faces.

  2. Adagrad

     The slope could be adapted and thus speed up the SGD. One of the main benefits of the Adagrad is that it eliminates the need to manually tune the learning rate. However, one of the major disadvantages is the learning rate decreasing, which will lead to no further learning / acquisition of new information.

  3. Adam

     Adam is a tool for calculating the learning rate and the parameter that has been proven to perform well in practice and to compare favorably against other adaptive learning algorithms.

  To sum up, SGD, adagrad are very similar algorithm and since Adam was found to slightly outperform both of them, Adam is generally chosen as the best choice.

- Loss function

  In order to compare with the other two models, the unit of the loss function should be better same. Since the other two model's loss function has a unit of deg^2, the same loss function was chosen here.

- training subset and validation ratio

  In order to avoid the over-fitting problem of the training process, the whole training data should be split into the training subset and validation subset. Normally, the whole training data is divided into 70% for training and 30% for validation.

- Batch size

Table 7.1: Hyperparameters for the LSTM network

	Number of hidden layers	Number of neurons per cell	Activation functions	Optimizer	Loss function	Training-subset /Validation ratio	Batch size	Learning Rate	Epoches
LSTM	2	64, 64	Tanh/Sigmoid	Adam	L2 Loss	70%30%	16	0.001	50

The batch size determines the amount of the input series data. Adjusting the batch size could increase the efficiency of the training process, however, too large batch size could add extra computation cost.

- Learning rate

  Learning rate determines the learning speed of the training process. If the learning rate is too large, it will decrease the learning efficiency and decrease the performance of the model. However, if the learning rate is too small, the learning speed is too slow and this will lead to more training time.

- Epoches

  The number of training iterations is defined by the number of the epoch. The number of the epoch should be appropriate, if the number is too small, the training process may not have converged and may not reach the plateau of the training loss. But with too large epoch number, over-fitting may arise together with an extra waste of the computation.

Table. 7.1 shows all the hyperparameters of this LSTM model. All hyperparameters are validated and chosen based on the previous study [61], [62]. The modeling of the hysteresis by using the proposed LSTM model will be based on these hyperparameters.

7.3 Hysteresis modeling with the LSTM model

After selecting the hyperparameters of the proposed LSTM model, the training process of the LSTM model will be implemented. In order to guarantee the consistency of the comparison between these models, the training data generation is also based on (4.1) and (4.2). The whole training data is the same with the proposed DRDPI and SVR models.

As mentioned before, the loss function of the LSTM model is chosen as the L2 loss. Figure 7.5 demonstrates the training loss of the LSTM model. It is clear that the training loss reaches a plateau after about 30 iterations. In order to avoid the over-fitting problem, the 30% of the whole training data is used as validation data. The validation loss can also be seen in Figure 7.5. The validation loss does not increase, which means that no over-fitting problem occurred.

Figure 7.6 illustrates the structure of the proposed LSTM model as well as all the hyperparameters. At each iteration 50 time steps are input into the LSTM model. First, the input data will go through the 2 stacked layer. Secondly, the output information will go through a fully connected layer. Therefore, the sequential pressure data $p^{(t-49)}$,$p^{(t-48)}$,...$p^{(t)}$ were entered into the LSTM to predict one bending angle $\theta^{(t)}$. Zeros were padded on the left side of the input sequence for predicting the first 49 bending angles. A fully connected layer with input dimensions = 64 and output dimension = 1 was added after the last LSTM cell.

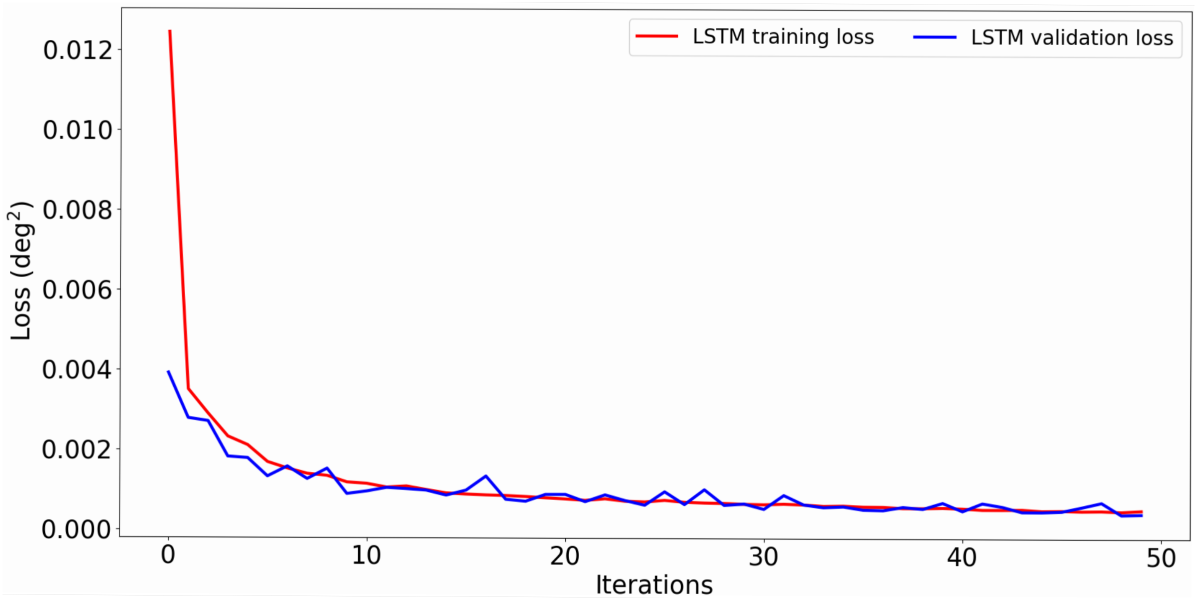

Figure 7.5: Training loss of the LSTM

To increase the learning efficacy, all the training data were normalized between [-1,1] (7.9) to match the range of the activation function in (7.1).

$$y_{norm} = (b-a)\frac{y - y_{min}}{y_{max} - y_{min}} + a, a = -1, b = 1 \tag{7.9}$$

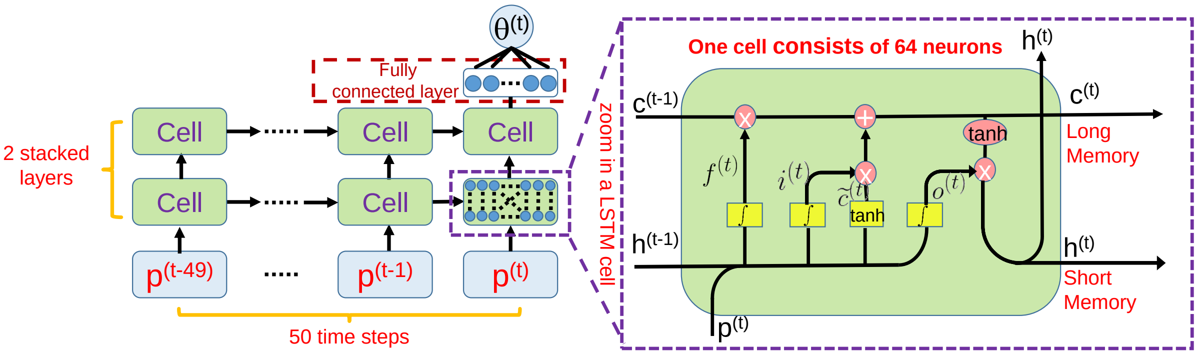

Figure 7.6: LSTM structure

The network was implemented in an open source machine learning framework named Pytorch. The total number of trainable parameters is 50497. The training was performed on an 4 GB NVIDIA CUDA-capable GPU. The LSTM was trained for 50 epochs. With 4 GB GPU, the whole training time was around 10 minutes.

Figure 5.5, Figure 6.3, and 7.5 show the training loss of the proposed DRDPI, SVR, and LSTM model, respectively. The y-axis of the DRDPI model starts from the 5 deg^2 and ends at about 0.2 deg^2 after 1000 iterations. Compare to the SVR model, the training loss starts from 0.5 deg^2 and ends at about 0.02 deg^2 after 100 iterations. However, the LSTM model shows more efficient learning, which starts from the 0.012 deg^2 and ends at the very low level only after 50 iterations.

7.4 Modeling performance of the LSTM model

7.4.1 Performance evaluation of the LSTM model

In order to test the performance of the proposed LSTM after training, four groups of test data are used. Again these data are the same as the test data that was used in the DRDPI and SVR model. The implemented model in this thesis was based on the previous work [51]. The four groups of the test data and their results are introduced as follows:

1. Based on the (4.1) with different f = 0.7 Hz and τ = 0.12 from the training data, the same descending sine wave was generated.

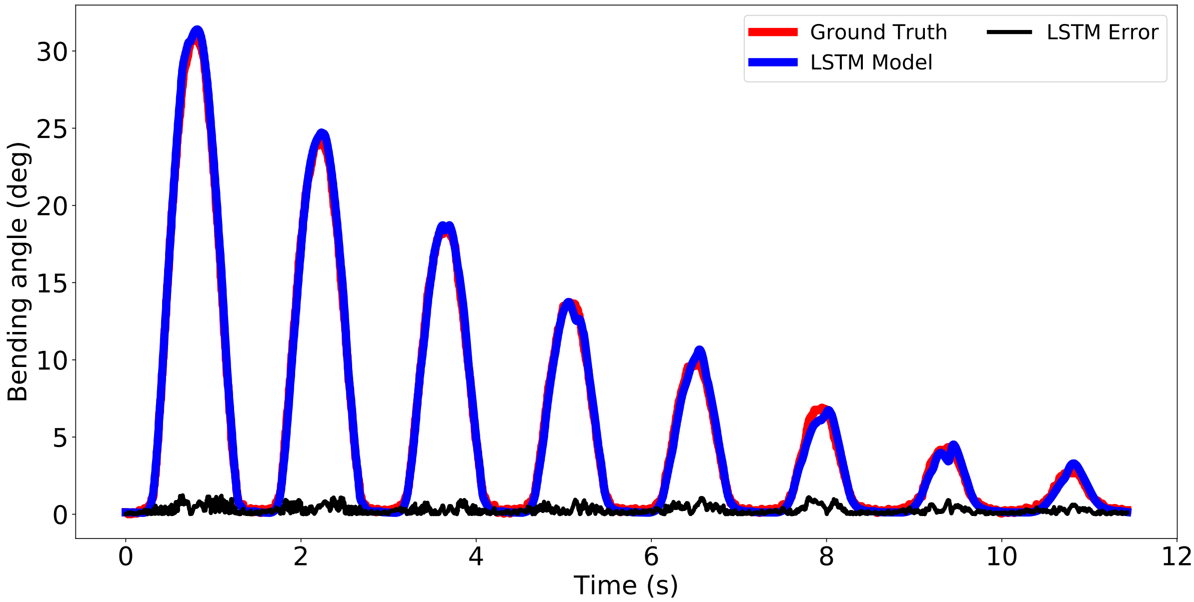

Figure 7.7: 1. Descending sine wave differing from training data (LSTM)

Except for the performance on the second last and third last wave, which shows a little disturbance at the peak of the output, the general performance of the proposed LSTM model is promising. The model output fits the ground truth during the increasing and decreasing parts well. This explains the relative small RMSE = 0.4 degrees. Because of the good fitting of the peak and the valley of the wave, the MAE of the output is also small. The MAE = 1.2 degrees in this type of the test data.

2. Based on (5.16), the same test data with A = 0.9, B = 1.2, f = 0.7, τ = 0.1, and c = -0.1 was generated.

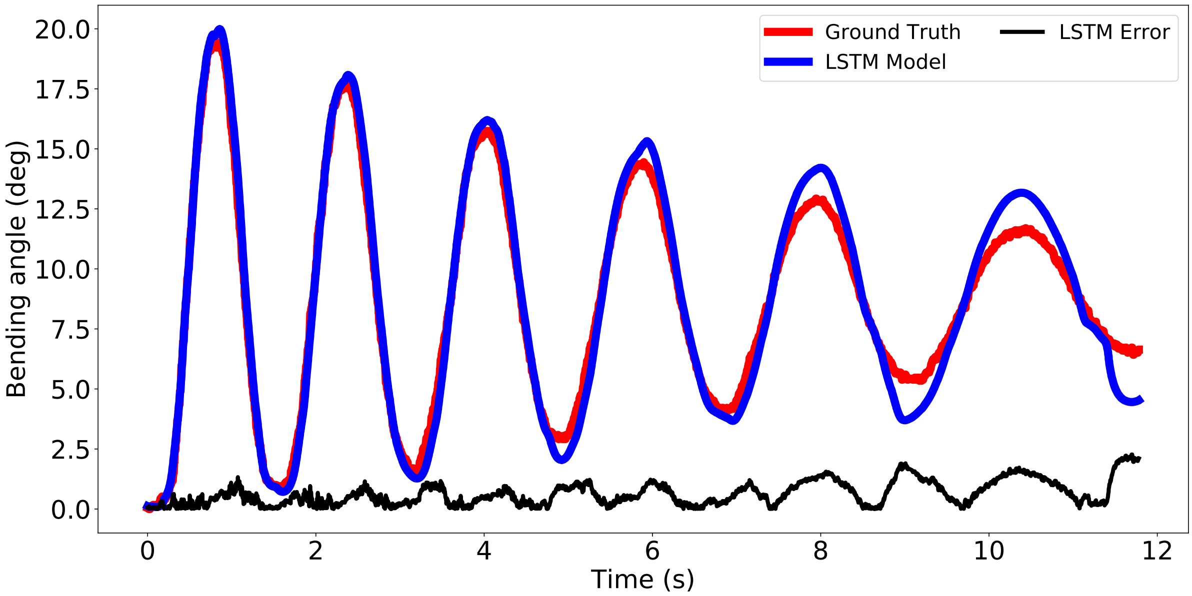

Figure 7.8: 2. Attenuated down-chirp sine wave with shifted baseline (LSTM)

At the first and second waves of the test data, the output of the LSTM can fit the ground truth in increasing, decreasing, the peak, and the valley of the wave. However, when the frequency of the test data is decreasing, the output of the LSTM starts to show worse performance, especially at the first last and second last waves. Thus the RMSE and the MAE are higher than the error in the first test data, reaching 0.8 degrees and 2.3 degrees, respectively.

3. Following (5.17), the same test data with A = 0.6, f = 0.3, τ = 0.05 and c = 0.15, an ascending up-chirp sine wave was generated.

   The performance of the LSTM in this type of test data is getting better when the frequency is increasing. However, at the beginning of the test data, the output of the LSTM has relative large errors at the peak and the valley of the wave. It is also clear to observe from the error plotting that the amount of the error is higher than other periods. Even if the RMSE and the NRMSE are also relative low errors equal to 0.5 degrees and 2.34%, respectively.

4. Similar to the proposed DRDPI and SVR model, an arbitrarily varying signal with plateaus test data was used to explore the potential of the LSTM model in general applications.

   The LSTM performs good at the plateau period and the most increasing and decreasing period. At the last valley of the wave, the output of the LSTM shows relative offset with

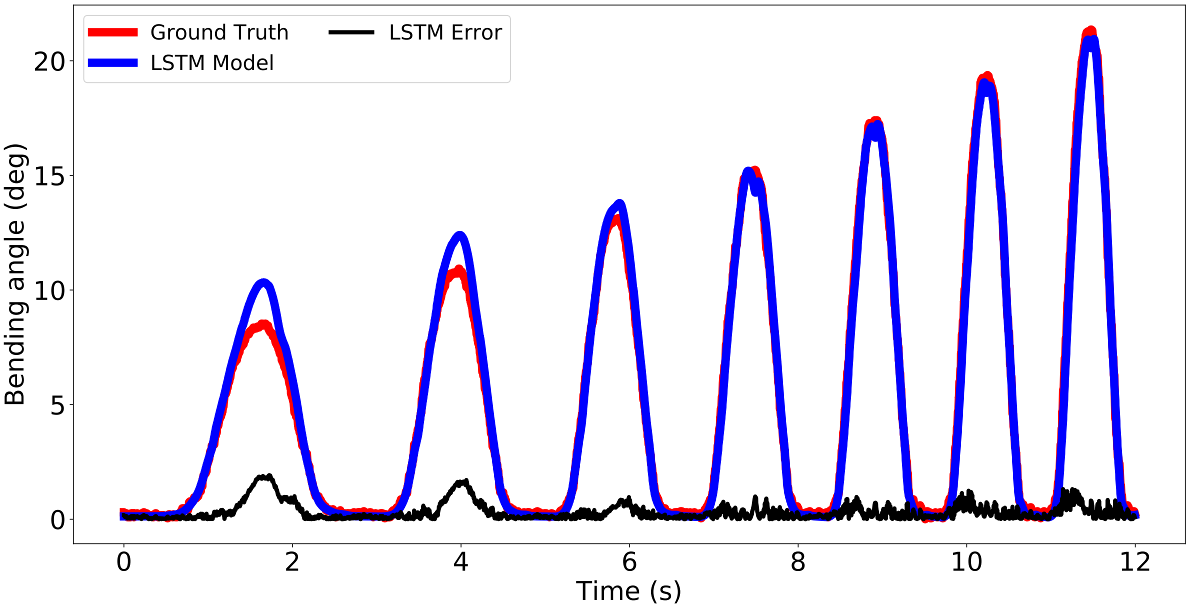

Figure 7.9: 3. Ascending up-chirp sine wave with zero baseline (LSTM)

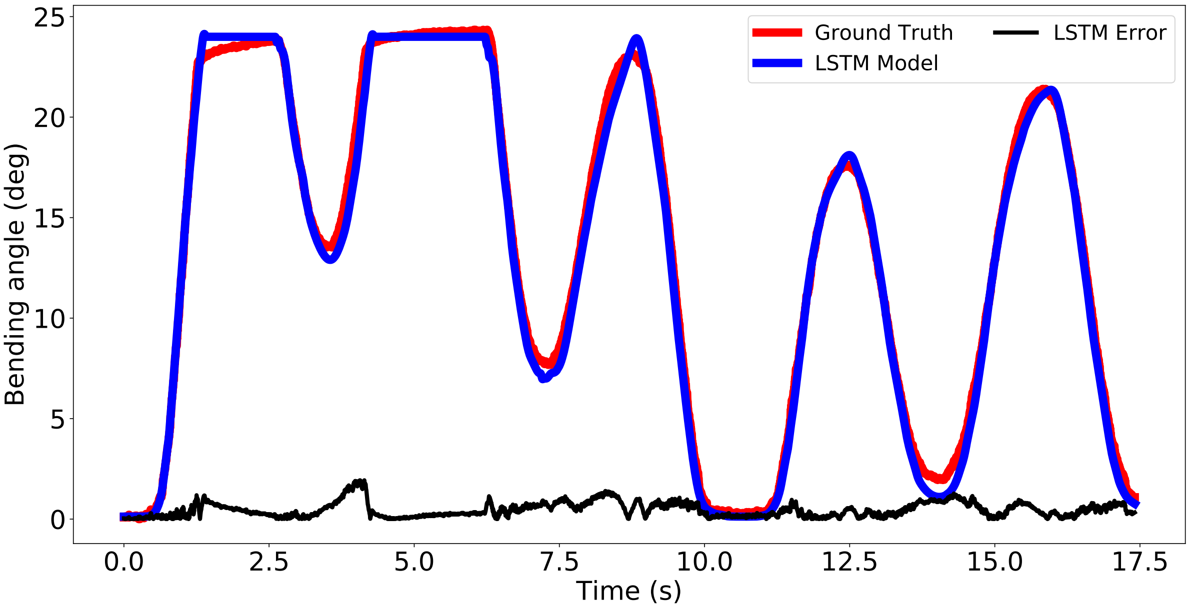

Figure 7.10: 4. Arbitrarily varying signal with plateaus (LSTM)

the ground truth, which leads to the MAE = 1.9 degrees. Generally the performance of the LSTM is acceptable with RMSE = 0.6 degrees and NRMSE = 2.42%.

7.4.2 Discussion of the LSTM model performance

Figure 7.11 illustrates the three metrics of the modeling performance in the LSTM model. It is clearthat the maximum NRMSE appears at the second type of the test data, which equals to 4.2%. Also, the maximum MAE appears at the second type of the test data with 2.3 degrees. The LSTM performs best at the first test data with minimum RMSE = 0.4 degrees because the character of this type is same with the training data.

To sum up, the overall performance of the LSTM shows that this model can model the hysteresis in this PAM-driven robotic catheter system accurately. The LSTM can provide a sub-degree accuracy. Also, the training time and the computation cost is relative small and acceptable. With 4 GB NVIDIA CUDA-capable GPU, the LSTM model takes around 10 minutes. The DRDPI model takes 3.5 hours with CPU (Intel Core i7 CPU @ 2.80GHz with a RAM of 8GB) since there is no widely spread GPU library.

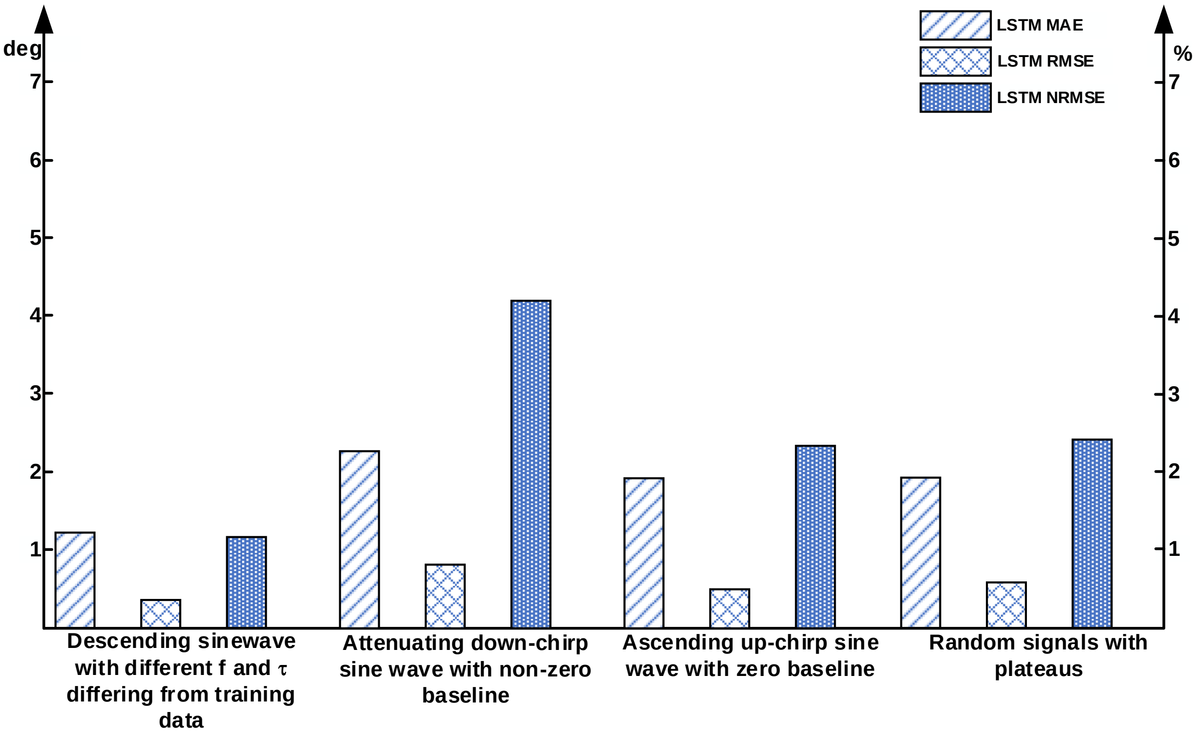

Figure 7.11: Three metrics evaluation of the modeling performance (LSTM)

8 Comparison of modeling performance between proposed models

In this thesis, three models are proposed for hysteresis modeling of a PAM-driven robotic catheter system, namely Deadband Rate-Dependent Prandtl-Ishlinskii, Support Vector Regression, and Long Short-Term Memory model. The DRDPI model is an analytical model while the other two models are learning-based model (see Figure 8.1). Since all the unknown parameters of DRDPI can be listed in Table. 5.1, the training process is relative transparent and classified as white box. The SVR model is learning-based model, more specifically it is a machine learning model, considered as a gray box method. The last proposed model-LSTM model, is a deep-learning method and considered as a black box method, since all the weights and bias of the LSTM cell are impossible to be interpreted.

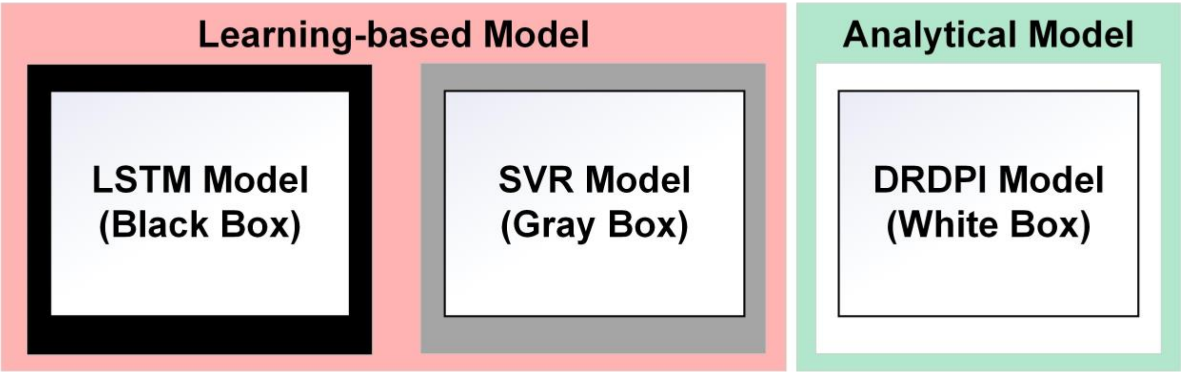

Figure 8.1: Classification of proposed models

In order to evaluate the performance of these models, also three metrics are used for the comparison of these three models, namely Maximum Absolute Error, Root Mean Square Error, and Normalized Root Mean Square Error. All three proposed models make use of the same training data and the same test data, it is therefore meaningful to compare these models with the four types of the test data. The comparison result of each test data is introduced as follows:

1. Descending sine wave differing from training data

   The first test data has the same character with the training data but with different f and τ. Figure 8.2 shows the performance of all three proposed models and plots the errors. When looking at the peaks in the signal, one can clearly notice that the DRDPI model performs the worst. Also during the increasing period of the wave, the DRDPI

Table 8.1: Quantitative performance of the proposed models in the first test data

Modeling Methods	**Descending sine wave differing from the training data**		
	RMSE (deg)	**MAE (deg)**	**NRMSE**
DRDPI	**1.404**	**6.226**	**4.54%**
SVR	**0.758**	**1.833**	**2.45%**
LSTM	**0.363**	**1.227**	**1.17%**

model's output shows an offset with respect to the ground truth, which leads to a large error that can be seen clearly in the error plot. Moreover, because of the SVR's worse performance at each deadband, the RMSE of the SVR is also higher than that of LSTM. A local zoom in on the Figure 8.2 also further illustrates the performance of these three models.

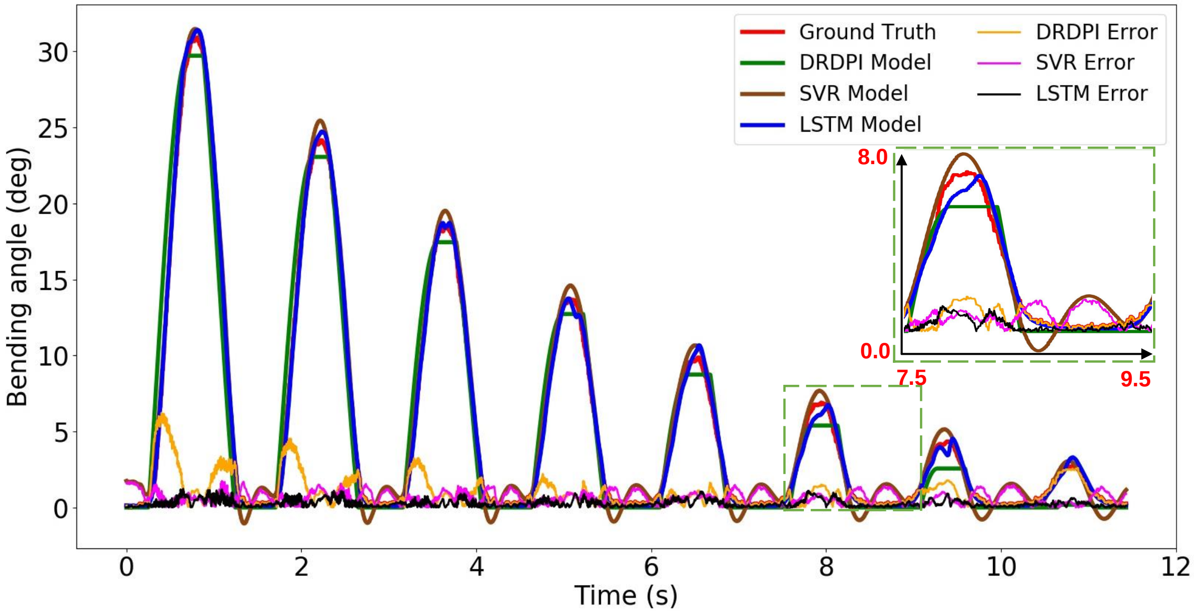

Figure 8.2: Descending sine wave differing from training data

Table. 8.1 provides the obtained results. The cell with green background highlights model with best performance for each metric. It can be seen that the LSTM model has the best performance of all three metrics in this type of the test data. Compared to the DRDPI and SVR model, the LSTM model performs 74.14% and 52.11% better in the RMSE, respectively. It is also noteworthy that the MAE of the DRDPI model is much larger than the SVR and LSTM model, which proves that the learning-based models are more stable than the analytical model for these experiments.

2. Attenuated down-chirp sine wave with shifted baseline

Based Figure 8.3, it is hard for the proposed models to model the hysteresis under this type of the test data when the frequency is decreasing. All three models show worse performance near the end of the ground truth. This leads to higher RMSE compared to the first type of the test data. The DRDPI model has difficulty to follow the trajectory, the other two learning-based models show a better performance. It is therefor that the DRDPI model has the worst performance in all three metrics, which can be proved in details in the Table. 8.2.

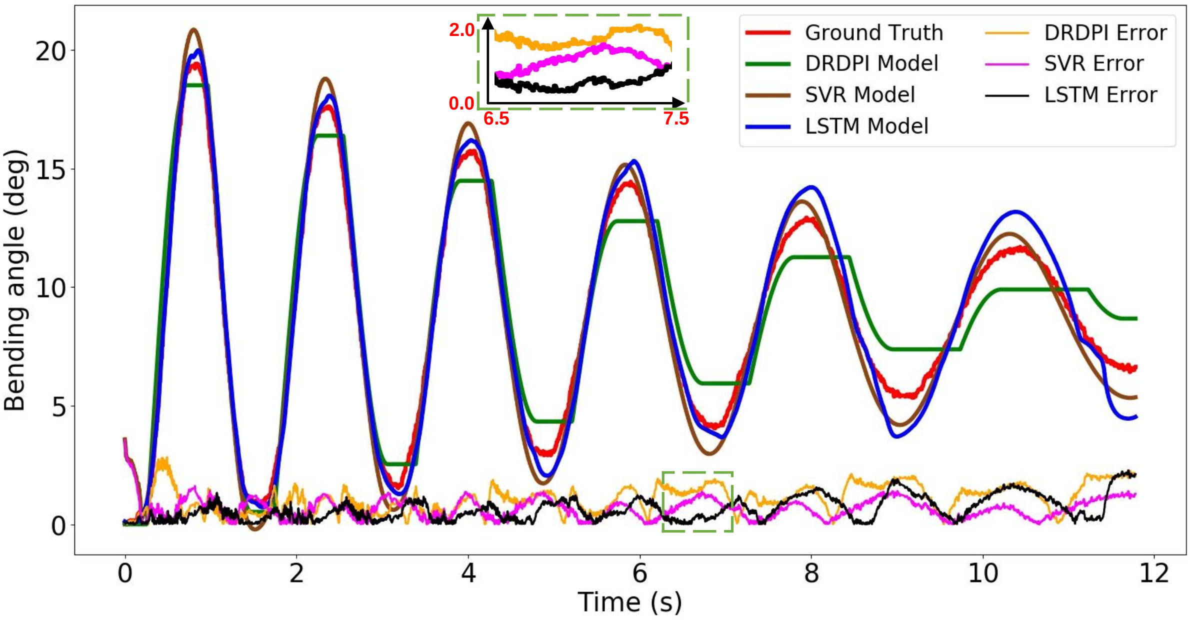

Figure 8.3: Attenuated down-chirp sine wave with shifted baseline

It is noteworthy that two out of three green cells appear for the SVR model. Compared to the DRDPI model, the SVR model improves the performance with 34.67% at the RMSE and 34.71% at NRMSE. However, because the disturbance at the beginning of the SVR model, the SVR model has the worst performance for MAE with 3.5 degrees. Although the SVR model has a better performance, the LSTM has a more stable and accepted performance, with only less 2.32% and 2.38% lower accuracy compared to SVR.

3. Ascending up-chirp sine wave with zero baseline

   The third set of test data has an ascending pattern that is not contained in the training data. The performance of the SVR and LSTM improve with growing frequency, whereas the DRDPI model achieves a similar level of performance in the first two loops. However, the performance of the DRDPI starts to deteriorate until reaching a MAE of 4.3 degrees in the loading phase at the last loop.

   Compared to the performance of the three models at the second test data, the green cells at the Table. 8.3 are exactly opposite to the Table. 8.2. The LSTM has the best

Table 8.2: Quantitative performance of the proposed models in the second test data

Modeling Methods	**Attenuated down-chirp sine wave with shifted baseline**		
	RMSE (deg)	**MAE (deg)**	**NRMSE**
DRDPI	**1.220**	**2.830**	**6.28%**
SVR	**0.797**	**3.535**	**4.10%**
LSTM	**0.816**	**2.271**	**4.20%**

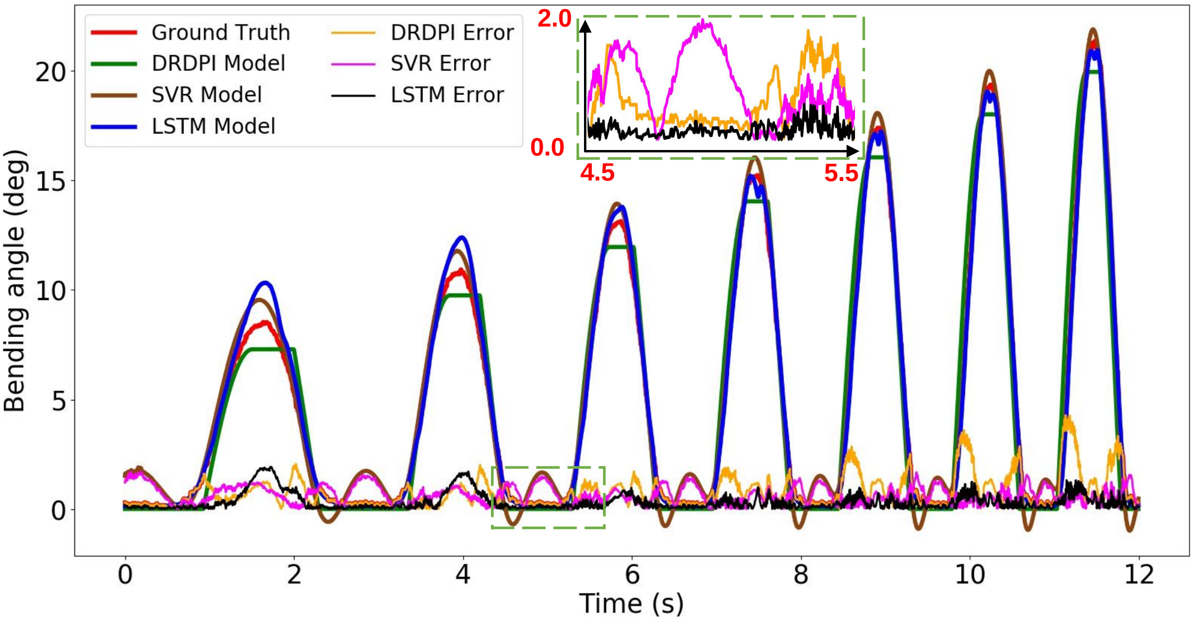

Figure 8.4: Ascending up-chirp sine wave with zero baseline

Table 8.3: Quantitative performance of the proposed models in the third test data

Modeling	Ascending up-chirp sine wave with zero baseline		
Methods	RMSE (deg)	MAE (deg)	NRMSE
DRDPI	1.066	4.284	4.99%
SVR	0.760	1.687	3.56%
LSTM	0.500	1.924	2.34%

performance at the RMSE and the NRMSE with 53.10% and 53.11% improvement compared to the DRDPI model. Moreover, the MAE of the DRDPI model is much larger than the SVR model with 60.62%, where the SVR model has the lowest MAE at this type of the test data. Based on the performance of the DRDPI on the MAE metric, the DRDPI model shows a unstable character.

4. Arbitrarily varying signal with plateaus

   Among of these four types of the test data, the arbitrarily varying signal is the most practical data compared to the other three types. The two learning based models show a good performance at the plateaus of the signal, while the DRDPI model creates a lager MAE with 3.0 degrees here. Moreover, at the rough of the wave the DRDPI model also shows a worse performance than the other two models.

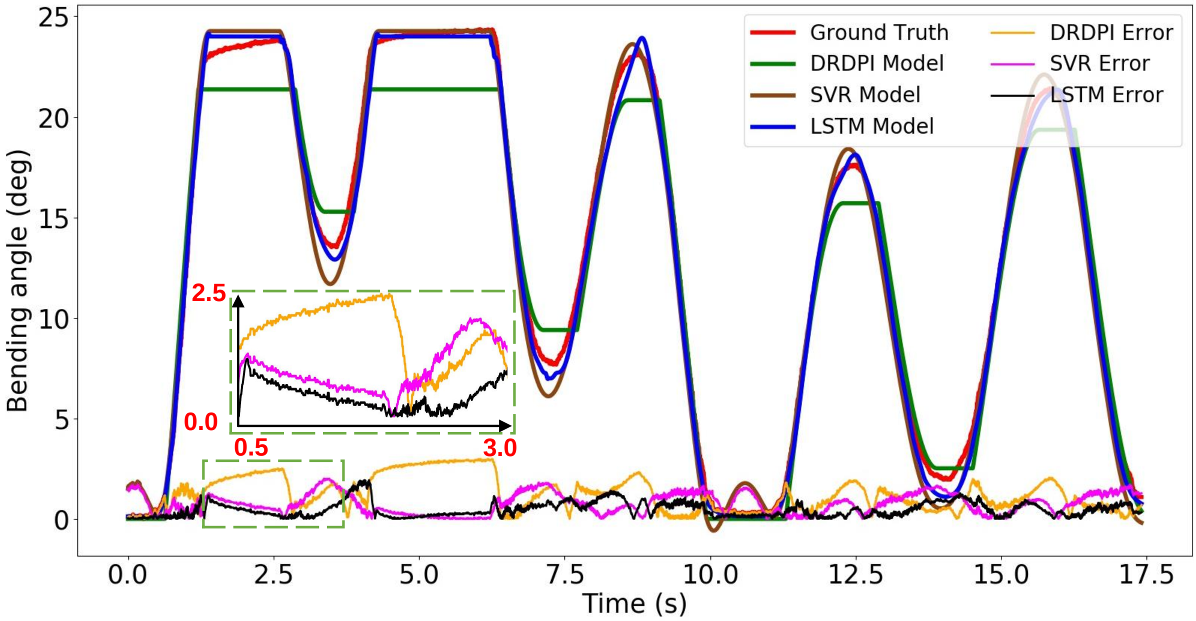

Figure 8.5: Arbitrarily varying signal with plateaus

Table. 8.4 demonstrates the concrete number of the three metrics. All three green cells are shown in the LSTM model, which indicate the advantage of this model. Compared

Table 8.4: Quantitative performance of the proposed models in the fourth test data

Modeling Methods	**Arbitrarily varying signal with plateaus**		
	RMSE (deg)	**MAE (deg)**	**NRMSE**
DRDPI	**1.479**	**2.979**	**6.07%**
SVR	**0.920**	**2.010**	**3.78%**
LSTM	**0.588**	**1.932**	**2.42%**

Table 8.5: optimization results of three models

Modeling Methods	**Iterations**	**Training Loss (deg^2)**	**Mean of Relative Change**
DRDPI	**1000**	**0.344**	**0.013%**
SVR	**100**	**0.002782**	**0.708%**
LSTM	**50**	**0.00056**	**0.190%**

to the analytical modelDRDPI, the LSTM improves the RMSE, MAE, and NRMSE with 60.24%, 35.14%, and 60.13%, respectively. In this practical signal the LSTM performs the best, and also the SVR model proforms better than the DRDPI model. It proofs that when the signal is much more complicated, the learning-based models have more advantage than the analytical model.

In order to compare these three models more efficient, also the training data of all the models is the same, it is meaningful to list the training process result. Table. 8.5 shows the parameters of three models training process. The DRDPI goes through 1000 iterations before it reaches the plateau, while the SVR and the LSTM model only go through 100 and 50 iterations, respectively. Moreover, the last training loss of the DRDPI equals to 0.344 deg^2, but the last training loss of the SVR is only around 0.0028 deg^2 and the LSTM is more less with 0.00056 deg^2. The DRDPI model has a larger training loss even if it has gone through 20 times iterations than the LSTM. The learning efficiency and speed of the two learning-based models are much larger than the analytical model. To sum up, the learning-based, especially the LSTM model, has less training time and can model the hysteresis more accurate.

9 Inverse Deadband Rate-Dependent Prandtl-Ishlinskii model

9.1 Overview of the Inverse DRDPI model

9.1.1 Proposed unknown functions of the IDRDPI model

In the previous chapter 5, the DRDPI model has been introduced and proved to be able to model the hysteresis in the PAM-driven robotic catheter system. In the modeling procedure of the DRDPI model, the input of the model is the input pressure of the PAM and the output is the bending angle. However, the goal of the DRDPI model is to compensate for the hysteresis of the system and achieves linear relationship between the input and the output. Thus the Inverse Deadband Rate-Dependent Prandtl-Ishlinskii (IDRDPI) model is proposed to compensate for the hysteresis.

In the section 5.2 all unknown functions of the DRDPI model are introduced. The inverse model of the proposed IDRDPI model is derived from the DRDPI model. An identity mapping between the input bending angle and input pressure is obtained by applying the IDRDPI model as a feed forward compensator. The mapping relationship is shown as follows:

$$u_r(t) = E(E^{-1}(u(t))) \rightarrow u_r = u \tag{9.1}$$

The inverse model of the proposed DRDPI E^{-1} can be calculated as [55]:

$$u_r(t) = E(E^{-1}(u(t))) = y^{-1}(Z^{-1}(u(t))) \tag{9.2}$$

The (5.6) introduces the threshold function of the DRDPI model, a threshold function in the IDRDPI model is also shown as follows [54]:

$$\rho_1(\dot{u}(t)) = a_0 r_1(\dot{u}(t)) \tag{9.3}$$

$$\rho_{i+1}(\dot{u}(t)) = \rho_i(\dot{u}(t)) + \sum_{j=0}^{i} a_j(r_{i+1}(\dot{u}(t)) - r_i(\dot{u}(t))) \tag{9.4}$$

Based on the inverse threshold function and weighting constants c_i, the output of the IDRDPI model is formulated [44]:

$$y^{-1}(t) = c_0 u(t) + \sum_{i=1}^{N} c_i \Phi_{\rho_i}(t) \tag{9.5}$$

The weighting constants of the inverse model $c_0, c_0, ..., c_n$ are calculated based on the weighting constants in the (5.7) of the DRDPI model [63]:

$$c_0 = \frac{1}{a_0}, c_i = -\frac{a_i}{(a_0 + \sum_{i=1}^{j} a_i)(a_0 + \sum_{i=1}^{j-1} a_i)} \tag{9.6}$$

Besides the inverse of the rate-dependent part of the DRDPI model, the deadband of the DRDPI model should also be derived. The inverse model of the deadband function Z^{-1} is derived from weighted deadband operators Λ_{b_i} and shown as follows [40]:

$$Z^{-1}(t) = \sum_{i=-k}^{k} q_i \Lambda_{b_i}(u(t)) \tag{9.7}$$

where q_i and b_i are weighting constants and thresholds of the inverse deadband function Z^{-1}. The threshold function b_i of the inverse deadband function and constants q_i are computed based on the d_i and g_i in the deadband function Z in (5.5). Inverse threshold and constants corresponding to positive indices are introduced as follows [40]:

$$b_j = \sum_{i=0}^{j} g_i(d_j - d_i), \quad for \quad j = 0, ..., k \tag{9.8}$$

and

$$q_0 = \frac{1}{g_0}, q_j = -\frac{g_i}{(g_0 + \sum_{i=1}^{j} g_i)(g_0 + \sum_{i=1}^{j-1} g_i)} \quad for \quad j = 1, ..., k \tag{9.9}$$

Similarly, constants and thresholds of the inverse model corresponding negative indices are represented as follows [40]:

$$b_j = \sum_{i=k}^{0} g_i(d_j - d_i), \quad for \quad j = -k, ..., -1 \tag{9.10}$$

and

$$q_0 = \frac{1}{g_0}, q_j = -\frac{g_i}{(g_0 + \sum_{i=j}^{-1} g_i)(g_0 + \sum_{i=j+1}^{-1} g_i)} \quad for \quad j = -k, ..., -1 \tag{9.11}$$

Unlike the modeling process, the unknown parameters of the IDRDPI model can be directly derived from the modeling parameters based on the formulas (9.1)-(9.11). Normally, the compensation of the hysteresis based on the IDRDPI model is related to the DRDPI model. After the training process of the DRDPI, all unknown parameters can be obtained from the modeling process. In the next subsection, the preliminary test of the direct IDRDPI model will be validated.

9.1.2 Preliminary test of the IDRDPI model

In order to validate whether the direct inverse DRDPI model can compensate for the hysteresis in this PAM-driven robotic catheter system, it is meaningful first to compare the compensated pressure from the output of the direct IDRDPI and the input pressure. Figure 9.1 illustrates the comparison between these two pressures.

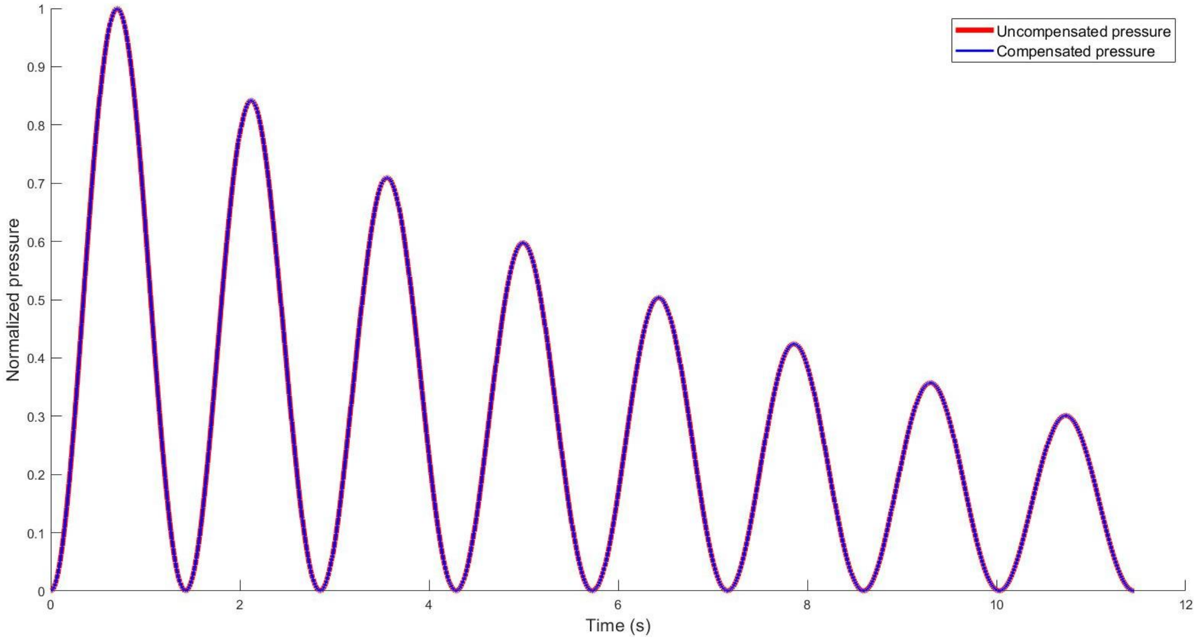

Figure 9.1: Output of the direct IDRDPI

It is clear to observe that the tendency of the compensated pressure from the IDRDPI model is same with the uncompensated pressure. Thus if this compensated pressure is input into the setup, the deadband will not be compensated for and also the hysteresis still exists. Therefore, an optimization for the IDRDPI model is needed.

9.2 Optimization of the Inverse DRDPI model

Throughout the preliminary test of the direct IDRDPI model, this model cannot exactly characterize the hysteresis in this experimental setup. In order to compensate for the hysteresis, a optimization of the IDRDPI model [50] is introduced in this section.

Figure 9.3 illustrates the optimization process of the IDRDPI model. The optimization of the IDRDPI model can be divided into three steps:

1. Step 1: The Generalized Prandtl-Ishlinskii (GPI) identification and the its inverse.

   The GPI proposed in the step 1 is based on the [27] and [64]. The output of the GPI model can be formed as follows [65]:

$$y_p(k) = q\gamma(x(k)) + \sum_{i=1}^{n} p(r_i) F_{r_i}^{\gamma}[x](k) \tag{9.12}$$

with,

$$F_{r_i}^{\gamma}[x](k) = max(\gamma(x) - r_i, min(\gamma(x) + r_i, F_{r_i}^{\gamma}[x](k-1))) \tag{9.13}$$

where n is the number of the operators and k represents the time points. The following envelope function is used in this GPI model [66]:

$$\gamma(x(t)) = c_0 \tanh[c_1 x(t) + c_2] + c_3 \tag{9.14}$$

where $c_0 > 0$, $c_1 > 0$, c_2 and c_3 are constants that need to be identified with the same GA methods as mentioned before. The following equations are used to define the density function $p(r_i)$ and the threshold function r_i [66]:

$$p(r_i) = \rho e^{-\tau r_i} \tag{9.15}$$

and

$$r_i = \beta i \tag{9.16}$$

where ρ, β, and τ are constants that also will be determined by the GA identification process.

After the modeling process of the GPI model, the inverse model is also proposed for the hysteresis compensation and the output of the inverse GPI model is shown as follows [65]:

$$y'_p(k) = \gamma^{-1}\left(q' y_p(k) + \sum_{i=1}^{n} p_i' x_i(k)\right) \tag{9.17}$$

with,

$$x_i(k) = max\left\{y_p(k) - r_i', min\left\{y_p(k) + r_i', x_i(k-1)\right\}\right\} \tag{9.18}$$

$$r_i' = q r_i + \sum_{j=1}^{i-1} p_j (r_i - r_j) \tag{9.19}$$

$$q' = \frac{1}{q}, \quad p_i' = \frac{-p_i}{(q + \sum_{j=1}^{i} p_j)(q + \sum_{j=1}^{i-1} p_j)} \tag{9.20}$$

$$\gamma'^{-1}(x) = \frac{1}{2c_1} \ln\left(\frac{x + c_0 - c_3}{c_0 + c_3 - x}\right) - \frac{c_2}{c_1} \tag{9.21}$$

where $\gamma'^{-1}(x)$ is the inverse envelop function, p_i' and r_i' are the weights and thresholds of the inverse GPI model.

Table 9.1: Unknown parameters of the GPI model

Model	Unknown parameters
Proposed GPI	$c_0, c_1, c_2, c_3, \beta, \rho, \tau, q$

Table 9.2: Identified parameters of the GPI model under 0.2 Hz

Parameter	c_0	c_1	c_2	c_3	β	ρ	τ	q
Value	10	0.715	-1.134	7.435	1.084	9.875	1.684	0.714

To sum up, there are in total 8 unknown parameters should be identified in the GPI model, which are shown in the Table. 9.1. All the unknown parameters are identified by the GA introduced in the previous chapters. Also, because the GPI model is rate-independent, the GPI model will be identified under each frequency and the parameter in each frequency is also different.

2. Step 2: Data sets for identification of the unknown parameters of the inverse DRDPI.

   As mentioned before, the proposed GPI model is a rate-independent model, which means the model need to be identified at each different frequency. The training data is generated following the (4.1) under f = 0.2 Hz, f = 0.4 Hz, f = 0.6 Hz, and f = 0.8 Hz with τ = 0.1. The identification results of the unknown parameters are listed in the Table. 9.2, Table. 9.3, Table. 9.4, and Table. 9.5 under f = 0.2 Hz, f = 0.4 Hz, f = 0.6 Hz, and f = 0.8 Hz, respectively.

   The inverse model is based on the parameters from the modeling process, all the parameter will be calculated and applied under each frequency. The trained inverse GPI model is the training data source of the inverse DRDPI model. Four corresponding inverse GPI models could be obtained by directly inverting the previously identified GPI model. The inverse GPI models were able to generate training data for the IDRDPI model. In this thesis, 150000 virtual training data is generated for the IDRDPI model.

3. Step 3: Identification the unknown parameters of the inverse DRDPI model.

   All functions of the IDRDPI model are introduced in subsection 9.1.1, based on the training data from the inverse GPI model, the identification process using GA Toolboxin MATLAB®. The identification process was performed on CPU (Intel Corei7-7700 CPU @ 2.80GHz with a RAM of 8.00GB), since there was no wide-spread library for a GPU-based GA training. The whole identification procedure of the IDRDPI model took

Table 9.3: Identified parameters of the GPI model under 0.4 Hz

Parameter	c_0	c_1	c_2	c_3	β	ρ	τ	q
Value	4.798	0.543	-0.966	3.237	0.490	8.945	1.923	1.418

Table 9.4: Identified parameters of the GPI model under 0.6 Hz

Parameter	c_0	c_1	c_2	c_3	β	ρ	τ	q
Value	7.751	0.602	-0.971	5.037	1.096	9.919	1.272	0.745

Table 9.5: Identified parameters of the GPI model under 0.8 Hz

Parameter	c_0	c_1	c_2	c_3	β	ρ	τ	q
Value	6.453	0.556	-0.803	3.418	1.162	9.932	1.023	0.948

around 3.5 hours, which is basically same with the DRDPI identification process. The identification of the IDRDPI model took 300 iterations. The unknown parameters of the IDRDPI model is shown in the Table 9.6. The mean of the relative change in the last 20% of the training data, the iterations comparison loss was just 0.07% when the stop condition was activated, which proofed that the model was fully converged. The well-trained IDRDPI model will be applied for further testing in the next section.

9.3 Performance of the Inverse DRDPI model for position control

9.3.1 Performance evaluation of the IDRDPI model

In order to validate the IDRDPI model, same with the logic of the modeling test process, four different trajectories are also designed for the validation of the performance. The training data of the IDRDPI model is descending sine wave with zero base line and non-zero base line in total 150000 data points. The four trajectories and the performance of the IDRDPI model are introduced as follows:

1. Descending sinusoidal trajectory

   A descending sinusoidal trajectory is generated as follows:

   $$\theta(t) = Ae^{-\tau t}(\sin(2\pi(f+ct)t - \frac{\pi}{2}) + 1) \qquad [\text{deg}] \tag{9.22}$$

   In this equation a trajectory is generated with f = 0.3 and τ = 0.05. The generated trajectory is as input to the IDRDPI model, the output of the IDRDPI model is compensated pressure, which can compensate for the hysteresis of the experimental setup. The compensated pressure of the trajectory form the (9.22) is shown in the Figure 9.3.

Table 9.6: Identified parameters of the IDRDPI model

Parameter	α_0	α_1	α_2	β_0	β_1	β_2	d_p	d_n	ζ	β	a_0	g_0
Value	7.604	9.964	-1.734	3.710	1.713	1.569	2.964	4.112	5.460	4.428	5.988	0.700

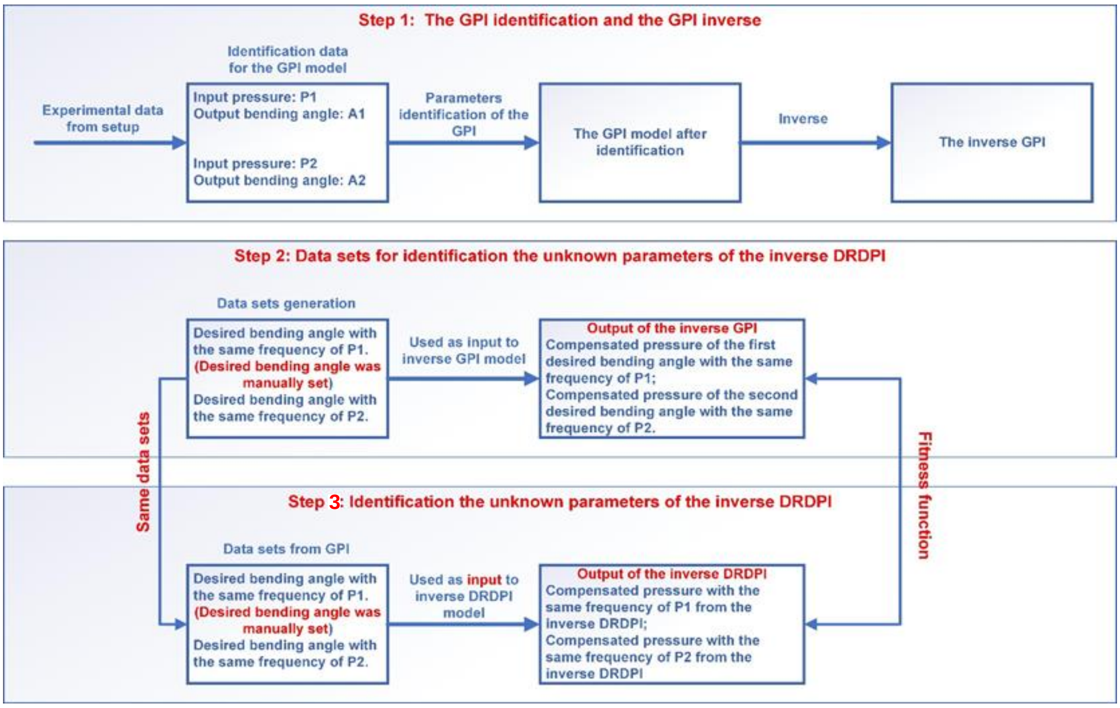

Figure 9.2: Optimization process of the IDRDPI

Because of the deadband of the hysteresis, at each beginning of the trajectory wave, the slope of the compensated pressure is sharp. The compensated pressure is afterwards input into the PAM-driven robotic catheter and the trajectory will be measured.

Figure 9.4 shows the ground truth of the generated trajectory based on the (9.22) and the measured trajectory based on the compensated pressure shown in the Figure 9.3. It is clear to observe a offset at each increasing and decreasing period of the wave. These offsets cause a large RMSE equals to 1.7 degrees and also large MAE equals to 4.4 degrees. Also it is noteworthy that the error of this type of test data shows a regularity increasing at each decreasing period of the wave.

2. Attenuated down-chirp sinusoidal trajectory with shifted baseline

   The trajectory with time-varying frequency is worth to investigate the performance of the IDRDPI. A trajectory with $A = 10$, $B = 8$, $f = 0.5$, and $\tau = 0.05$ was generated based on the following equation:

$$\theta(t) = Ae^{-\tau t}(\sin(2\pi(f + ct)t - \frac{\pi}{2}) + 1) + B \qquad [\text{deg}] \tag{9.23}$$

   The generated trajectory is then input into the IDRDPI model, which can generate the compensated pressure shown in the Figure 9.5. Also at the beginning of each wave, a remarkable pressure increasing can be observed and the tendency of the compensated pressure is same with the trajectory.

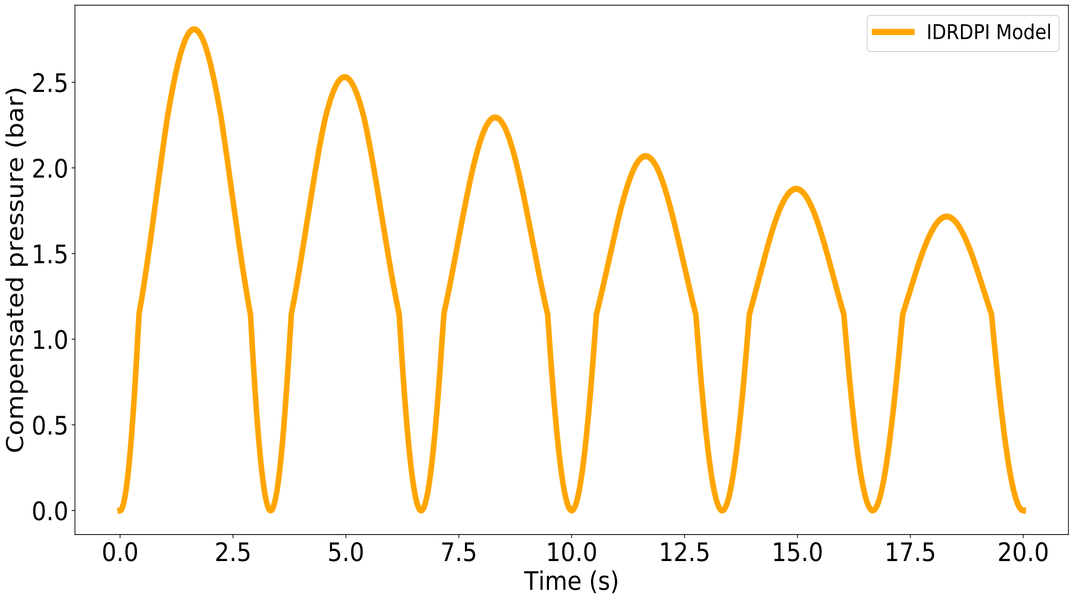

Figure 9.3: Compensated pressure of the descending sinusoidal trajectory (IDRDPI)

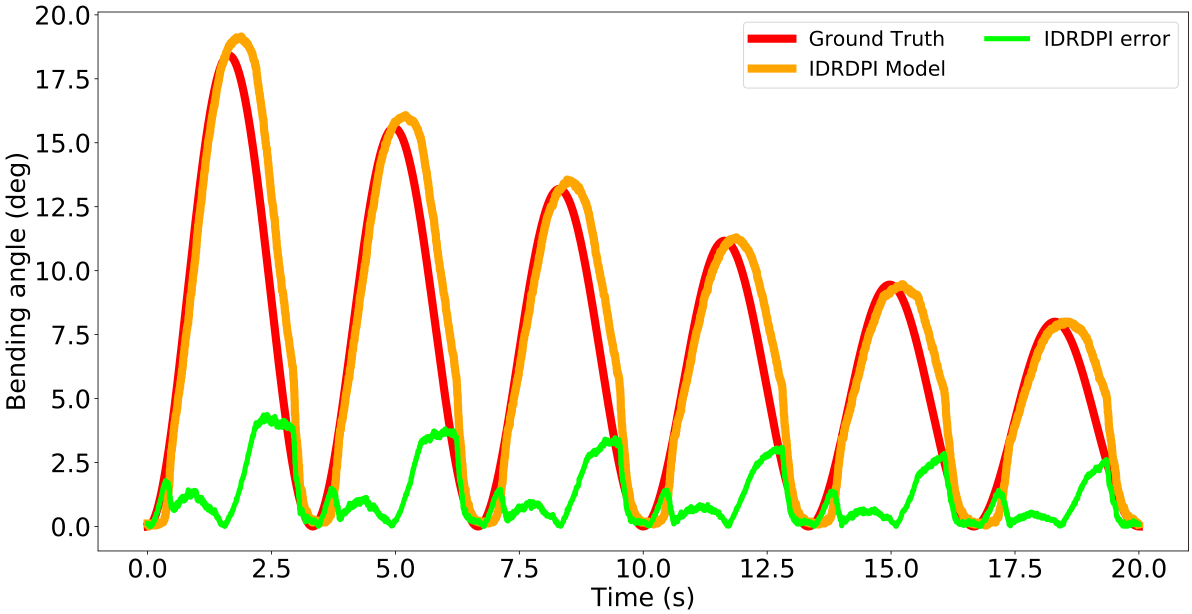

Figure 9.4: 1. Descending sinusoidal trajectory (IDRDPI)

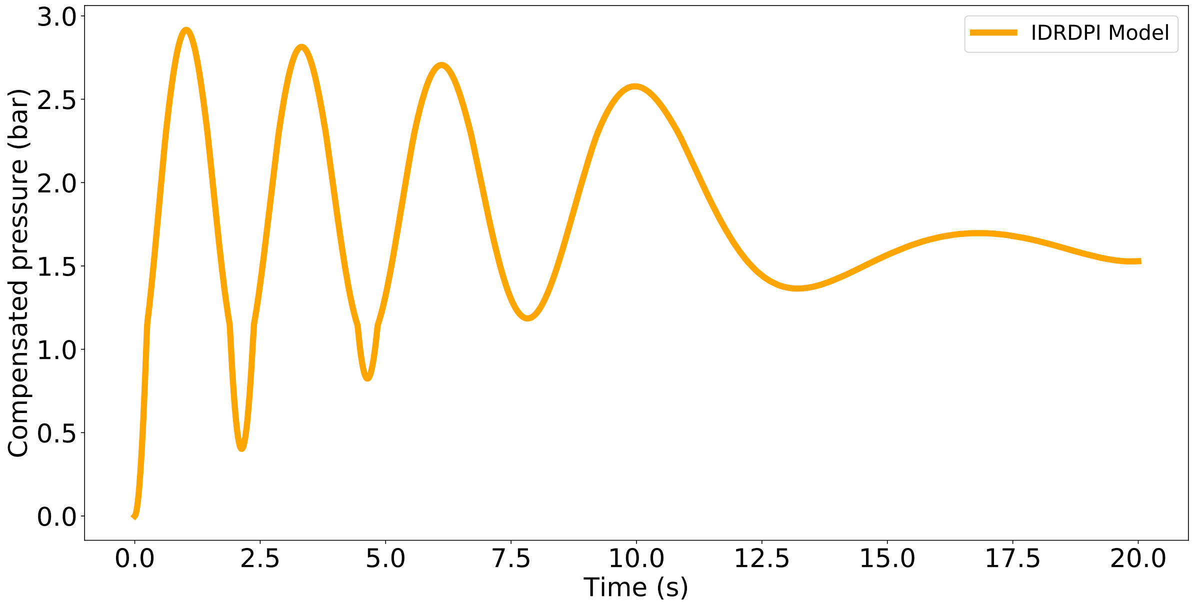

Figure 9.5: Compensated pressure of the attenuated down-chirp sinusoidal trajectory with shifted baseline (IDRDPI)

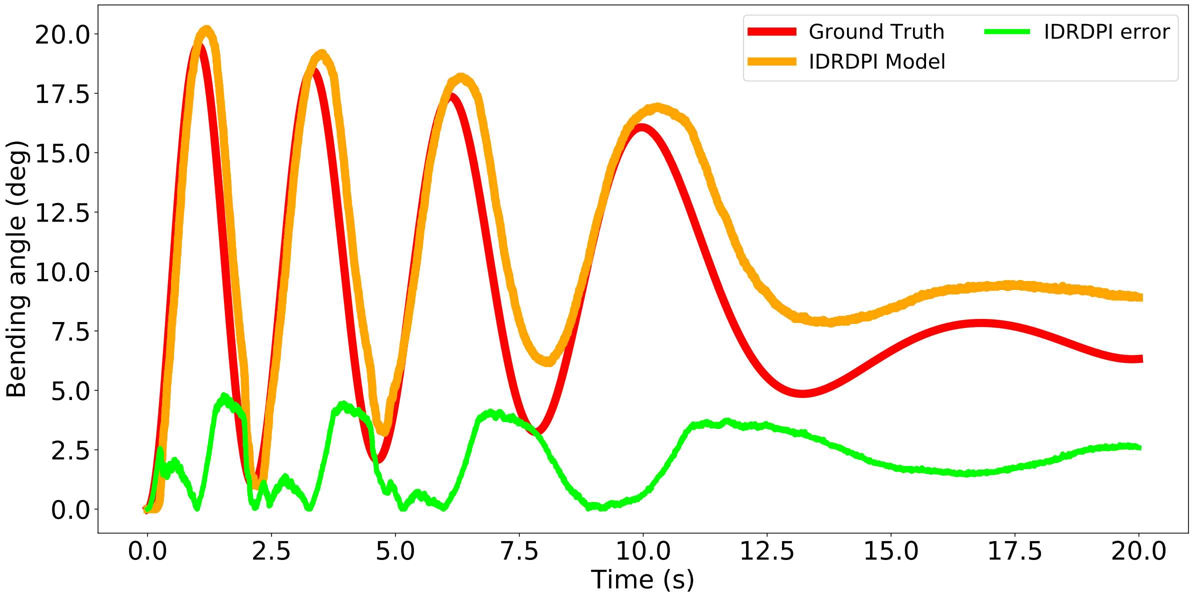

Figure 9.6: 2. Attenuated down-chirp sinusoidal trajectory with shifted baseline (IDRDPI)

The compensated pressure shown in the Figure 9.5 is input into the experimental setup. The resulting measured bending angle compared with the ground truth is shown in the Figure 9.6. The performance of the IDRDPI model is worst than the first type of the test data. The IDRDPI model both fails to accurately predict the peak accurately and under-compensates in the unloading phase, thus causing offsets from the set trajectory, which leads to a larger RMSE equals to 2.5 degrees. Also a large MAE equals to 4.8 degrees occurs when the frequency of the trajectory is decreasing.

3. Ascending up-chirp sinusoidal trajectory with zero baseline

   The third tested trajectory is a frequency-ascending trajectory, and this pattern was also not included in the training data. An ascending up-chirp sinusoidal trajectory is generated following (5.17) with A = 6, f = 0.3 , τ = -0.02 and c = 0.005.

$$\theta(t) = Ae^{-\tau t}(\sin(2\pi(f+ct)t - \frac{\pi}{2}) + 1) \qquad [\text{deg}] \tag{9.24}$$

The input of the IDRDPI model is chosen as this type of the test data, the resulting output of the compensated pressure is shown in the Figure 9.7.

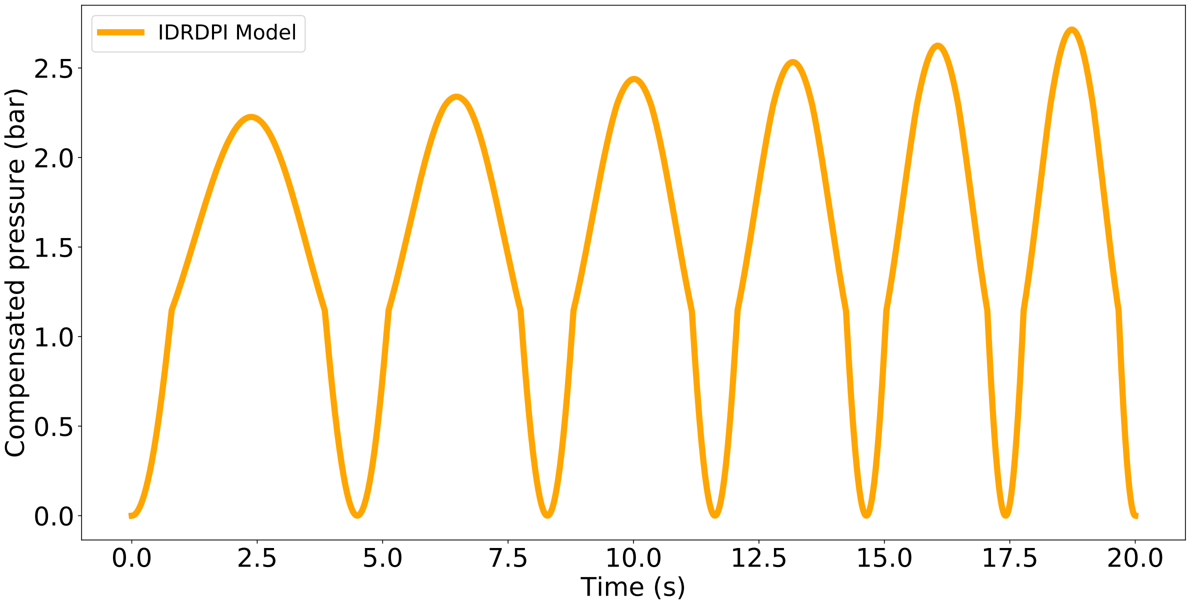

Figure 9.7: Compensated pressure of the ascending up-chirp sinusoidal trajectory with zero baseline (IDRDPI)

The compensated pressure is input into the experimental setup. The matching result between the measured bending angle and the generated trajectory is shown in the Figure 9.8. The error of the IDRDPI model fluctuates periodically, the IDRDPI model fails to accurately predict the peak accurately and under-compensates in the unloading phase, thus causing offsets from the set trajectory. The offsets cause a RMSE with 1.8 degrees and NRMSE with 5.295%.

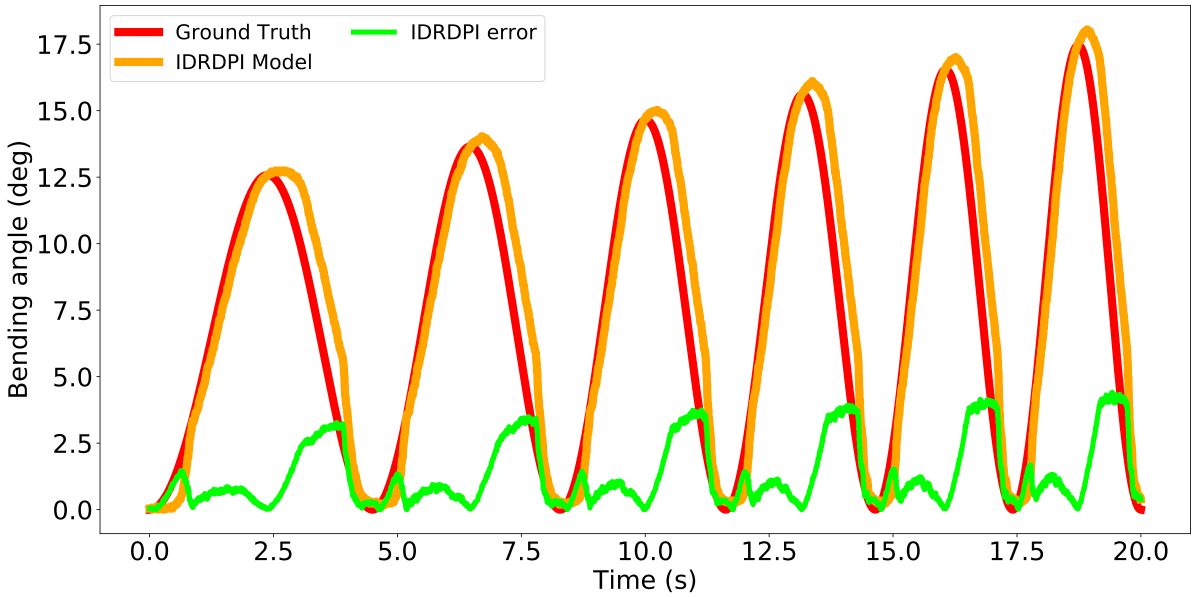

Figure 9.8: 3. Ascending up-chirp sinusoidal trajectory with zero baseline (IDRDPI)

4. Arbitrarily varying trajectory with small variations

   This type of the test data is the most practical trajectory and most challenging test data for the IDRDPI model. The generated trajectory is input into the IDRDPI model and the compensated pressure of this type test data is shown in the Figure 9.9.

   The performance of the IDRDPI mdoel in this type of the test data is the worst among the other three types of the test data. The measured bending angle and the ground truth have a very large offset, which cause very large error in all three metrics with RMSE = 4.0 degrees, MAE = 8.5 degrees, and NRMSE = 7.37%. Moreover, in the region of small variations, the IDRDPI model does not respond to small fluctuations, but rather produces a plateau.

9.3.2 Discussion of the IDRDPI model performance

The histogram in the Figure 9.11 illustrates all three metrics in this position control experiment. All four test trajectories are measured five times to avoid random error and the standard deviations are also represented in the histogram. The IDRDPI model has the best performance in the first type of the test data with the smallest RMSE = 1.7 degrees and standard deviation 0.6 degrees. The worst performance of the IDRDPI model appears at the fourth test data with the largest MAE = 8.5 degrees and standard deviation 2 degrees. To sum up, the IDRDPI model is not able to implement an accurate position control when the trajectory is more arbitrary and more practical.

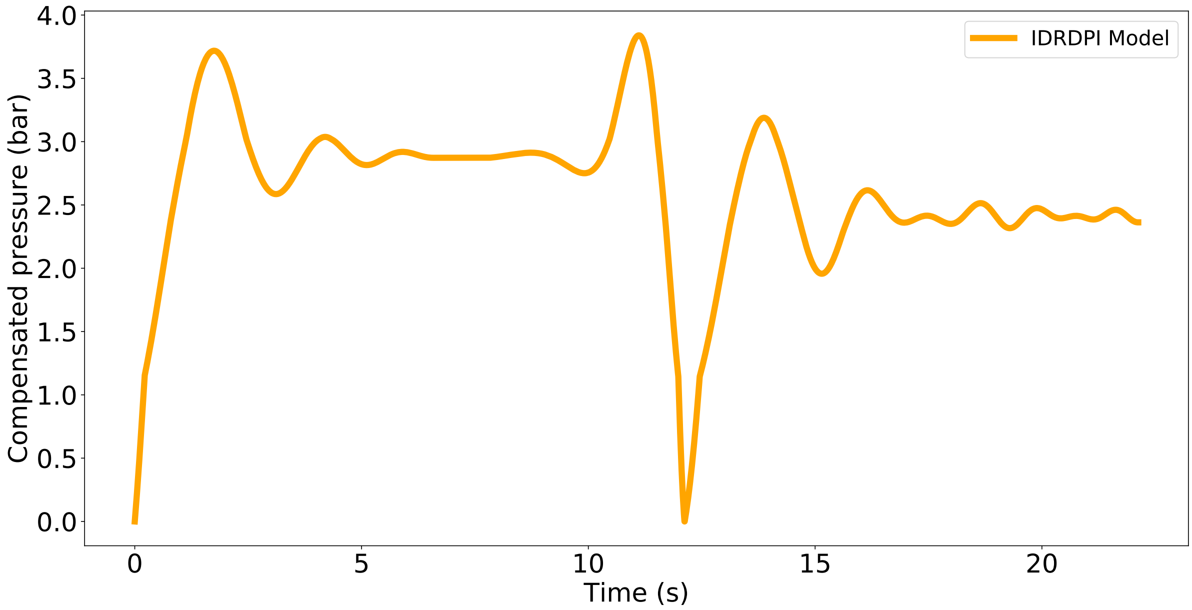

Figure 9.9: Compensated pressure of the arbitrarily varying trajectory with small variations (IDRDPI)

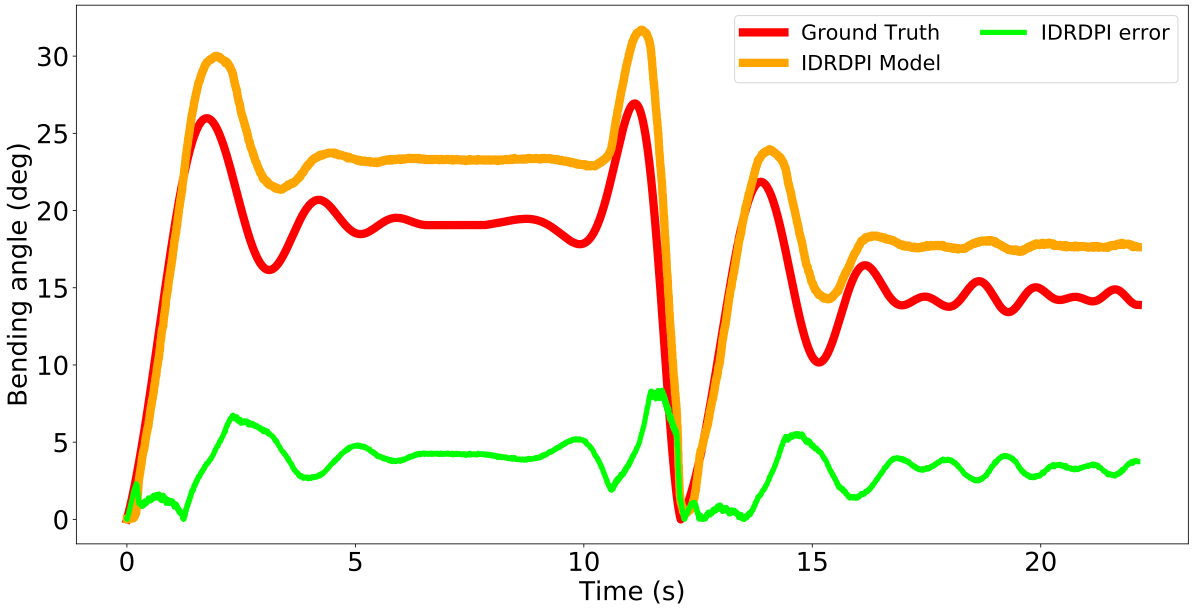

Figure 9.10: 4. Arbitrarily varying trajectory with small variations (IDRDPI)

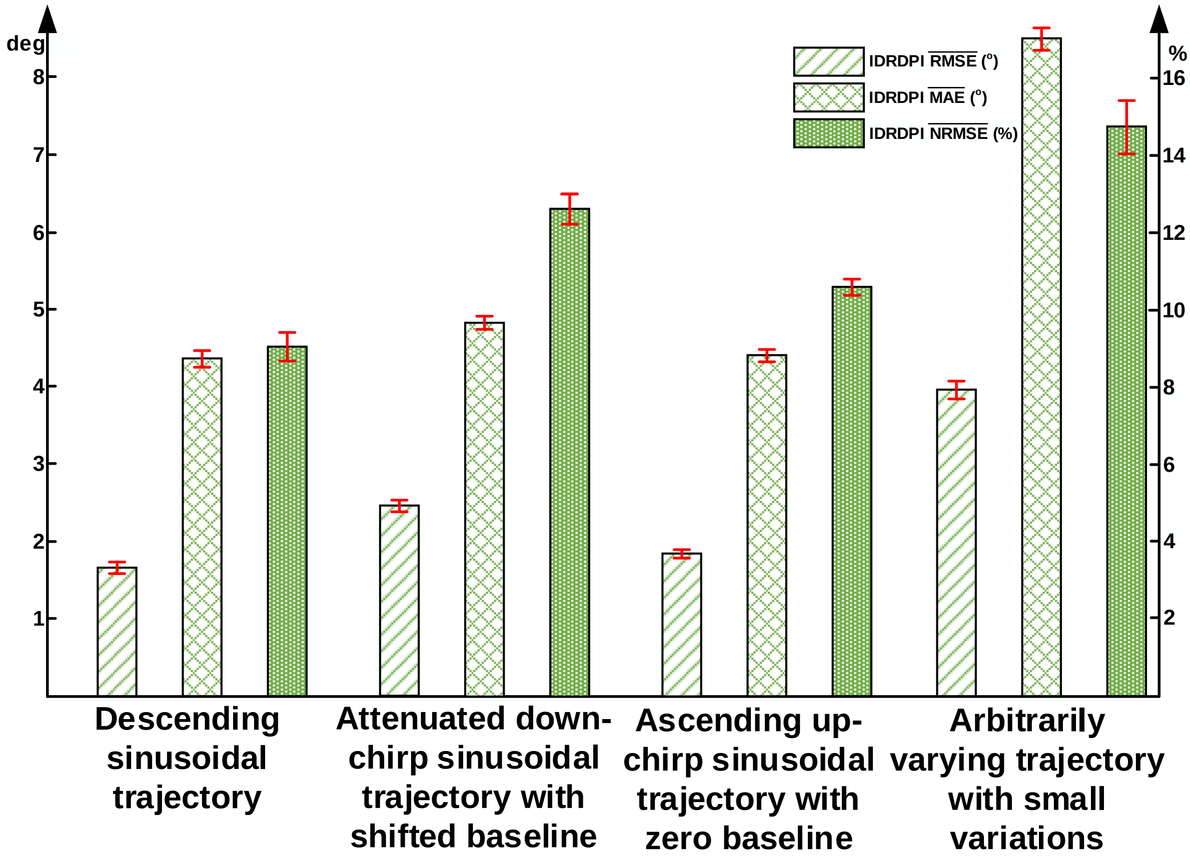

Figure 9.11: Three metrics evaluation of the position control performance (IDRDPI)

10 Control Long Short-Term Memory model

10.1 Structure of the c-LSTM model

Control Long Short-Term Memory (c-LSTM) is the LSTM network for the compensation and control of the robotic catheter. The LSTM in the c-LSTM model is same with the one introduced in the Chapter. 7. However, the input and the output of the c-LSTM are inverted compared to the LSTM model for modeling. In the c-LSTM, the input of the model is the desired trajectory while the output of the model is compensated pressure.

As introduced before, the output at a certain moment is not only determined by the corresponding input but also by the past inputs. It is therefore the historic input should be considered in each input and also a data preprocessing is necessary. Figure 10.1 illustrates the data preprocessing of the c-LSTM model. Compared to the Figure 7.4, the major difference is now the a) is the desired trajectory.

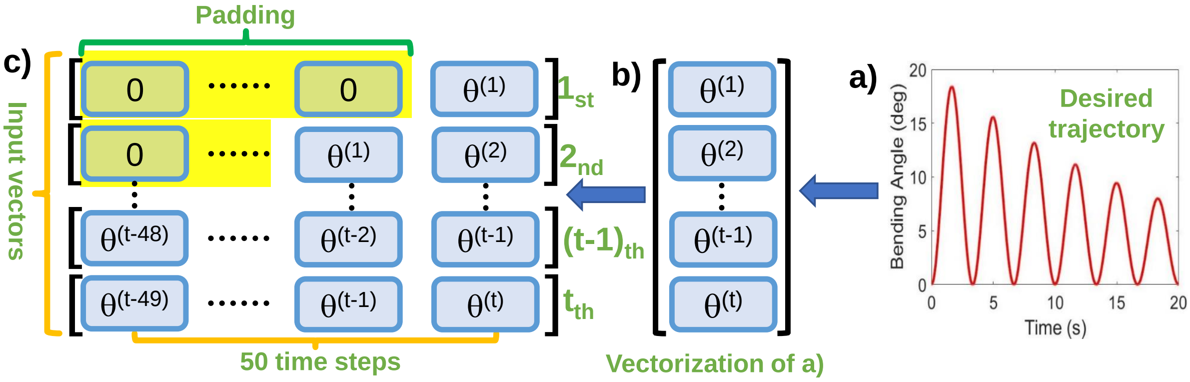

Figure 10.1: c-LSTM input data preprocessing

Similar with the data preprocessing of the modeling process, the data preprocessing is also divided into three parts:

a) The input of the c-LSTM model is the desired trajectory along with the time. The input desired trajectory need to be processed before send to the c-LSTM model.

b) The desired trajectory will go through the vectorization process and be viewed as a matrix with only one column. Each input data is vectored based on the time points.

c) After the b) process, the input desired trajectory will be split into several input vectors based on the time window size. The padding is also adopted at the initialization process.

The window size equals to 50 was shown a good performance based on the previous pilot study [67].

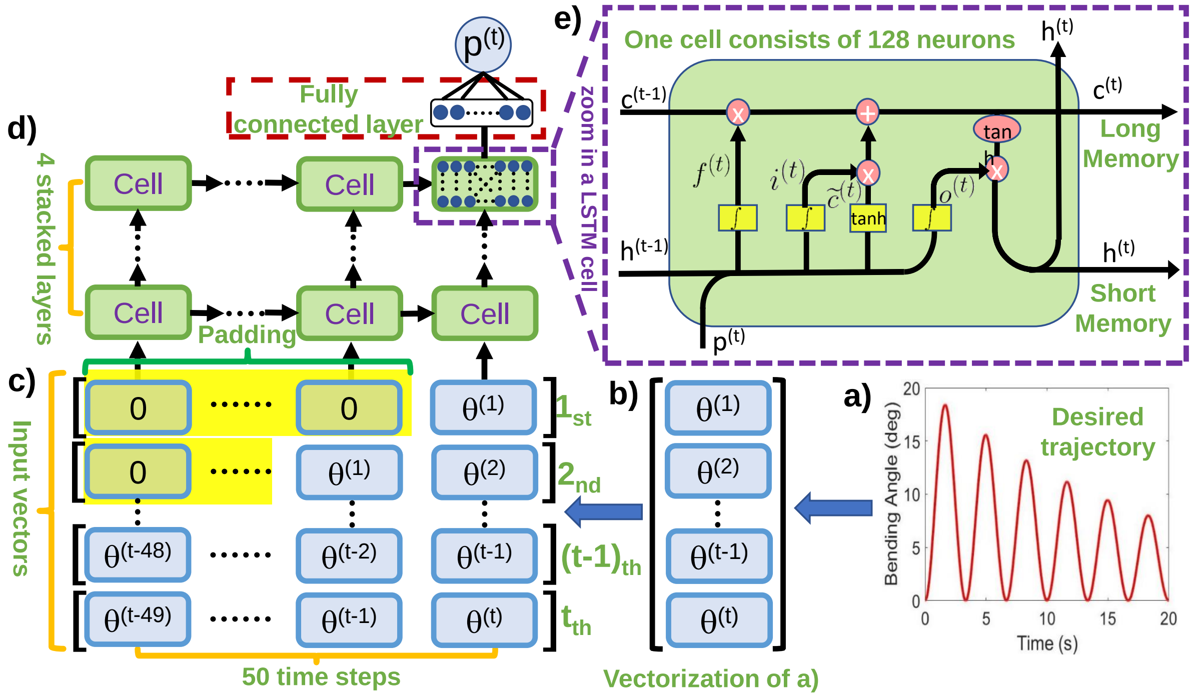

Figure 10.2: c-LSTM structure

Figure 10.2 illustrates the c-LSTM structure for the hysteresis compensation and position control. The c-LSTM structure is divided into five parts:

a) The input of the c-LSTM model is the desired trajectory. The output of the c-LSTM is the compensated pressure.

b) Vectorization of the desired trajectory will be processed in this step. After the vectorization the input data will be more organized.

c) In this step, the input data will be further processed, including the adding of the window size and the padding of the input data.

d) This step shows the whole c-LSTM structure cells and also the input data flowing direction.

e) This figure shows the details of the LSTM cell in the c-LSTM model. The introduction of the LSTM cell has been explained in the section 7.1.

The structure of the c-LSTM is basically same as the LSTM model for the hysteresis modeling. However, the input now for the c-LSTM is the desired trajectory and the compensated pressure as output will be obtained. This compensated pressure can compensate the hysteresis of the experimental setup and follow the trajectory, which can guarantee an accurate position control of the catheter.

10.2 Hyperparameters of the c-LSTM model

Same with other proposed models, the c-LSTM model is also a learning-based model. Therefore the hyperparameters of the c-LSTM are also essential for the performance of the model. The type of hyperparameters of the c-LSTM model are same with the LSTM and are introduced as follows:

- Number of hidden layers

  c-LSTM can also be viewed as a stack of LSTM cells that contain a number of units in an LSTM cell. Therefore the layers of the LSTM will be defined. In this thesis, 2 hidden layers are chosen same as the LSTM for the proposed c-LSTM model.

- Number of neurons per cell

  The number of neurons need also be defined, and the number of neurons is better with the power of 2. However, the training process is complicated than the modeling LSTM. Therefore in this experiment 128 neurons in each c-LSTM cell are adopted.

- Activation function

  As introduced in the modeling LSTM, the activation functions of the LSTM are sigmoid (see Figure 7.2) and tahn (see Figure 7.3).

- Optimizer

  Optimizers play a very crucial role to increasing the accuracy of the model. As discussed before in the modeling LSTM, the Adam optimizer is chosen.

- Loss function

  In order to compare with the IDRDPI model with unit deg^2, the same loss function was chosen here.

- training subset and validation ratio

  In order to avoid the over-fitting problem of the training process, same with the modeling LSTM, the whole training data should be split into the training subset and validation subset. Normally, the whole training data is divided into 70% for training and 30% for validation.

- Batch size

  The batch size determines the amount of the input series data. Using batch size could increase the efficient of the training process, however, too large batch size could add extra computation cost. Based on the good performance of the modeling LSTM, the batch size is also chosen to 16.

- Learning rate

  Learning rate determines the learning speed of the training process. If the learning rate is too large, it will decrease the learning efficient and decrease the performance of the

Table 10.1: Hyperparameters for the c-LSTM network

	Number of hidden layers	Number of neurons per cell	Activation functions	Optimizer	Loss function	Training-subset /Validation ratio	Batch size	Learning Rate	Epoches
LSTM	4	128	Tanh/Sigmoid	Adam	L2 Loss	70% 30%	16	0.001	100

model. However, if the learning rate is too small, the learning speed is too slow and will lead to more training time. Same with the modeling LSTM, same learning rate 0.001 is chosen here for c-LSTM.

- Epoches

  The number of iterations is defined by the number of the epoch. The training process of the c-LSTM is more challenging than the modeling LSTM model, it is therefore more epoches need to be chosen here.

Table. 10.2 shows all the detailed hyperparameters of this c-LSTM model. The compensation of the hysteresis by using the proposed c-LSTM model will be based on these hyperparameters.

10.3 Position control with the c-LSTM model

10.3.1 Training process of the c-LSTM model

The training data of the c-LSTM is also generated based on the (4.1) and the (4.2), different frequencies and pressure amplitudes are combined to excited the catheter system. The frequency f was switched between 0.2, 0.4, 0.6, 0.8, while τ was chosen as 0.02, 0.05, 0.1, 0.15, 0.2. The combination of the different frequencies and τ result in 40 groups of training data. A total of 260269 samples were in the training data set. This was done with a sampling frequency setting of 250 Hz.

All the training data is processed through the normalization in (7.9), which can normalize all data between [-1,1]. The network was implemented in an open source machine learning library named Pytorch. The training procedure was performed on an 4 GB NVIDIA® CUDA-capable GPU. The c-LSTM was adequately optimized after 100 iterations. The whole training duration took around 2.25 hours. The training and validation loss of the c-LSTM model is shown in the Figure 10.3. After 100 iterations, the training loss as well as the validation loss are reaching a plateau and do not show a increasing tendency, which means the training process is thorough and do not occur the over-fitting problem.

10.3.2 Position control performance of the c-LSTM model

In order to test the performance of the c-LSTM model, and check whether an accurate position control can be achieved by the c-LSTM model, four same desired trajectories, which have different characters with the training data, are also tested and illustrated as follows:

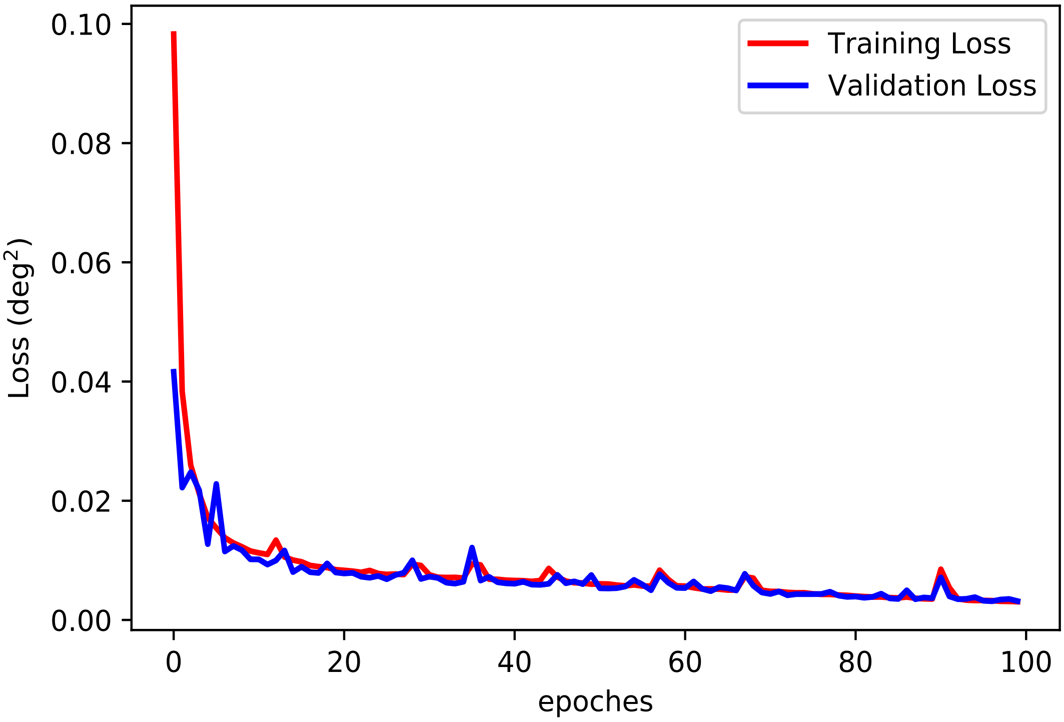

Figure 10.3: c-LSTM training and validation loss

1. Descending sinusoidal trajectory

   Same with the trajectory in the IDRDPI model, the generation of the desired trajectory is also follow the (9.22) with $f = 0.3$ Hz and $\tau = 0.05$. This test trajectory is also input into the c-LSTM model and the generated compensated pressure is the output, which shown in the Figure 10.4. A sharp increasing at each beginning of the wave can also be observed, and this compensated pressure is afterwards input into the experimental setup.

   The c-LSTM model shows a good performance in this type of the test data. During the ascending and descending period the c-LSTM model can follow the desired trajectory accurately, which leads to a low RMSE = 0.3 degrees. At the first peak, the last peak, the model shows a big spike that occurs MAE = 1.5 degrees. The test is performed five times and the standard deviation is quite small with 0.6 degrees that proofs the stability of the c-LSTM model.

2. Attenuated down-chirp sinusoidal trajectory with sifted baseline

   The performance of the c-LSTM model is further investigated on the trajectory with time-varying frequency. Based on the (9.23) with $A = 10$, $B = 8$, $f = 0.5$ Hz, and $\tau = 0.05$, the trajectory was generated. The desired trajectory was input into the c-LSTM model and the compensated pressure is shown in the Figure 10.6. Since this type of the test data has shifted baseline, after the first wave, the compensated pressure is no longer

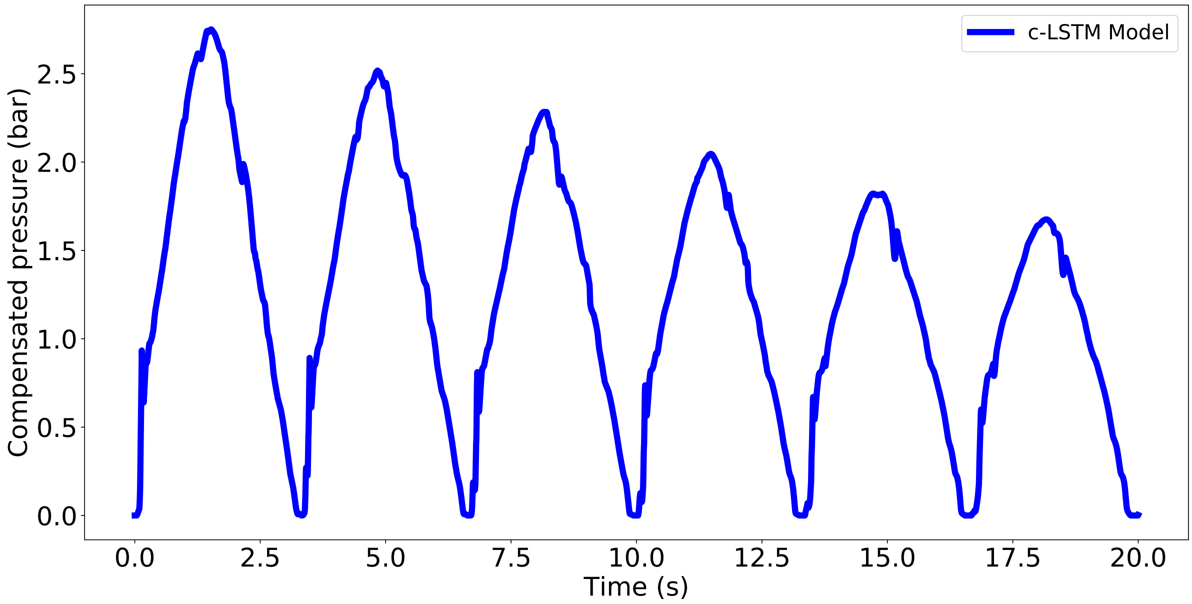

Figure 10.4: Compensated pressure of descending sinusoidal trajectory (c-LSTM)

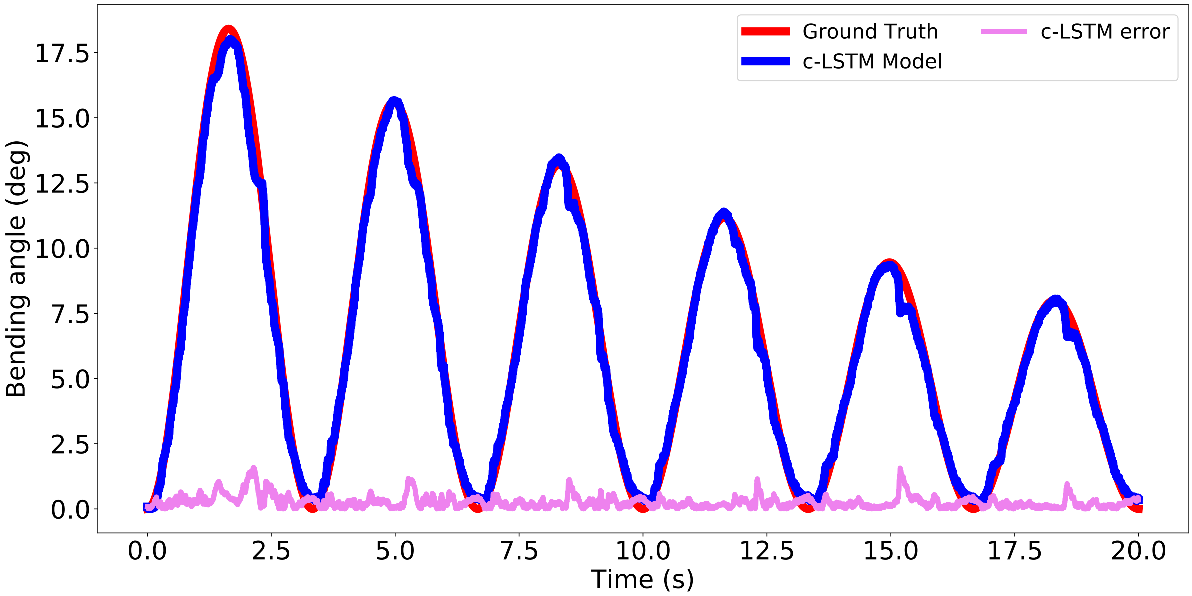

Figure 10.5: 1. Descending sinusoidal trajectory (c-LSTM)

with the sharp increment.

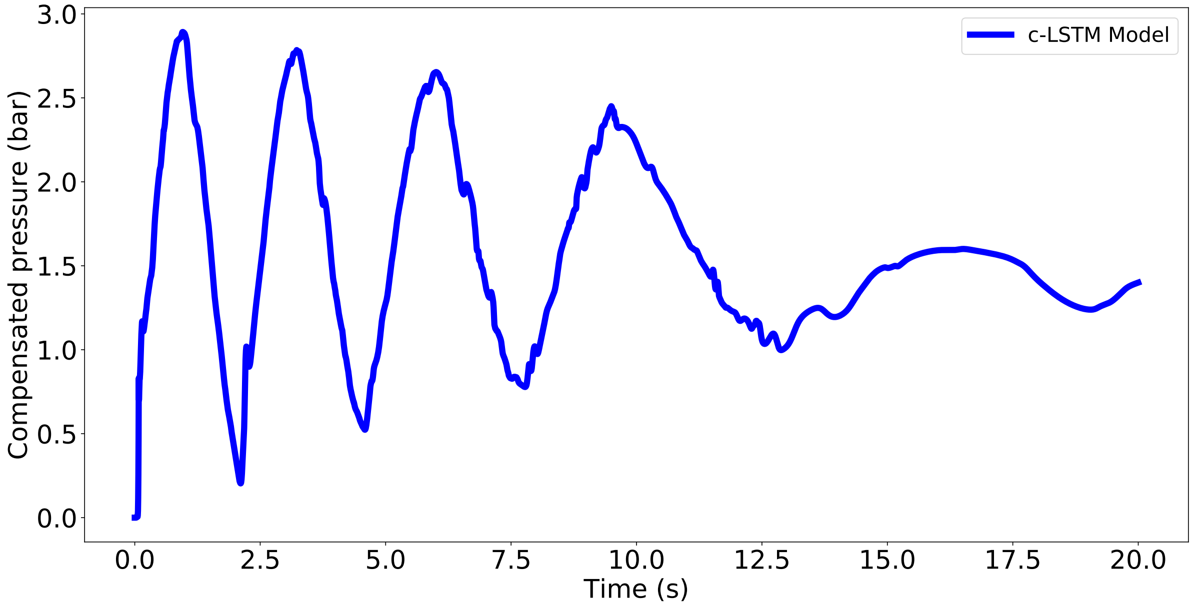

Figure 10.6: Compensated pressure of attenuated down-chirp sinusoidal trajectory with sifted baseline (c-LSTM)

At the first two waves, the c-LSTM model can follow the desired trajectory well. However, the last three waves lead to a larger RMSE = 0.6 degrees and at the fourth wave, the MAE = 1.5 degrees is occured. Based on the performance of the model in the Figure 10.7, it can be concluded that the performance of the c-LSTM decreases when the frequency is decreases.

3. Ascending up-chirp sinusoidal trajectory with zero baseline

   The same trajectory with the IDRDPI model, which is a frequency-ascending trajectory, was generated following (9.24) with A = 6, f = 0.3Hz, τ = -0.02, and c = 0.005. The desired trajectory is then input into the c-LSTM model and output the compensated pressure (see Figure 10.8).

   The compensated pressure is input into the experimental setup and the resulting bending angle is measured. The c-LSTM model successes to accurately predict the peak and compensates the hysteresis of the catheter. The error of the c-LSTM model still remains at a very low level with RMSE = 0.3 degrees and NRMSE = 0.82%.

4. Arbitrarily varying trajectory with small variations

   The arbitrary data is the most challenging test data type for the c-LSTM model, but also the most practical data. The generated trajectory is input into the model and the compensated pressure is shown in the Figure 10.10.

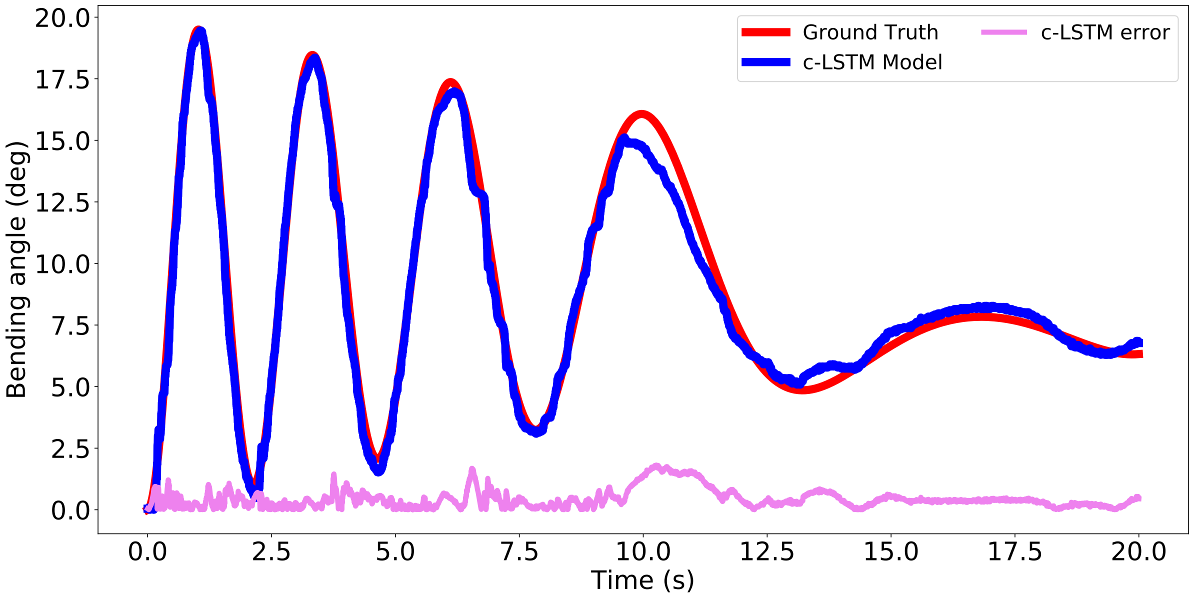

Figure 10.7: 2. Attenuated down-chirp sinusoidal trajectory with sifted baseline (c-LSTM)

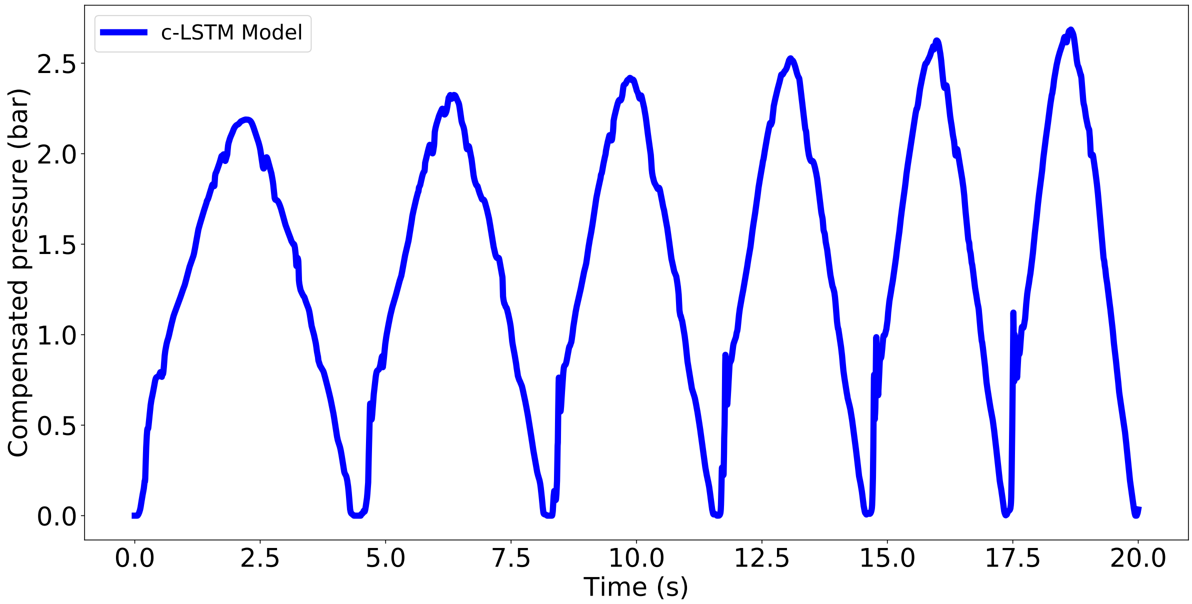

Figure 10.8: Compensated pressure of ascending up-chirp sinusoidal trajectory with zero baseline (c-LSTM)

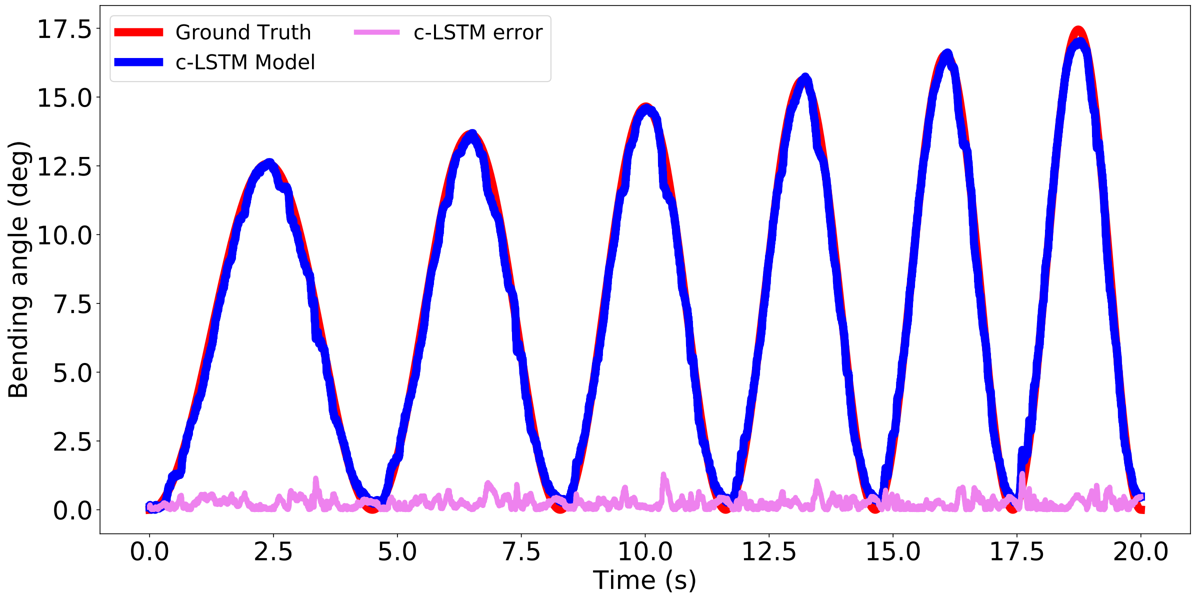

Figure 10.9: 3. Ascending up-chirp sinusoidal trajectory with zero baseline (c-LSTM)

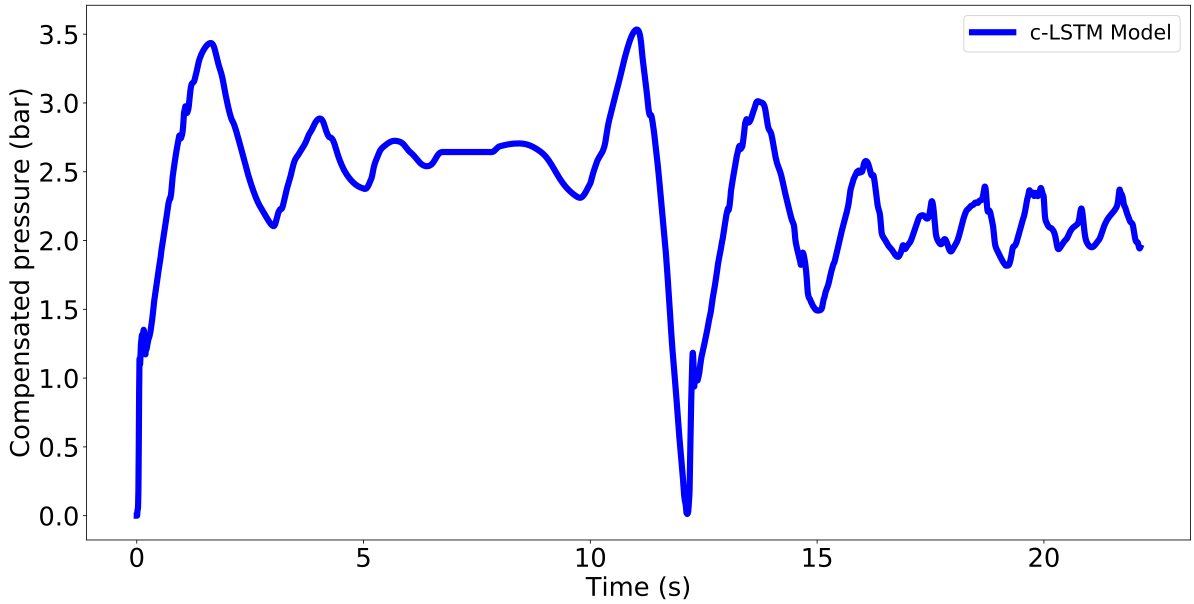

Figure 10.10: Compensated pressure of arbitrarily varying trajectory with small variations (c-LSTM)

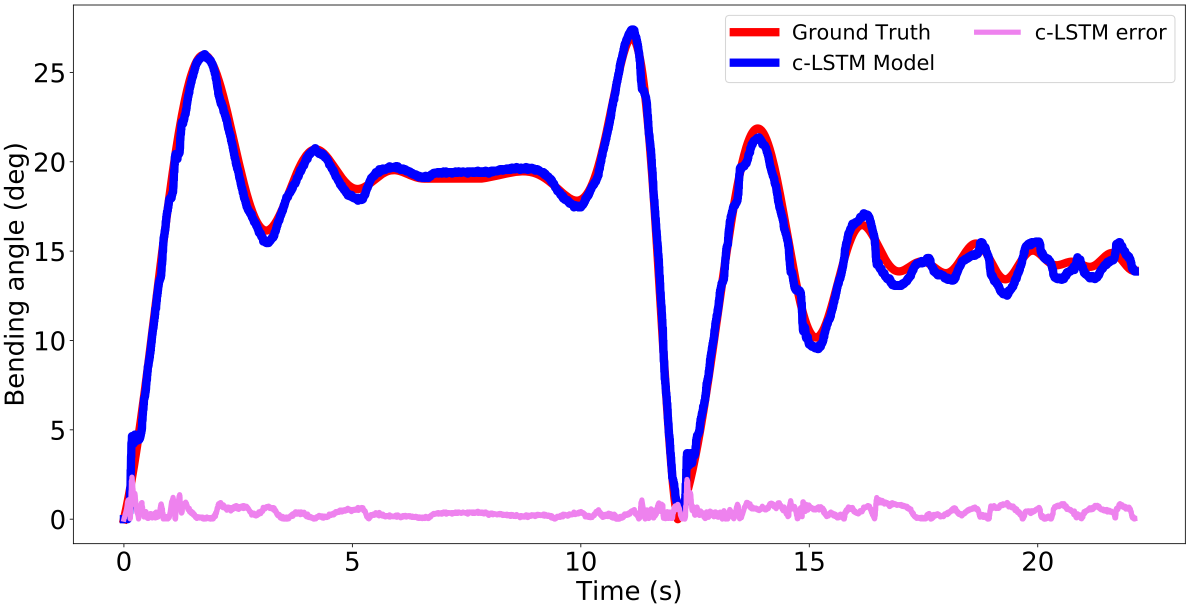

Figure 10.11: 4. Arbitrarily varying trajectory with small variations (c-LSTM)

Although the arbitrary data is challenging for the c-LSTM model, it still shows a acceptable performance. The model output follows the plateau very well and also in the small variations period. The good performance in this two challenging periods leads to a quite lower RMSE = 0.5 degrees. However, at the beginning of this type of the data, the output of the data shows a little instability that cause MAE = 2.4 degrees. Overall, the c-LSTM model has a well acceptable performance.

10.3.3 Discussion of the c-LSTM model performance

To sum up, the c-LSTM model shows a well accepted performance and all three metrics in this position control experiment are shown in the histogram (see Figure 10.12). Each est trajectory is measured five times to avoid random error, and the results of the standard deviation also indicates that the performance of the c-LSTM is stable. The smallest RMSE = 0.3 degrees, MAE = 2.4 degrees, and NRMSE = 1.44% also indicate that the error of the c-LSTM remains at a quite low level.

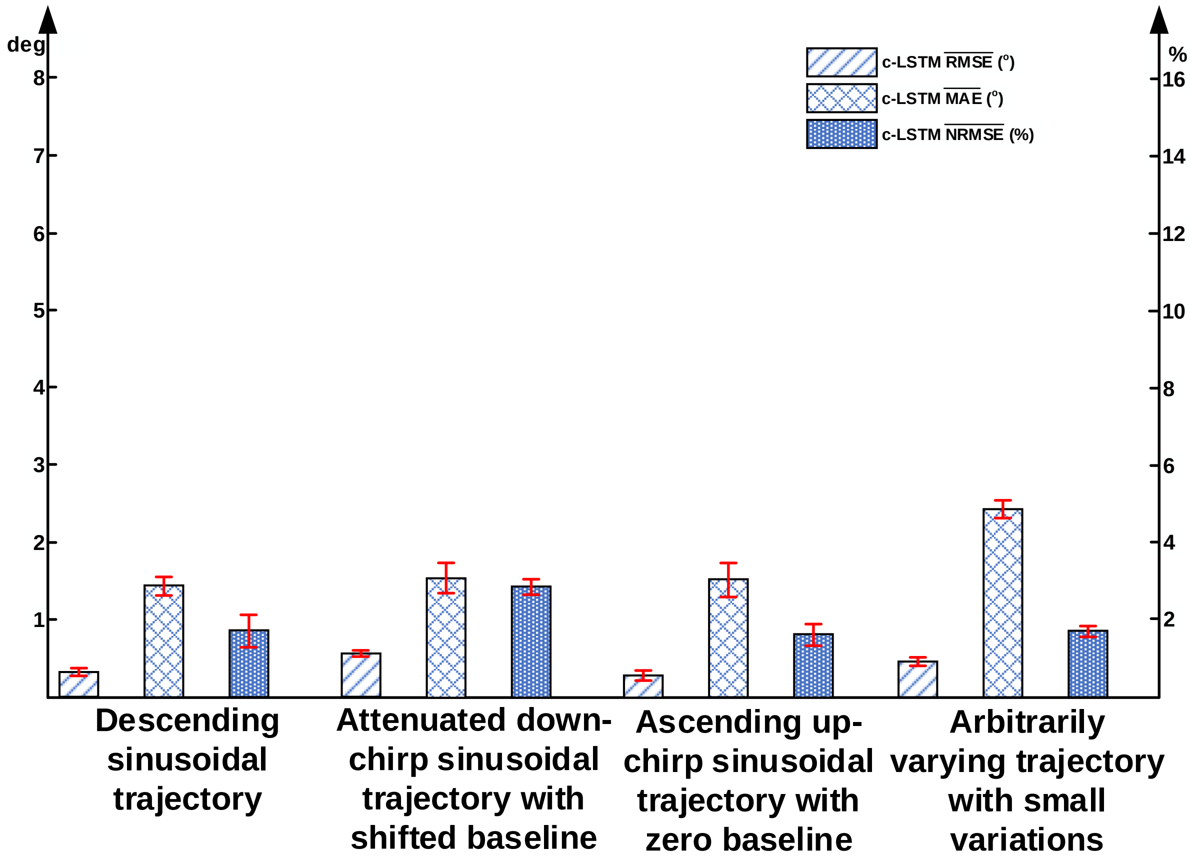

Figure 10.12: Three metrics evaluation of the position control performance (c-LSTM)

11 Comparison of control performance between proposed models

The main goal of this thesis is to compensate the hysteresis in this experimental PAM-driven robotic catheter system and furthermore achieves an accurate position control. Since the proposed DRDPI is the analytical model, and the SVR, LSTM models are learning-based type, it is therefore more meaningful and time-saving to compare two different types of the model. Thus for the control part, only the Inverse Deadband Rate-Dependent Prandtl-Ishlinskii and the Control Long Short-Term Memory models are investigated for the hysteresis compensation and the position control.

In the previous two chapters, four different types of the desired trajectories, are used to evaluate the performance of these two models. The comparison of these two models based on the four test desired trajectories is illustrated as follows:

1. Descending sinusoidal trajectory

   The first type of the test data has the same character with the training data but with different f and τ. Figure 11.1 illustrates the performance of these two models. It is noteworthy that the IDRDPI model has a clear offset with the desired trajectory. This offset could lead to a larger RMSE = 1.7 degrees than the c-LSTM with RMSE = 0.3 degrees. The MAE of c-LSTM (1.5 degrees) happens in the unloading phase immediately after the peak where there is an over-compensation, which is shown in the local zoom figure.

   Table. 11.1 shows all the detailed metrics of the performance of these two models. The green cells represent the best result, which all locate in the c-LSTM model. It is shown that the c-LSTM performs much better than the IDRDPI model at all three metrics.

2. Attenuated down-chirp sinusoidal trajectory with sifted baseline

   The performance of the IDRDPI and c-LSTM model on the trajectory with time-varying frequency is worth to investigate. Figure 11.2 shows both the performance of these two

Table 11.1: Quantitative performance of the position control in the first test data

Model	**Descending sinusoidal trajectory**		
	RMSE (deg)	**MAE (deg)**	**NRMSE**
IDRDPI	**1.67 ± 0.03**	**4.37 ± 0.06**	**9.04% ± 0.28%**
c-LSTM	**0.32 ± 0.03**	**1.45 ± 0.08**	**1.73% ± 0.34%**

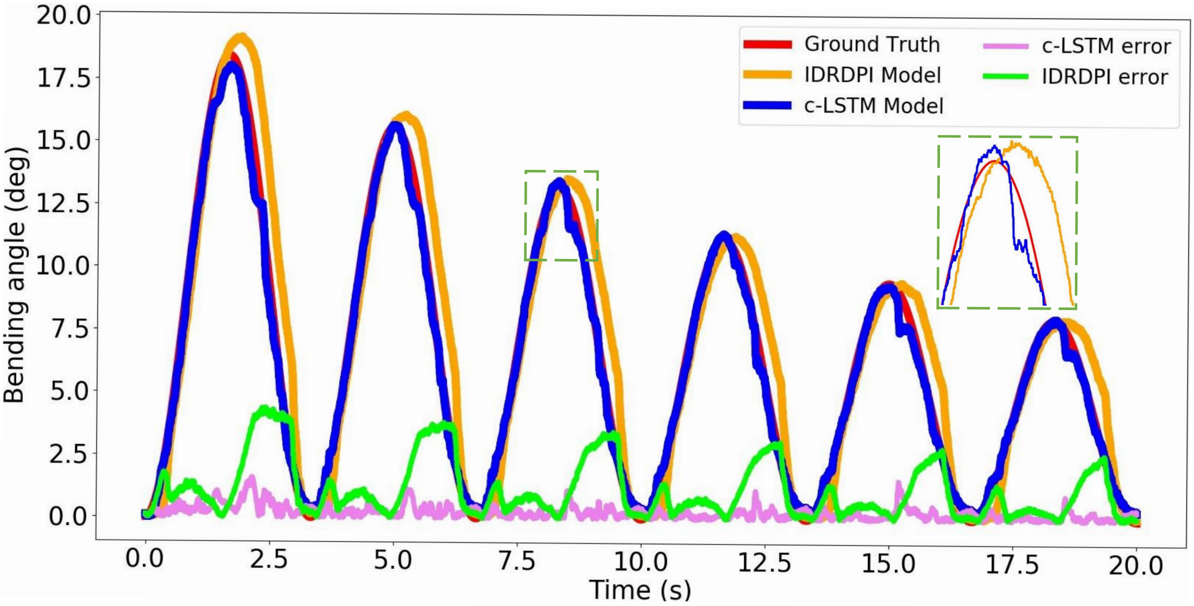

Figure 11.1: Descending sinusoidal trajectory

Table 11.2: Quantitative performance of the position control in the second test data

Model	Attenuated down-chirp sinusoidal trajectory with sifted baseline		
	RMSE (deg)	**MAE (deg)**	**NRMSE**
IDRDPI	**2.46 ± 0.03**	**4.84 ± 0.04**	**12.62% ± 0.3%**
c-LSTM	**0.56 ± 0.01**	**1.54 ± 0.15**	**2.87% ± 0.12%**

models and also the error plotting. Based on the error plotting, the error of the IDRDPI model fluctuates periodically. Compared to the c-LSTM, the IDRDPI model fails to accurately predict the peak accurately and under-compensates in the unloading phase, thus causing offsets from the set trajectory, which can cause larger RMSE.

Table. 11.2 provides all three metrics of these two models. Also all the three green cells are located in the c-LSTM model. The c-LSTM improves the performance at the RMSE by 77.24%, which is very remarkable. Also a 77.26% improvement at the NRMSE and 68.18% at MAE the c-LSTM can achieve.

3. Ascending up-chirp sinusoidal trajectory with zero baseline

   The third tested trajectory is a period-ascending trajectory, and this pattern was also not contained in the training data. The desired trajectory was generated following the (9.24) with A = 6, f = 0.3 , τ = -0.02 and c = 0.005. As with the previous two tests, the IDRDPI model performs poorly in the unloading phase (see Figure 11.3). These offsets in the unloading phase can cause large RMSE, which is shown in the Table .11.3.

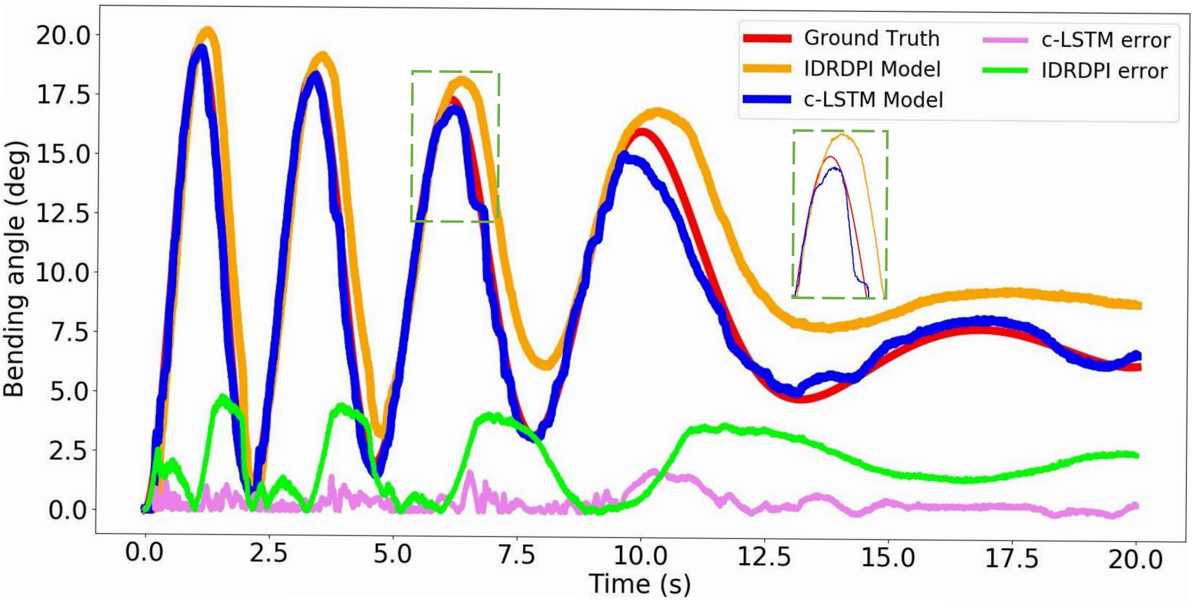

Figure 11.2: Attenuated down-chirp sinusoidal trajectory with sifted baseline

Table 11.3: Quantitative performance of the position control in the third test data

Model	**Ascending up-chirp sinusoidal trajectory with zero baseline**		
	RMSE (deg)	**MAE (deg)**	**NRMSE**
IDRDPI	**1.85 ± 0.01**	**4.41 ± 0.04**	**10.59% ± 0.12%**
c-LSTM	**0.29 ± 0.02**	**1.53 ± 0.18**	**1.64% ± 0.2%**

The amount of the three metrics shown in the table indicate that these two models perform the best among the four trajectories. The c-LSTM model also shows great advantage compared to the IDRDPI model with average 78.05%.

4. Arbitrarily varying trajectory with small variations

Arbitrarily varying trajectories are the most practical signals, thus arbitrarily varying trajectories have been tested to explore the versatility of the IDRDPI and the c-LSTM model. In addition, some high-frequency small-amplitude variations are added, which also have a lot of practical research values, for example, in order to compensate for the heart beat and avoid contacting with vessel walls, the catheter often needs to perform high-frequency small-amplitude movements.

In general, the error of the c-LSTM is consistently very small, also in regions with small fluctuations, which is shown in the Figure 11.4. The one can be noticed that the c-LSTM more better than the IDRDPI model, another one is noteworthy that the c-LSTM model can follow the small variations while the IDRDPI model is not able to follow.

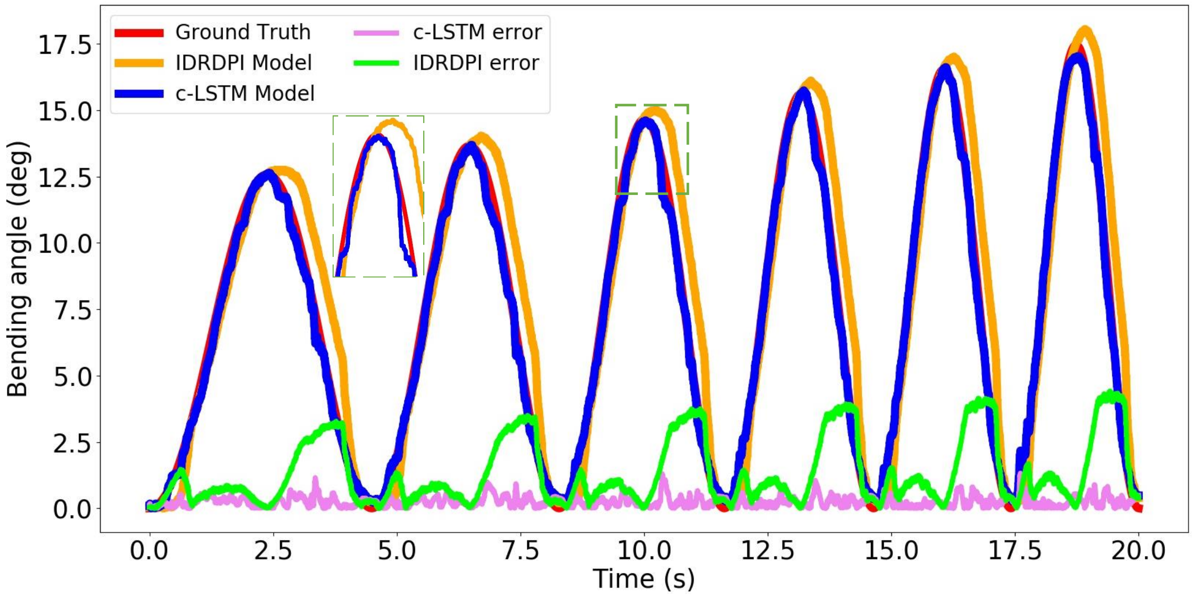

Figure 11.3: Ascending up-chirp sinusoidal trajectory with zero baseline

Table 11.4: Quantitative performance of the position control in the fourth test data

Model	**Arbitrarily varying trajectory with small variations**		
	RMSE (deg)	**MAE (deg)**	**NRMSE**
IDRDPI	**3.97 ± 0.07**	**8.52 ± 0.1**	**14.74% ± 0.6%**
c-LSTM	**0.47 ± 0.01**	**2.44 ± 0.07**	**1.73% ± 0.06%**

Table. 11.4 shows that the IDRDPI model performs the worst among all the four test data. The worst performance at the plateau and the small variations of the IDRDPI model leads to the MAE = 8.5 degrees, while the MAE of the c-LSTM is only 2.4 degrees. This type of test data proofs that the IDRDPI model is not able to achieve an accurate position control under the signal with plateaus and small variations, while the c-LSTM model can follow the desired trajectory.

The whole position control process of these two models are illustrated in the Figure 11.5. The desired trajectory is through reshape, padding, and normalization process and input into the c-LSTM and IDRDPI model. The output of these two models are compensated pressure and inout into the experimental setup. The bending angle of the catheter can be measured and compared with the desired trajectories.

Furthermore in order to compare these two models, Figure 11.6 shows the comparison of the desired trajectory and the measured trajectory. If the hysteresis is compensated well, the output should be a straight line, otherwise it will still show a hysteresis pattern. Based on the plotting in the Figure 11.6, the output of the c-LSTM is almost a line, while the output of the IDRDPI model is showing a hysteresis pattern.

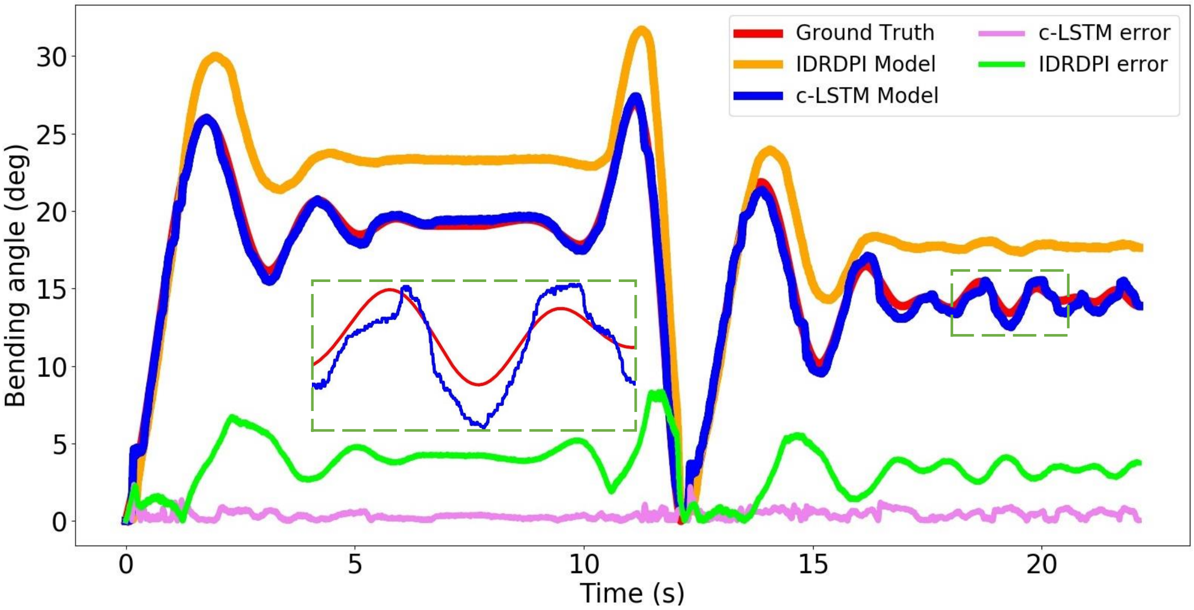

Figure 11.4: Arbitrarily varying trajectory with small variations

To sum up, among all the four trajectories the c-LSTM performs much better and more stable than the IDRDPI model. Furthermore, the training data of the c-LSTM is less than the IDRDPI model, also the training time of the c-LSTM is smaller than the IDRDPI model. In the future, with the great potential of the c-LSTM model, it could be used for other robotic catheter system that with hysteresis, which can achieve an accurate position control of the PAM-driven robotic catheter.

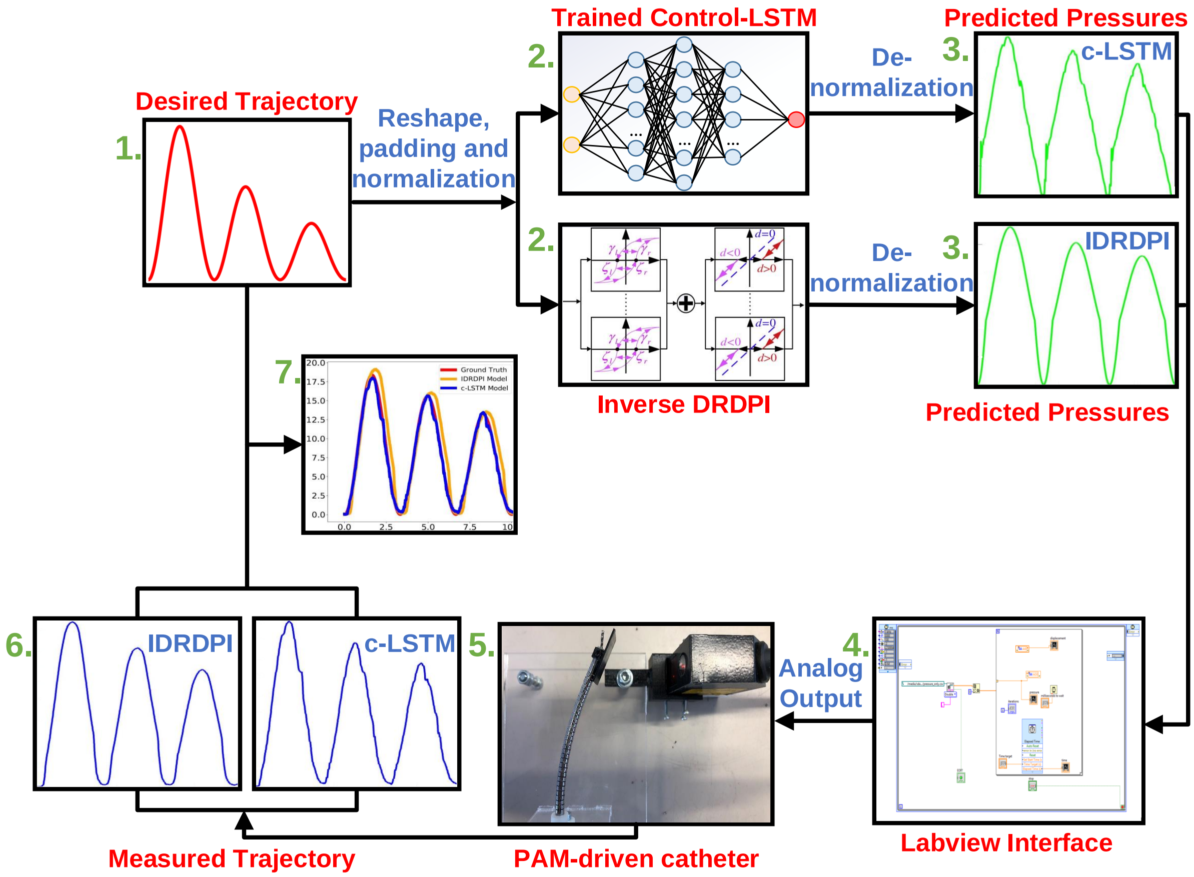

Figure 11.5: Experimental procedure to verify the performance of c-LSTM and its comparator i.e. the IDRDPI model

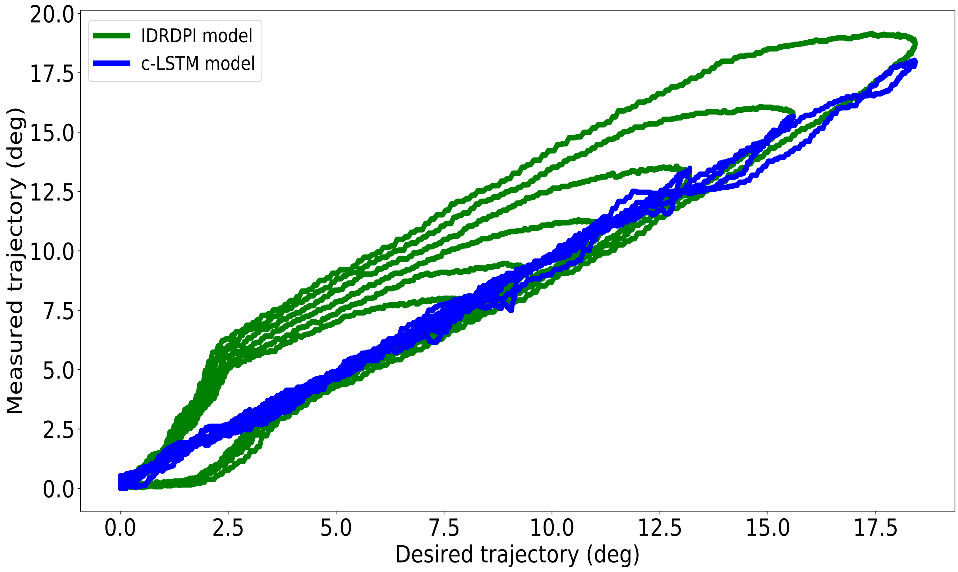

Figure 11.6: Relation between the desired trajectory and the measured trajectory

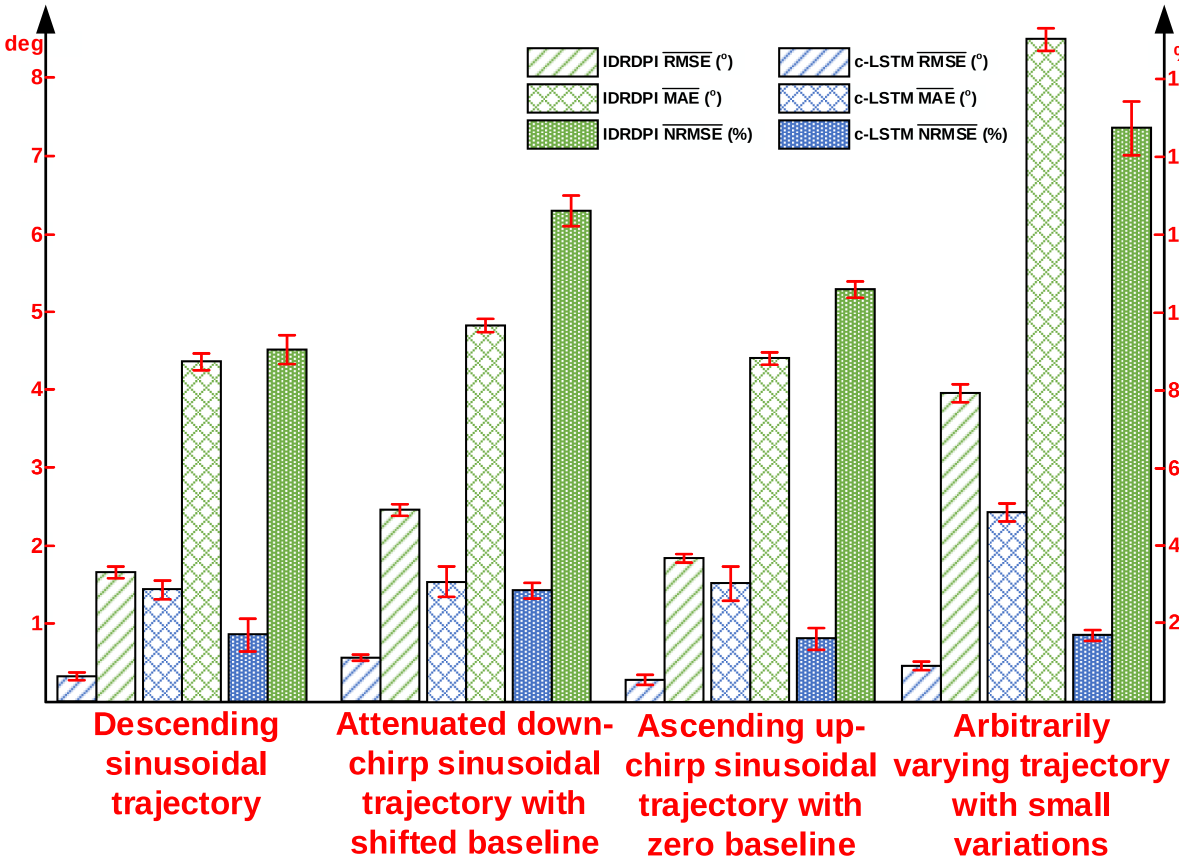

Figure 11.7: Three metrics evaluation of the position control performance

12 Discussion and conclusion

12.1 Conclusion

To address the hysteresis problem in this Pneumatic Artificial Muscle-driven robotic catheter system, an analytical Deadband Rate-Dependent Prandtl-Ishlinskii model, a machine learning Support Vector Regression model, and a deep learning Long Short-Term Memory model are first established to model the hysteresis. Descending sinusoidal pressure signals with zero and non-zero baseline under four different frequencies were used to generate multi-loop hysteresis. This data was later used as training data for the above-mentioned models. The training data collected from the setup reveals a complex pattern. The response is found to behave asymmetric, saturated, and rate-dependent hysteresis. The hysteresis is not only from the PAM itself, but also form the entire experimental setup. For the modeling process, the DRDPI and the SVR model were established for comparison to the LSTM model.

The LSTM was validated on four test signals containing diverse patterns The most challenging signal was an arbitrarily varying signal with plateaus. The RMSE, MAE, and NRMSE were found to be 0.6 degrees, 1.9 degrees, and 2.42%, respectively. The other learning-based model SVR performs less 35.87%, 3.98%, and 35.98% than the LSTM model in all three metrics. The analytical model DRDPI, performs worst with performance drops of 60.14%, 35.23%, and 60.13% compared to the LSTM model. In general, with the LSTM, the bending angle can be predicted with sub-degree precision only based on the input pressure. The good performance irrespective to the test signal shows the robustness of the LSTM even for robotic catheters that exhibit a complicated hysteresis behaviours.

Furthermore, after the modeling process of the hysteresis, the compensation and an accurate position control of the robotic catheter system is also investigated. In this thesis, based on the DRDPI and the LSTM model, a data-driven approach IDRDPI and c-LSTM model are proposed to compensate for the hysteresis. Installing the sensors at the catheter tip is challenging due to the space and sterilization issues, it is therefore meaningful to find methods to design adequate open-loop controllers and further achieve accurate control of the catheter in free space.

The analytical model IDRDPI model requires two steps, namely, first the identification of the unknown parameters and then an inversion of the forward model. In contrast, the proposed c-LSTM model can be directly trained as a controller. The tedious process of inversion can be avoided. Both the IDRDPI and c-LSTM models are tested with four different trajectories in the PAM-driven robotic catheter system. These trajectories are generated to simulate some specific clinical scenarios, such as signals with plateaus, with small-amplitude variations to compensate for the heart beating. The experimental results indicate that the c-LSTM model could achieve a sub-degree position control precision and outperformed the

IDRDPI model at least 75% among over all four test trajectories. Based on the results from the experiments, the precision of the controller also meets the existing clinical requirements, which ranges from 1-3 mm in the cardiovascular interventions [68].

To sum up, both the IDRDPI model and the c-LSTM model could be used as a controller for the PAM-driven robotic catheter system. However, the precision of the controller based on the c-LSTM model is better than from the analytical IDRDPI. Moreover, the training time of the c-LSTM (2.25 hours) is lower than the IDRDPI (3.5 hours). Considering promising advancements for efficient deep learning and the GPU development, the training time of the c-LSTM could be further reduced and can be also foreseen to used as a controller for other driven robotic catheter systems, which exist complex hysteresis.

12.2 Future work

Due to the time limit many experiments and tasks have been left for future work. In this work, the position control of the catheter is in the free space. With the IDRDPI model and the c-LSTM model, the catheter tip can follow the desired trajectory. A next step would consist of exploring the potential of these two models when the catheter is in contact with the environment. This investigation will be more practical because during the intervention, the contact with the blood vessels cannot be fully avoided.

Moreover, the current PAM-driven robotic catheter is a one-DoF system. The bending of the catheter remains in one plane and also all the test trajectories have been so. Thus in the future work, a new experimental setup with multi-DoFs is worth to investigate. In this thesis, the potential of the c-LSTM in one direction has been proved, the combination of multi-DoF of the c-LSTM model will also be validated in the future work.

List of Figures

1.1 Coronary Artery Disease (CAD) 1
1.2 Percutaneous Coronary Intervention (PCI) Procedure and Recanalization of the occlusion 2
1.3 X-Ray imaging of Recanalization 3

2.1 Search matrix of the conducted literature review in this thesis 6
2.2 Hysteresis with deadband 8
2.3 The different implementations of Prandtl-Ishlinskii models 9

3.1 Five parts of PAM 11
3.2 Bending segment 12
3.3 Bending angle measurement 13
3.4 Transformation formula 14
3.5 Experimental setup 15
3.6 PAM-driven setup in the lab 16

4.1 LabVIEW® Front panel 18
4.2 Block diagram of zero data 19
4.3 Asymmetric deadband hysteresis pattern (zero data) 20
4.4 Asymmetric deadband hysteresis pattern (non-zero data) 21

5.1 The effect of dynamic threshold function on the output 23
5.2 Threshold of deadband operators 23
5.3 Genetic Algorithm process 28
5.4 Training data collection 29
5.5 DRDPI training loss 29
5.6 Descending sine wave differing from training data (DRDPI) 31
5.7 Attenuated down-chirp sine wave with shifted baseline (DRDPI) 32
5.8 Ascending up-chirp sine wave with zero baseline (DRDPI) 32
5.9 Arbitrarily varying signal with plateaus (DRDPI) 33
5.10 Three metrics evaluation of the modeling performance (DRDPI) 34

6.1 Support Vector Machine in 2D 35
6.2 Support Vector Regression 36
6.3 SVR training and validation loss 38
6.4 Descending sine wave differing from training data (SVR) 39
6.5 Attenuated down-chirp sine wave with shifted baseline (SVR) 40

6.6 Ascending up-chirp sine wave with zero baseline (SVR) 41
6.7 Arbitrarily varying signal with plateaus (SVR) 41

7.1 Long Short-Term Memory cell 43
7.2 Sigmoid function 44
7.3 Tanh function 45
7.4 Input data preprocessing 46
7.5 Training loss of the LSTM 49
7.6 LSTM structure 49
7.7 Descending sine wave differing from training data (LSTM) 50
7.8 Attenuated down-chirp sine wave with shifted baseline (LSTM) 51
7.9 Ascending up-chirp sine wave with zero baseline (LSTM) 52
7.10 Arbitrarily varying signal with plateaus (LSTM) 52
7.11 Three metrics evaluation of the modeling performance (LSTM) 53

8.1 Classification of proposed models 54
8.2 Descending sine wave differing from training data 55
8.3 Attenuated down-chirp sine wave with shifted baseline 56
8.4 Ascending up-chirp sine wave with zero baseline 57
8.5 Arbitrarily varying signal with plateaus 58

9.1 Output of the direct IDRDPI 62
9.2 Optimization process of the IDRDPI 66
9.3 Compensated pressure of the descending sinusoidal trajectory (IDRDPI) . . . 67
9.4 Descending sinusoidal trajectory (IDRDPI) 67
9.5 Compensated pressure of the attenuated down-chirp sinusoidal trajectory with shifted baseline (IDRDPI) 68
9.6 Attenuated down-chirp sinusoidal trajectory with shifted baseline (IDRDPI) . 68
9.7 Compensated pressure of the ascending up-chirp sinusoidal trajectory with zero baseline (IDRDPI) 69
9.8 Ascending up-chirp sinusoidal trajectory with zero baseline (IDRDPI) 70
9.9 Compensated pressure of the arbitrarily varying trajectory with small variations (IDRDPI) 71
9.10 Arbitrarily varying trajectory with small variations (IDRDPI) 71
9.11 Three metrics evaluation of the position control performance (IDRDPI) 72

10.1 c-LSTM input data preprocessing 73
10.2 c-LSTM structure 74
10.3 c-LSTM training and validation loss 77
10.4 Compensated pressure of descending sinusoidal trajectory (c-LSTM) 78
10.5 Descending sinusoidal trajectory (c-LSTM) 78
10.6 Compensated pressure of attenuated down-chirp sinusoidal trajectory with sifted baseline (c-LSTM) 79
10.7 Attenuated down-chirp sinusoidal trajectory with sifted baseline (c-LSTM) . . 80

10.8 Compensated pressure of ascending up-chirp sinusoidal trajectory with zero baseline (c-LSTM) 80
10.9 Ascending up-chirp sinusoidal trajectory with zero baseline (c-LSTM) 81
10.10 Compensated pressure of arbitrarily varying trajectory with small variations (c-LSTM) 81
10.11 Arbitrarily varying trajectory with small variations (c-LSTM) 82
10.12 Three metrics evaluation of the position control performance (c-LSTM) 83

11.1 Descending sinusoidal trajectory 85
11.2 Attenuated down-chirp sinusoidal trajectory with sifted baseline 86
11.3 Ascending up-chirp sinusoidal trajectory with zero baseline 87
11.4 Arbitrarily varying trajectory with small variations 88
11.5 Experimental procedure to verify the performance of c-LSTM and its comparator i.e. the IDRDPI model 89
11.6 Relation between the desired trajectory and the measured trajectory 90
11.7 Three metrics evaluation of the position control performance 90

List of Tables

2.1 Methods to minimize hysteresis in flexible surgical robots from previous literature 7

5.1 Unknown parameters of the proposed model 25
5.2 Identified parameters of the model 30

6.1 Hyperparameters of the SVR model 37
6.2 Three metrics of the first test data in SVR model 39
6.3 Three metrics of the second test data in SVR model 39
6.4 Three metrics of the third test data in SVR model 40
6.5 Three metrics of the fourth test data in SVR model 42

7.1 Hyperparameters for the LSTM network 48

8.1 Quantitative performance of the proposed models in the first test data 55
8.2 Quantitative performance of the proposed models in the second test data . . . 57
8.3 Quantitative performance of the proposed models in the third test data 58
8.4 Quantitative performance of the proposed models in the fourth test data . . . 59
8.5 optimization results of three models 59

9.1 Unknown parameters of the GPI model 64
9.2 Identified parameters of the GPI model under 0.2 Hz 64
9.3 Identified parameters of the GPI model under 0.4 Hz 64
9.4 Identified parameters of the GPI model under 0.6 Hz 65
9.5 Identified parameters of the GPI model under 0.8 Hz 65
9.6 Identified parameters of the IDRDPI model 65

10.1 Hyperparameters for the c-LSTM network 76

11.1 Quantitative performance of the position control in the first test data 84
11.2 Quantitative performance of the position control in the second test data 85
11.3 Quantitative performance of the position control in the third test data 86
11.4 Quantitative performance of the position control in the fourth test data 87

Acronyms

API Application Programming Interface. 5, 6

c-LSTM Control Long Short-Term Memory. iv, 3, 73–77, 79, 82, 84–88, 91, 92

CAD Coronary Artery Disease. iv, 1, 5, 93

CD Cable-Driven. 6, 7

DRDPI Deadband Rate-Dependent Prandtl-Ishlinskii. iv, 3, 22, 24–27, 30–34, 37, 38, 42, 48, 50, 51, 53–56, 58–62, 64, 65, 84, 91, 93

EM Electromagnetic. 7, 10

GA Genetic Algorithm. 26–28, 63, 64, 93

GPI Generalized Prandtl-Ishlinskii. 62–64

IDRDPI Inverse Deadband Rate-Dependent Prandtl-Ishlinskii. iv, 3, 60–62, 64–66, 69, 70, 75, 77, 79, 84–88, 91, 92, 94

LS-SVM least square support vector machine. 8, 10

LSTM Long Short-Term Memory. iv, 3, 43, 45–56, 58, 59, 73–76, 84, 91, 94

MAE Maximum Absolute Error. 30, 31, 33, 37, 39, 40, 42, 50, 51, 53–56, 58, 59, 66, 69, 70, 77, 79, 82, 84, 85, 87, 91

MIS Minimally Invasive Surgery. iv, 2, 5

MSE Mean Square Error. 25–27, 38

NRMSE Normalized Root Mean Square Error. 30, 31, 33, 34, 42, 51, 53, 54, 56, 58, 59, 69, 70, 79, 82, 85, 91

PAM Pneumatic Artificial Muscle. iv, 3, 6, 11, 12, 22, 24, 27, 34, 42, 44, 53, 54, 60, 62, 66, 84, 88, 91–93

PCI Percutaneous Coronary Intervention. iv, 1–3, 5, 6, 17, 93

PI Prandtl-Ishlinskii. 8–10, 22, 93

RBF Radial Basis Function. 37

RDPI Rate-Dependent Prandtl-Ishlinskii. 22–24

RMSE Root Mean Square Error. 30, 31, 34, 39, 40, 42, 50, 51, 53–56, 58, 59, 66, 69, 70, 77, 79, 82, 84, 85, 91

SGD Stochastic Gradient Descent. 47

SMA Shape Memory Alloy. iv

SVM Support Vector Machine. 35, 36

SVR Support Vector Regression. iv, 3, 35–42, 46, 48, 50, 51, 54–56, 58, 59, 84, 91, 93, 94

TSM Tendon-Sheath Mechanism. 6

Bibliography

[1] E. Kandaswamy and L. Zuo. *Recent advances in treatment of coronary artery disease: Role of science and technology*. 2018. DOI: `10.3390/ijms19020424`.

[2] "Silcent Myocardial Ishemia: Diagnosis, Treatment, and Prognosis". In: (). URL: `https://forum.facmedicine.com/threads/silent-myocardial-ischemia-diagnosis-treatment-and-prognosis.37616/`.

[3] S. Mendis, P. Puska, and B. Norrving. "Global atlas on cardiovascular disease prevention and control". In: *World Health Organization* (2011).

[4] A. Shah. *Chronic Total Occlusion Coronary Intervention: In Search of a Definitive Benefit*. 2018. DOI: `10.14797/mdcj-14-1-50`.

[5] P. F. Ludman. *Percutaneous coronary intervention*. 2018. DOI: `10.1016/j.mpmed.2018.06.007`.

[6] *Global Percutaneous Coronary Intervention Market: Information by Product Type, by End User and by Region - Forecast till 2023*. URL: `https://www.medgadget.com/2018/08/advancements-in-percutaneous-coronary-intervention-market-for-prevention-and-treatment-with-evolving-cagr-of-10-8-in-the-healthcare-industry-through-2017-2023.html`.

[7] A. Zaghloul, C. Iorgoveanu, K. Balakumaran, D. V. Balanescu, and T. Donisan. "Limitations of Coronary Computed Tomography Angiography in Predicting Acute Coronary Syndrome in a Low to Intermediate-risk Patient with Chest Pain". In: *Cureus* 10.5 (2018). ISSN: 2168-8184. DOI: `10.7759/cureus.2649`.

[8] K. Yoshimitsu, T. Kato, S. E. Song, and N. Hata. "A novel four-wire-driven robotic catheter for radio-frequency ablation treatment". In: *International Journal of Computer Assisted Radiology and Surgery* (2014). ISSN: 18616429. DOI: `10.1007/s11548-014-0982-3`.

[9] D. Wu, G. Li, N. Patel, J. Yan, R. Monfaredi, K. Cleary, and I. Iordachita. "Remotely Actuated Needle Driving Device for MRI-Guided Percutaneous Interventions". In: *2019 International Symposium on Medical Robotics, ISMR 2019*. 2019. ISBN: 9781538678251. DOI: `10.1109/ISMR.2019.8710176`.

[10] J. Legrand, K. Niu, Z. Qian, K. Denis, V. Vander Poorten, L. Van Gerven, and E. Vander Poorten. "A Method Based on 3D Shape Analysis Towards the Design of Flexible Instruments for Endoscopic Maxillary Sinus Surgery". In: *Annals of Biomedical Engineering* (2021). ISSN: 15739686. DOI: `10.1007/s10439-020-02700-z`.

[11] P. Qi, C. Qiu, A. Mehndiratta, I. M. Chen, and H. Yu. "Large-deflection statics analysis of active cardiac catheters through co-rotational modelling". In: *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*. 2016. ISBN: 9781457702204. DOI: `10.1109/EMBC.2016.7591150`.

[12] P. Qi, H. Liu, L. Seneviratne, and K. Althoefer. "Towards kinematic modeling of a multi-DOF tendon driven robotic catheter". In: *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2014*. 2014. ISBN: 9781424479290. DOI: `10.1109/EMBC.2014.6944256`.

[13] M. Khoshnam and R. V. Patel. "Robotics-assisted control of steerable ablation catheters based on the analysis of tendon-sheath transmission mechanisms". In: *IEEE/ASME Transactions on Mechatronics* (2017). ISSN: 10834435. DOI: `10.1109/TMECH.2017.2688320`.

[14] Y. Chen, L. Wang, K. Galloway, I. Godage, N. Simaan, and E. Barth. *Modal-based kinematics and contact detection of soft robots*. 2019. DOI: `10.1089/soro.2019.0095`.

[15] I. S. Godage, Y. Chen, and I. D. Walker. "Dynamic control of pneumatic muscle actuators". In: *arXiv* (2018).

[16] M. De Volder, A. J. Moers, and D. Reynaerts. "Fabrication and control of miniature McKibben actuators". In: *Sensors and Actuators, A: Physical* (2011). ISSN: 09244247. DOI: `10.1016/j.sna.2011.01.002`.

[17] K. Niu, J. Legrand, L. V. Gerven, and E. V. Poorten. "Statistical Shape Modelling of the Human Nasal Cavity and Maxillary Sinus for Minimally Invasive Surgery". In: *10 th Conference on New Technologies for Computer and Robot Assisted Surgery* (2020), pp. 74–75.

[18] B. Bardou, F. Nageotte, P. Zanne, and M. De Mathelin. "Improvements in the control of a flexible endoscopic system". In: *Proceedings - IEEE International Conference on Robotics and Automation* (2012), pp. 3725–3732. ISSN: 10504729. DOI: `10.1109/ICRA.2012.6225050`.

[19] R. J. Roesthuis, S. Janssen, and S. Misra. "On using an array of fiber Bragg grating sensors for closed-loop control of flexible minimally invasive surgical instruments". In: *IEEE International Conference on Intelligent Robots and Systems*. 2013. ISBN: 9781467363587. DOI: `10.1109/IROS.2013.6696715`.

[20] R. Reilink, S. Stramigioli, and S. Misra. "3D position estimation of flexible instruments: Marker-less and marker-based methods". In: *International Journal of Computer Assisted Radiology and Surgery* (2013). ISSN: 18616429. DOI: `10.1007/s11548-012-0795-1`.

[21] P. Cabras, F. Nageotte, P. Zanne, and C. Doignon. "An adaptive and fully automatic method for estimating the 3D position of bendable instruments using endoscopic images". In: *International Journal of Medical Robotics and Computer Assisted Surgery* (2017). ISSN: 1478596X. DOI: `10.1002/rcs.1812`.

[22] D. Baek, J. H. Seo, J. Kim, and D. S. Kwon. "Hysteresis Compensator with Learning-Based Hybrid Joint Angle Estimation for Flexible Surgery Robots". In: *IEEE Robotics and Automation Letters* (2020). ISSN: 23773766. DOI: `10.1109/LRA.2020.2972821`.

[23] T. N. Do, T. Tjahjowidodo, M. W. Lau, T. Yamamoto, and S. J. Phee. “Hysteresis modeling and position control of tendon-sheath mechanism in flexible endoscopic systems”. In: *Mechatronics* (2014). ISSN: 09574158. DOI: `10.1016/j.mechatronics.2013.11.003`.

[24] X. Wang, D. Bie, J. Han, and Y. Fang. “Active Modeling and Compensation for the Hysteresis of a Robotic Flexible Ureteroscopy”. In: *IEEE Access* (2020). ISSN: 21693536. DOI: `10.1109/ACCESS.2020.2984424`.

[25] Z. Sun, Z. Wang, and S. J. Phee. “Elongation modeling and compensation for the flexible tendon - Sheath system”. In: *IEEE/ASME Transactions on Mechatronics* (2014). ISSN: 10834435. DOI: `10.1109/TMECH.2013.2278613`.

[26] O. M. Omisore, S. P. Han, L. X. Ren, G. S. Wang, F. L. Ou, H. Li, and L. Wang. “Towards Characterization and Adaptive Compensation of Backlash in a Novel Robotic Catheter System for Cardiovascular Interventions”. In: *IEEE Transactions on Biomedical Circuits and Systems* 12.4 (2018), pp. 824–838. ISSN: 19324545. DOI: `10.1109/TBCAS.2018.2825359`.

[27] J. Legrand, D. Dirckx, M. Durt, M. Ourak, J. Deprest, S. Ourselin, J. Qian, T. Vercauteren, and E. V. Poorten. “Active handheld flexible fetoscope-design and control based on a modified generalized prandtl-ishlinski model”. In: *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM* 2020-July (2020), pp. 367–374. DOI: `10.1109/AIM43001.2020.9158868`.

[28] W. Xu, J. Chen, H. Y. Lau, and H. Ren. “Data-driven methods towards learning the highly nonlinear inverse kinematics of tendon-driven surgical manipulators”. In: *International Journal of Medical Robotics and Computer Assisted Surgery* (2017). ISSN: 1478596X. DOI: `10.1002/rcs.1774`.

[29] R. A. Porto, F. Nageotte, P. Zanne, and M. De Mathelin. “Position control of medical cable-driven flexible instruments by combining machine learning and kinematic analysis”. In: *Proceedings - IEEE International Conference on Robotics and Automation*. 2019. ISBN: 9781538660263. DOI: `10.1109/ICRA.2019.8793692`.

[30] O. M. Omisore, S. Han, T. Zhou, Y. Al-Handarish, W. Du, K. Ivanov, and L. Wang. “Learning-based Parameter Estimation for Hysteresis Modeling in Robotic Catheterization”. In: *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS* (2019), pp. 5399–5402. ISSN: 1557170X. DOI: `10.1109/EMBC.2019.8856718`.

[31] Z. Jiang, M. Grimm, M. Zhou, J. Esteban, W. Simson, G. Zahnd, and N. Navab. “Automatic Normal Positioning of Robotic Ultrasound Probe Based only on Confidence Map Optimization and Force Measurement”. In: *IEEE Robotics and Automation Letters* (2020). ISSN: 23773766. DOI: `10.1109/LRA.2020.2967682`.

[32] K. Niu, J. Homminga, V. Sluiter, A. Sprengers, and N. Verdonschot. “Measuring relative positions and orientations of the tibia with respect to the femur using one-channel 3D-tracked A-mode ultrasound tracking system: A cadaveric study”. In: *Medical Engineering and Physics* (2018). ISSN: 18734030. DOI: `10.1016/j.medengphy.2018.04.015`.

[33] K. Niu, V. Sluiter, A. Sprengers, J. Homminga, and N. Verdonschot. "A Novel Tibiafemoral Kinematics Measurement System Based on Multi-Channel A-Mode Ultrasound System". In: 2018. DOI: `10.29007/p65f`.

[34] V. Hassani, T. Tjahjowidodo, and T. N. Do. *A survey on hysteresis modeling, identification and control*. 2014. DOI: `10.1016/j.ymssp.2014.04.012`.

[35] H. Hu and R. Ben Mrad. "On the classical Preisach model for hysteresis in piezoceramic actuators". In: *Mechatronics* 13.2 (2003), pp. 85–94. ISSN: 09574158. DOI: `10.1016/S0957-4158(01)00043-5`.

[36] J. L. Ha, R. F. Fung, and C. S. Yang. "Hysteresis identification and dynamic responses of the impact drive mechanism". In: *Journal of Sound and Vibration* (2005). ISSN: 0022460X. DOI: `10.1016/j.jsv.2004.05.032`.

[37] D. Song and C. J. Li. "Modeling of piezo actuator's nonlinear and frequency dependent dynamics". In: *Mechatronics* (1999). ISSN: 09574158. DOI: `10.1016/S0957-4158(99)00005-7`.

[38] Z. Wang, Z. Zhang, J. Mao, and K. Zhou. "A Hammerstein-based model for rate-dependent hysteresis in piezoelectric actuator". In: *Proceedings of the 2012 24th Chinese Control and Decision Conference, CCDC 2012* (2012), pp. 1391–1396. DOI: `10.1109/CCDC.2012.6244223`.

[39] M. Rakotondrabe. "Bouc-Wen modeling and inverse multiplicative structure to compensate hysteresis nonlinearity in piezoelectric actuators". In: *IEEE Transactions on Automation Science and Engineering* (2011). ISSN: 15455955. DOI: `10.1109/TASE.2010.2081979`.

[40] M. Al Janaideh and O. Aljanaideh. "Further results on open-loop compensation of rate-dependent hysteresis in a magnetostrictive actuator with the Prandtl-Ishlinskii model". In: *Mechanical Systems and Signal Processing* 104 (2018), pp. 835–850. ISSN: 10961216. DOI: `10.1016/j.ymssp.2017.09.004`.

[41] M. Al Janaideh, M. Rakotondrabe, and O. Aljanaideh. "Further Results on Hysteresis Compensation of Smart Micropositioning Systems with the Inverse Prandtl-Ishlinskii Compensator". In: *IEEE Transactions on Control Systems Technology* (2016). ISSN: 10636536. DOI: `10.1109/TCST.2015.2446959`.

[42] M. Al Janaideh, S. Rakheja, and C. Y. Su. "An analytical generalized Prandtl-Ishlinskii model inversion for hysteresis compensation in micropositioning control". In: *IEEE/ASME Transactions on Mechatronics* 16.4 (2011), pp. 734–744. ISSN: 10834435. DOI: `10.1109/TMECH.2010.2052366`.

[43] M. Al Janaideh and P. Krejč. "An inversion formula for a PrandtlIshlinskii operator with time dependent thresholds". In: *Physica B: Condensed Matter* 406.8 (2011), pp. 1528–1532. ISSN: 09214526. DOI: `10.1016/j.physb.2011.01.062`.

[44] O. Aljanaideh, M. A. Janaideh, S. Rakheja, and C. Y. Su. "Compensation of rate-dependent hysteresis nonlinearities in a magnetostrictive actuator using an inverse Prandtl-Ishlinskii model". In: *Smart Materials and Structures* 22.2 (2013). ISSN: 09641726. DOI: `10.1088/0964-1726/22/2/025027`.

[45] R. Li, K. Niu, D. Wu, and E. Vander Poorten. "A Framework of Real-time Freehand Ultrasound Reconstruction based on Deep Learning for Spine Surgery". In: *10th Conference on New Technologies for Computer and Robot Assisted Surgery(CRAS)* (), pp. 48–49.

[46] M. Zhou, X. Wang, J. Weiss, A. Eslami, K. Huang, M. Maier, C. P. Lohmann, N. Navab, A. Knoll, and M. A. Nasseri. "Needle localization for robot-assisted subretinal injection based on deep learning". In: *Proceedings - IEEE International Conference on Robotics and Automation*. 2019. ISBN: 9781538660263. DOI: `10.1109/ICRA.2019.8793756`.

[47] H. Su, Y. Hu, H. R. Karimi, A. Knoll, G. Ferrigno, and E. De Momi. "Improved recurrent neural network-based manipulator control with remote center of motion constraints: Experimental results". In: *Neural Networks* (2020). ISSN: 18792782. DOI: `10.1016/j.neunet.2020.07.033`.

[48] M. A. Ahmad, M. Ourak, C. Gruijthuijsen, J. Deprest, T. Vercauteren, and E. Vander Poorten. "Deep learning-based monocular placental pose estimation: towards collaborative robotics in fetoscopy". In: *International Journal of Computer Assisted Radiology and Surgery* (2020). ISSN: 18616429. DOI: `10.1007/s11548-020-02166-3`.

[49] R. J. Webster and B. A. Jones. "Design and kinematic modeling of constant curvature continuum robots: A review". In: *International Journal of Robotics Research* 29.13 (2010), pp. 1661–1683. ISSN: 02783649. DOI: `10.1177/0278364910368147`.

[50] S. Shakiba, M. Ourak, E. V. Poorten, M. Ayati, and A. Yousefi-Koma. "Modeling and compensation of asymmetric rate-dependent hysteresis of a miniature pneumatic artificial muscle-based catheter". In: *Mechanical Systems and Signal Processing* 154 (2021), p. 107532. ISSN: 10961216. DOI: `10.1016/j.ymssp.2020.107532`. URL: `https://doi.org/10.1016/j.ymssp.2020.107532`.

[51] D. Wu, Y. Zhang, M. Ourak, K. Niu, J. Dankelman, and E. B. Vander Poorten. "Hysteresis Modeling of Robotic Catheters based on Long Short-Term Memory Network for Improved Environment Reconstruction". In: *IEEE Robotics and Automation Letters* 6.2 (2021), pp. 2106–2113. ISSN: 23773766. DOI: `10.1109/LRA.2021.3061069`.

[52] S. Shakiba, A. Yousefi-Koma, M. Jokar, M. R. Zakerzadeh, and H. Basaeri. "Modeling and characterization of the shape memory alloy–based morphing wing behavior using proposed rate-dependent Prandtl-Ishlinskii models". In: *Proceedings of the Institution of Mechanical Engineers. Part I: Journal of Systems and Control Engineering* (2020). ISSN: 20413041. DOI: `10.1177/0959651819861249`.

[53] M. Al Janaideh. "A time-dependent stop operator for modeling a class of singular hysteresis loops in a piezoceramic actuator". In: *Physica B: Condensed Matter* (2013). ISSN: 09214526. DOI: `10.1016/j.physb.2012.12.021`.

[54] M. Al Janaideh and P. Krejčí. "Inverse rate-dependent Prandtl-Ishlinskii model for feedforward compensation of hysteresis in a piezomicropositioning actuator". In: *IEEE/ASME Transactions on Mechatronics* (2013). ISSN: 10834435. DOI: `10.1109/TMECH.2012.2205265`.

[55] S. Shakiba, A. Yousefi-Koma, and M. Ayati. "Tracking control of an SMA-driven actuator with rate-dependent behavior using an inverse model of hysteresis". In: *Journal of the Brazilian Society of Mechanical Sciences and Engineering* 42.8 (2020). ISSN: 18063691. DOI: `10.1007/s40430-020-02486-0`. URL: `https://doi.org/10.1007/s40430-020-02486-0`.

[56] O. Aljanaideh, M. D. AL-Tahat, and M. Al Janaideh. "Rate-bias-dependent hysteresis modeling of a magnetostrictive transducer". In: *Microsystem Technologies* (2016). ISSN: 09467076. DOI: `10.1007/s00542-015-2566-8`.

[57] S. Shakiba, M. R. Zakerzadeh, and M. Ayati. "Experimental characterization and control of a magnetic shape memory alloy actuator using the modified generalized rate-dependent Prandtl–Ishlinskii hysteresis model". In: *Proceedings of the Institution of Mechanical Engineers. Part I: Journal of Systems and Control Engineering* (2018). ISSN: 20413041. DOI: `10.1177/0959651818758910`.

[58] S. Zhang, M. Wang, P. Zheng, G. Qiao, F. Liu, and L. Gan. "An Easy-to-Implement Hysteresis Model Identification Method Based on Support Vector Regression". In: *IEEE Transactions on Magnetics* (2017). ISSN: 00189464. DOI: `10.1109/TMAG.2017.2702168`.

[59] T. Kleynhans, M. Montanaro, A. Gerace, and C. Kanan. "Predicting top-of-atmosphere thermal radiance using MERRA-2 atmospheric data with deep learning". In: *Remote Sensing* (2017). ISSN: 20724292. DOI: `10.3390/rs9111133`.

[60] S. Hochreiter and J. Schmidhuber. "Long Short-Term Memory". In: *Neural Computation* (1997). ISSN: 08997667. DOI: `10.1162/neco.1997.9.8.1735`.

[61] D. Wu, M. Ourak, K. Niu, Y. Zhang, M. Ahmad, J. Dankelman, and E. V. Poorten. "Towards Modeling of Hysteresis in Robotic Catheters based on LSTM". In: *32nd Annual Conference of the International Society for Medical Innovation and Technology (iSMIT)* (2020), pp. 10–13.

[62] D. Wu, M. Ourak, M. Ahmad, K. Niu, G. Borghesan, J. Dankelman, and E. V. Poorten. "Feasibility of using a Long Short-Term Memory Network for Robotic Catheter Control *". In: *10th Conference on New Technologies for Computer and Robot Assisted Surgery(CRAS)* (2020), pp. 68–69.

[63] W. T. Ang, P. K. Khosla, and C. N. Riviere. "Feedforward controller with inverse rate-dependent model for piezoelectric actuators in trajectory-tracking applications". In: *IEEE/ASME Transactions on Mechatronics* 12.2 (2007), pp. 134–142. ISSN: 10834435. DOI: `10.1109/TMECH.2007.892824`.

[64] H. Sayyaadi and M. R. Zakerzadeh. "Position control of shape memory alloy actuator based on the generalized Prandtl-Ishlinskii inverse model". In: *Mechatronics* (2012). ISSN: 09574158. DOI: `10.1016/j.mechatronics.2012.06.003`.

[65] S. L. Xie, H. T. Liu, J. P. Mei, and G. Y. Gu. "Modeling and compensation of asymmetric hysteresis for pneumatic artificial muscles with a modified generalized Prandtl–Ishlinskii model". In: *Mechatronics* 52.November 2017 (2018), pp. 49–57. ISSN: 09574158. DOI: `10.1016/j.mechatronics.2018.04.001`. URL: `https://doi.org/10.1016/j.mechatronics.2018.04.001`.

[66] M. Al Janaideh, J. Mao, S. Rakheja, W. Xie, and C. Y. Su. "Generalized Prandtl-Ishlinskii hysteresis model: Hysteresis modeling and its inverse for compensation in smart actuators". In: *Proceedings of the IEEE Conference on Decision and Control*. 2008. ISBN: 9781424431243. DOI: `10.1109/CDC.2008.4739202`.

[67] D. Wu, Y. Zhang, M. Ourak, K. Niu, J. Dankelman, and E. V. Poorten. "A Deep-learning-based Framework for Position Control of Robotic Catheters". In: *International Conference on Intelligent Robots and Systems (IROS)(Submitted)* (2021).

[68] F. Bourier, R. Fahrig, P. Wang, P. Santangeli, K. Kurzidim, N. Strobel, T. Moore, C. Hinkel, and A. Al-Ahmad. "Accuracy assessment of catheter guidance technology in electrophysiology procedures: A comparison of a new 3D-based fluoroscopy navigation system to current electroanatomic mapping systems". In: *Journal of Cardiovascular Electrophysiology* (2014). ISSN: 10453873. DOI: `10.1111/jce.12264`.