



Modelling and control of Piano Keystroke with a Soft Particle Jamming Finger

Author Name: Yuyi Zhang
Supervisor: Prof Fumiya Iida
Mentor: Huijiang Wang

Date: May 31, 2022

I hereby declare that, except where specifically indicated, the work submitted herein is my own original work.

Signed: 张宇迪 Date: May 31, 2022

Technical Abstract

During the last few decades, musical robot development has been an area arousing increasing research interest. Scientists have been trying to design various anthropomorphic robots to play music in a human-like manner, which not only entertains audiences around the world, but also helps with our understanding of human control in dexterous tasks. The most challenging part in musical robot development is to captivate the audience with expressive and passionate performance, which requires the robots to interact intelligently and engagingly with the musical instrument.

To bridge the gap between robot and human in expressive piano playing, this project proposes improvement on current solutions from two aspects: i) finger design: use a anthropomorphic variable stiffness soft finger for piano keystroke; ii) control design: develop a reduced-order analytical model which relates robot motion to piano sound generation for feedforward keystroke control. The aim of this project is to build a control system with both model-based feedforward component and various sensory feedback components. The feedforward-feedback control system will be capable of performing accurate, precise, timely and expressive keystrokes using a soft finger with complex particle jamming structure. In this project, we make one step towards the ultimate aim by investigating internal model used by feedforward control component.

While soft robots allow safe and natural interaction with the piano, the non-linear and time-variant nature of soft materials makes the modelling and control extremely challenging. Existing solutions like finite element model (FEM) either are computationally expensive or exclude the external environment from the model, making them inappropriate for high-timeliness interactive tasks like piano playing.

In this project, we proposed a lightweight analytical model to predict the passive behaviors of a finger-key system with variable material properties during a piano keystroke. It can also serve as an equivalent internal model for soft robot piano keystroke feedforward control. In our system, a soft finger based on particle jamming was fabricated to have soft-bodied contact with a piano. The non-linearity of soft-finger contact was modeled as a 2-DoF mass-spring-damper system, allowing the keystroke action to be analytically characterized and predicted. A state-space model was presented to show the keystroke as a single-input multiple-output (SIMO) system and to study the dynamic system's behavior under varied stiffness conditions. The experimental results demonstrate that the analytical model is capable of describing the soft finger piano keystroke with high accuracy. The method can be a valuable substitute for analyzing the high-order compliance of soft-bodied manipulation. It also contributes to robotic realisation of neuroscience feedforward control based on an internal model generated according to both body and environment dynamics.

Contents

1	Introduction	3
1.1	Motivation	3
1.2	Aims and Objectives	4
1.3	Contributions	4
1.4	Report Structure	5
2	Background	6
2.1	Piano Playing in Neuroscience Perspective	6
2.2	Piano Playing Robots	7
2.3	Soft Robot Hand	8
2.4	Soft Robot Stiffening	9
2.5	Soft Robot Modelling	9
2.5.1	Finite Element Analysis Model	10
2.5.2	Physical Model	10
2.5.3	Reduced-order Model	11
2.6	Soft Robot Control	12
2.6.1	Model-Based Control	12
2.6.2	Model-Free Control	12
3	Experimental Method	13
3.1	Experimental Setup	13
3.1.1	Soft Particle Jamming Finger	14
3.1.2	Musical Instrument	15
3.1.3	Motion Capture System	16
3.2	Robot Control	17
3.2.1	Robot Calibration	17
4	Theoretical Model	18
4.1	Learning-based Model of Keystroke	18
4.1.1	Pre-experiment	19
4.1.2	Data collection	19

4.1.3	Data modeling	20
4.2	Mass-spring-damper Model of Keystroke	20
4.2.1	Forward State-space Representation	21
4.2.2	Inverse State-space Representation	22
4.3	Definition of Multi-phase Keystroke	24
5	Results	25
5.1	Learning-based Model	25
5.1.1	Pre-experiments	25
5.1.2	Rigid finger single note	27
5.1.3	Soft finger single note	27
5.1.4	Rigid finger sequential notes	29
5.2	Mass-spring-damper Model	30
5.2.1	System Identification	30
5.2.2	Frequency Response	31
5.2.3	Keystroke Stage Prediction	32
5.2.4	Displacement Prediction	33
5.2.5	MIDI Message Prediction	35
6	Discussion and Conclusion	38
6.1	Rigid vs Soft Finger	38
6.2	Model Selection	38
6.3	Mass-spring-damper Model Analysis	40
6.4	Summary of Findings	40
6.5	Limitation and Future Work	41
6.5.1	Beyond Monophonic Piano Playing	42
6.5.2	Adaptive Feedforward Internal Model	42
6.5.3	Feedback Control	43
7	Bibliography	44
A	Risk Assessment Retrospective	49
B	Additional Resource	50
B.1	Project Repository	50
B.2	Video Demonstrations	50
B.3	Log Book	50

Chapter 1

Introduction

1.1 Motivation

Outstanding piano playing has been entralling people over centuries [1]. Excellent piano performance requires fast, accurate, dexterous, and efficient movements of the upper body muscles, which is built on exceptional cognitive ability of Central Neural System (CNS) towards the body and the piano [2]. For each keystroke action, the CNS needs to find the appropriate amount of stimulation so that the skeletal-muscle system makes the desired motion, which produces the sound of correct tone, strength and tempo during its interaction with the piano [3]. With decade-long continuous and deliberate practice, the pianists train their internal model of their body-piano system to relate the skeletal-muscle movements to the generated sound in a more accurate and time-efficient way, which is a key factor for improvements in piano playing performance.

Previous researches about the piano playing robots have made exploration in the finger actuation mechanism and the controller design for robot motion [4, 5, 6]. The developed piano robots have achieved accurate key pressings with optimal trajectory planning and collision avoidance algorithm [7, 8, 9]. Attentions have also been paid on robots' delicate performance of a single keystroke to improve the expressive and emotion of piano playing [10]. While some explorations have been made in piano playing using soft robotic hand with passive dynamics [11, 12], the vast majority of piano playing robots remain to have rigid structures.

In recent years, the adaptability, flexibility and bio-compatibility of soft material make soft robotics a rapidly rising research field with applications in sensing, locomotion, and human-robot interaction [13, 14]. Anthropomorphic soft robotic hands have been designed to bridge the gap between human and machines with their structure-inherent compliance, demonstrating great potential in dexterous manipulation tasks such as piano playing [15]. Equipped with the cutting-edge soft robotic prosthesis, musicians with limb disabilities will have the potential to deliver more natural playing style compared with those using

the traditional rigid robotic prosthesis.

The greatest challenge faced by soft robots in anthropomorphic piano playing is to acquire an appropriate model for the robot-piano system, namely how the soft robot interprets itself and the environment (the piano) it is interacting with. While the details of human sensory-motor control mechanism are still a mystery with heated ongoing discussion in neuroscience, the soft robot modeling remains an unsolved challenge due to the highly non-linear and time-variant nature of soft materials, the imperfection in the production process and the complexity of combined structure with hybrid materials [15, 16]. The requirement of making real-time control decisions in a short time window is present in music playing and many other daily manipulation tasks, which prefers an analytical, lightweight solution to the computationally expensive Finite Element Analysis (FEA) model or the physical model with large DoF [17]. To enable the soft robotic finger to perform natural human like musical instrument playing, we will need a precise systematic modeling of soft robots, which will serve as an equivalent to the internal model used in feedforward control component in limb motor mechanism [18], which is the subject of this project.

1.2 Aims and Objectives

The aim of this project is to enable robot to perform piano playing as expressively as human pianists can do. To achieve this, this project proposes a design and fabrication of a hybrid material particle jamming soft robotic finger with similar structure to human fingers. We then focus on controlling a single keystroke action performed by the soft finger. We refer to the internal model formed in human neural system during piano playing training and develop a similar model which describes the finger-key system and relates the input finger motion to the output piano key movement. We evaluate the model's performance in terms of accuracy in predicting piano key movement from finger motion, adaptability for finger and piano key stiffness changes, and computation speed. The further aim is to use the model for robot control to reproduce the desired MIDI message by keystroke using variable stiffness finger and changing key stiffness as accurately as possible, which can be evaluated by comparing the target MIDI messages and the real one produced.

1.3 Contributions

In this project, we creatively described the soft finger-key system using a 2-DoF mass-spring-damper model. We brought the idea of internal model based feedforward control from neuroscience to robotic realisation. The internal model we developed for soft finger keystroke outperforms existing soft robot models in terms of model complexity, computa-

tion speed, and adaptability to environment and body changes. Our solution of controlling piano playing soft robot with an internal mass-spring-damper model will not only improve the performance of robots in human-like piano playing tasks, but also help neurologists to gain a better understanding on human sensory-motor control system.

In summary, this project makes a number of contributions:

- Development of an analytical, lightweight and adaptive mass-spring-damper model for soft robotic particle jamming finger piano keystroke.
- Mathematically formulation of a state-space model to analyze the high-order non-linearity of soft deformation under variable stiffness conditions.
- Fabrication of an anthropomorphic finger with a variable-stiffness finger based on particle jamming.
- Robotic realisation of the neuroscience feedforward control based on internal model of body and environment.
- Collection of soft finger keystroke data using 3D motion capture system.

1.4 Report Structure

Section 2 starts with an introduction to the feedforward control in neuroscience and its relationship with the robot control. It then reviews the current methods of soft robot modelling. Based on the modelling method, various kinds of control techniques are discussed and compared.

Section 3 gives an introduction to the experimental apparatus used throughout the project, which includes the soft particle jamming finger, the piano instrument, the 3D motion capture system, and the UR5 robot arm.

Section 4 described two models we used to analyse the keystroke: the learning based model and the physical mass-spring-damper model. We also proposed the definition of different phases in a keystroke.

The results analysis of the above two models are included in section 5. We gained the model parameters from the keystroke data, and then use the trained model to predict the output of new keystrokes. A comparison of these models are provided.

Section 6 provides a conclusion of the project’s main finding. It also discusses limitation of the proposed model and suggestions for future work.

Chapter 2

Background

With the rapid development of robotic technology, there has been an increasing interest in creating robots that can play music in a human-like manner [10]. The creation of anthropomorphic piano performance robots have not only been entertaining audiences all around the world with their brilliant performances, but also helps us to understand human control in the dexterous piano playing tasks [1, 19]. In this chapter, we first review the human control on piano keystroke in a neuroscience perspective. We then introduce the existing piano playing robots and discuss possible improvements to be made in this project. At last, we talk about the fabrication, modelling and control of soft robots and the advantages and potential challenges in applying soft robots in music playing tasks.

2.1 Piano Playing in Neuroscience Perspective

Pianists perform piano keystrokes commonly by lifting the arm to some height, dropping the arm for the finger to hit and depress the target key against the reaction force from the key with a velocity determined by the target loudness level of the eliciting tone, and lifting the hand again to release the key depression [20]. Body motions in piano performance, like other musical actions, are goal-directed, striving to produce intended sounds with greatest precision and accuracy in expressive aspects including time, dynamics, timbre, and articulation [21]. With multiple upper body muscles and joints included in the keystroke action, piano keystroke serves as a typical example of multi-joint movement coordinated by CNS.

Several studies show that successful multi-joint movements require effective feedforward exploitation or compensation of inter-segmental interaction torques to keep the limb end moving along certain trajectory at specified speed under the environment interference [3, 22, 23, 24]. Such feedforward control model for muscular torque for certain task is developed in the training process. A training of limited time can effectively shorten movement duration, increase the movement speed and maximum acceleration at the shoulder, elbow, and wrist joints, while improving the overall task performance [25, 26]. The long-

term effect of training can be investigated by comparing the dominant and non-dominant arms in completing some daily multi-joint tasks. Sainburg et al. found a smaller elbow extension muscular torque accompanied by a larger inter-segmental interaction torque in the dominant side as compared with the nondominant side during a constant speed target-reaching task, which indicates the CNS could maximize the use of inter-segmental interaction and minimize the muscular work to a complementary level for performing the task after years of training, supporting the idea of Bernstein [20, 27, 28].

The features of internal feedforward control model in CNS can be summarized as follow:

- *Fast computation*: Prediction of a movement's effect can be made in a short time window, resulting in a
- *Training-based*: Model formation relies on the task training process. Model performance in terms of accuracy and efficiency improves with training.
- *Task-specified*: Separate models need to be trained for different kinds of tasks, say pianists cannot use their experience in piano playing to become proficient in violin in a short time. However, model for similar tasks might not be independent. The knowledge of CNS for one task might be migrated to a similar task (for example, playing the piano and the pipe organ).
- *Adaptability*: The model can adapt to the changing dynamic of body caused by bone growth and increasing muscle mass [3]. The model can update according to the environment change after some training in the new environment, say it usually takes a pianist some time to adapt to an unfamiliar piano.

Similar phenomenon can also be found by comparison of piano keystroke movement performed by expert and novice piano players [20], indicating a formation of internal feed-forward control model for the body-piano system in CNS. While the internal feedforward control model plays an indispensable role in human piano playing, it is natural to develop a similar feedforward control model for robot piano playing with the above features, which is the aim of this project.

2.2 Piano Playing Robots

Existing piano playing robots focus mainly on the finger actuation mechanism and the controller design for robot motion [12, 29]. Common actuation methods for finger keystroke motion includes tubular solenoids [4] and servomotors [5, 6]. On the top of various actuation methods, control algorithms are built at different levels from basic manually hard-coded one [5] to the more automatic ones capable of optimal trajectory planning and

collision avoidance [7, 8, 9]. Computer vision system has also been used to enable robot to read the music score and generate the keystroke control command accordingly [7, 4]. Furthermore, attention has also been paid on robots' delicate performance of a single keystroke to enable more expressive playing styles [10]. Learning-based control method like Gaussian Process (GP) [10] and Recurrent Neural Networks (RNN) [6] have been applied to relate different keystroke styles with robot motion commands.

While some explorations have been made in piano playing using soft robotic hand with passive dynamics [11, 12], the vast majority of piano playing robots remain to have rigid structures. The application of soft anthropomorphic hands in music playing can potentially improve the robot's performance by minimizing the contact noise and achieve a diversity of expressive playing style by introducing stiffness tuning techniques.

2.3 Soft Robot Hand

Recent years have witnessed a rapid development of anthropomorphic soft robotic hands, which bridges the gap between human and machines with their adaptability, flexibility and bio-compatibility, demonstrating great potential in safe and dexterous manipulation tasks such as adaptive grasping, in-hand manipulation [30, 31, 32]. Their capacity to adapt to an unknown environment enables steady performance in object grasping of varying shape, size, and position without explicit previous knowledge [33, 34]. Passive compliance also allows soft robots to interact extra gently with delicate target objects or other fragile structures in the environment [35]. Soft robotic hands also have advantage in in-hand manipulation tasks with less requirement of complex planning and control as in rigid robot hand cases [33]. Passive anthropomorphic soft hands have also demonstrate great potential in expressive, artistic tasks like piano playing [11, 12].

Various soft actuation methods are applied for soft finger actuation, from Mckibben artificial pneumatic muscles [36], to embedded shape memory alloy (SMA) actuator [32] and reinforced fiber [37]. Soft sensors are integrated in the hand skin to measure deformation and contact, providing tactile feedback for reliable control and information about the external environment [38, 39]. Furthermore, the nature of soft material makes the robot properties (like stiffness, etc.) easily changeable by manipulating the pneumatic pressure, electric voltage, light condition or temperature. The modifiable robot feature provides us an opportunity to investigate an advanced control system which adapts to robot property changes, which will possibly give us some insight in how human neural system adapt to body dynamic changes as we age.

2.4 Soft Robot Stiffening

The high deformability and compliance of soft robots make them a safer choice in human and environment interaction tasks, while bringing us new challenges in controlling their deformability and compliance. Soft robots need variable stiffness to selectively interact with the environment: deformation enables soft contact, while stiffening transfers force and energy [40]. The stiffening phenomenon is observed in octopus legs, squid tentacles, and elephant trunk in nature [40], and achieved artificially by carefully designing the arrangement of electroactive polymers (EAPs), elastomeric chambers as in flexible fluidic actuators (FFAs), shape memory alloys (SMAs) and jamming-based systems in robotic applications [41, 42, 43].

In particular, granular jamming techniques has been trending among scientists for its simplicity, feasibility, reliability and capability of switching between a fluid-like and a solid-like state with limited volume change in a reversible manner [40, 44, 45]. Brown et al. were the first to incorporate the ground particle jamming phenomena into a soft gripper design capable of grasping a wide range of irregular items without providing active feedback [46]. In the basic state, particles are loosely contained inside a membrane sac with low stiffness because they can readily flow around. When the air inside the membrane sac is pumped out, atmospheric pressure is applied to the particle system, causing interparticle forces and a high stiffness [45]. A particle jammed soft finger's variable stiffness and soft texture can potentially parody a human hand's performance in piano playing by displaying more flexibility and variation in how it interacts with the piano key. The potential of enabling local stiffness control demonstrates possibility of fabricating soft hands with complex variable stiffness structure using particle jamming [47]. By developing a model for particle jamming finger, we exploit the model's ability to adapt to soft robot stiffness changing, which will serve as a beneficial exploration to more advanced adaptive models.

2.5 Soft Robot Modelling

The soft robot modeling remains an unsolved challenge due to the highly non-linear and time-variant nature of soft materials, the imperfection in the production process and the complexity of combined structure with various materials [15, 16]. The interaction with dynamic environment adds further complexity to acquiring a universal model for the robot-environment system. Current modelling solutions include mainly FEA and physical models. Effort has also been made in exploring the reduced-order model. To establish effective feedforward control, we will need the internal model to be computationally efficient, adaptive to environmental changes and to the changing properties in soft robot itself by simple model modifications. The detailed discussions of current models in these perspectives are presented as below.

2.5.1 Finite Element Analysis Model

The state-of-the-art FEA method has been extensively used as a generic approach in soft robot non-linear deformation visualization, contact analysis, and design optimization [30, 48, 49]. In FEA, we subdivides a large system into smaller, simpler parts called finite elements by space discretization. The algebraic equations for these finite elements are then acquired and assembled into a larger system of equations which models the entire system. The FEA then approximates a solution by minimizing an associated error function via the calculus of variations.¹ Based on finite element models, techniques like differentiable simulation have been introduced to narrow down the sim-to-real gap raised by the non-linearity of soft materials [13, 16]. While FEA is able to numerically represent the continuous deformation of soft robots with high fidelity, the high-dimensional nature limits FEA to simple structured robots [50, 51]. Furthermore, the high-order matrix inverse required by FEA tremendously increases the cost of computation, limiting its use for real-time control problems [50]. Change in soft robot structure means reconstruction of the mesh structure used by FEA. The dynamic equation of each single element needs to be updated upon the change soft material properties, for example, caused by particle jamming. Considering the drawback in computational complexity, the lack of adaptive capability and the absence of environment element, FEA is not an ideal internal model for soft robot piano keystroke.

2.5.2 Physical Model

Apart from FEA, physical models have also been used to describe the soft robot locomotion [14] and electromechanical behavior [52]. Dynamic model developed by geometrically exact approach was effective for soft robots with uniform and isotropic properties [17]. However, such physical models can have infinite DoFs and become non-solvable for soft robots integrating multiple materials into a sophisticated structure, such as the particle jamming finger. In addition, physical models are task-specific and lack the transferability. However, in the context of soft robotic finger piano keystroke, a physical model with limited DoF is desired. The interactive environment here is simplified to piano key with changing stiffness, which can be easily integrated to the interaction model as a variable stiffness spring, give some degree of adaptedness to environmental changes. The stiffness change of particle jamming finger can also be represented by the changing stiffness in finger components in the model. The computation is straight-forward with a state-space representation of the physical model. The rest of this project will be elaborate on the physical model of our choice.

¹https://en.wikipedia.org/wiki/Finite_element_method

2.5.3 Reduced-order Model

Exploration has been made in approximating the infinite DoF soft robot structures using limited number of dimensions, as demonstrated in constant curvature (CC) approximation [53]. Although the CC approximation neglects some manipulator dynamics, it achieves good accuracy with uniform manipulator, symmetrical actuation , minimal external force and torsional effect [54]. More complex models can be provided by combining CC sections together to form the piecewise constant curvature (PCC) model [55]. However, the increase in accuracy offered by larger model complexity was not great enough, given their computing and sensing costs, and therefore their use has been limited [54].

Though current reduced-order models have yield satisfactory result in manipulation accuracy, the interaction with external environment is ignored. In this project, we will include the environment (piano key) as a part of our reduced-order model and study the interaction happening in finger-key system as a whole.

2.6 Soft Robot Control

Success of robots in various tasks relies on the development of fast, reliable, accurate, and energy-efficient controllers. However, this is no unified control solution for soft robots due to their high DoF, non-linear features such as compliance and hysteresis, and great diversity in structural design and actuation methods [54]. In this section, we talk about two major control strategies for soft robots: model-based control and model-free control.

2.6.1 Model-Based Control

Model-based controllers are derived by inverting the kinematic models to obtain the mapping from target motion to the actuation commands [54]. Constant curvature (CC) approximation is the most popular model of uniform, low-mass manipulators under steady-state assumption [53]. Adding complexity to CC models has not led to exceptional control performance improvements while increases the computational cost dramatically [56]. When the timing becomes an important factor in task performance (like in musical instrument playing), dynamic controllers become necessary. Current the model-based dynamic control approaches mainly focus on the joint space control. Open loop feedforward controllers are more commonly used due to the computational complexity of dynamic models [54].

2.6.2 Model-Free Control

Model-free learning-based approaches are also commonly investigated in various tasks without particularly looking into the soft robot structure or material properties, but focusing on the input and output of the system. The data-driven control models developed are more complex but can achieve accurate, task-specific performance [54]. Model-free static controller in [57] is built by direct learning of the inverse statics of a non-redundant soft robot using a neural network. While the control model provides accurate prediction of cable tension, it cannot scale for redundant systems and does not account for the stochastic character of soft robotics. The basic neural network-based strategy can usually outperform the computationally complicated analytical method [54].

Model-free control has wide usage in building expressive soft music robots as well. Using Gaussian Process, Scimeca et al.[10] achieved 10 different playing styles by mapping the non-linear relationship between MIDI signal and UR5 robot control parameters. By training an artificial neural network (ANN) to mimic a real flutist's vibrato and note duration, Solis et al.[58] achieved a significant gain in expressiveness of flutist robot's performance.

Chapter 3

Experimental Method

3.1 Experimental Setup

Figure 3.1 shows the overall experimental setup for piano keystroke action with a variable-stiffness soft finger. During the experiments, we program the code instructing the robot arm action which are sent to the robot controller by the computer processor. The robotic finger attached as an end-effector of the robot arm then performs mechanical interaction with the piano keys. The digital piano generates the audio data in MIDI format and sends the message back to computer which keeps monitoring the USB input port. The robot, finger and piano key movement during the whole keystroke process is recorded by the 3D motion capture system as sequences of marker position in Cartesian coordinates. The model parameters are then gained from analysing either the MIDI message or the camera captured marker position.

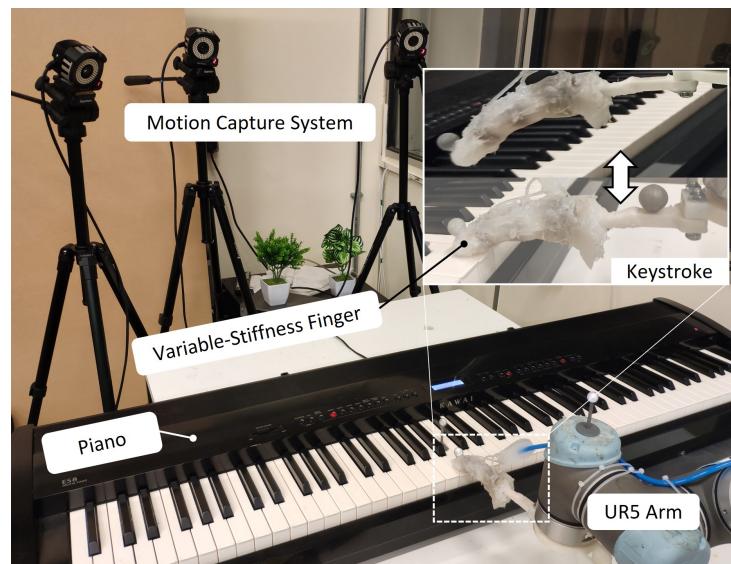


Figure 3.1: Experimental setup for keystroke action with a variable-stiffness soft finger.

3.1.1 Soft Particle Jamming Finger

Finger Fabrication

The soft finger is designed as a hybrid of soft and rigid system, with a 3D printed skeleton (Rigur RGD450) wrapped in a cast silicone skin (Ecoflex™ 00-30 rubber). The finger joints are filled by ground coffee acting as the media of particle jamming. Silicone rings are placed around the skeleton between joints to prevent granular particle transition between adjacent chambers, such that an independent vacuum chamber is formed at each joint. After assembly, the whole finger is dipped into liquid silicone for air tightness enhancement.¹

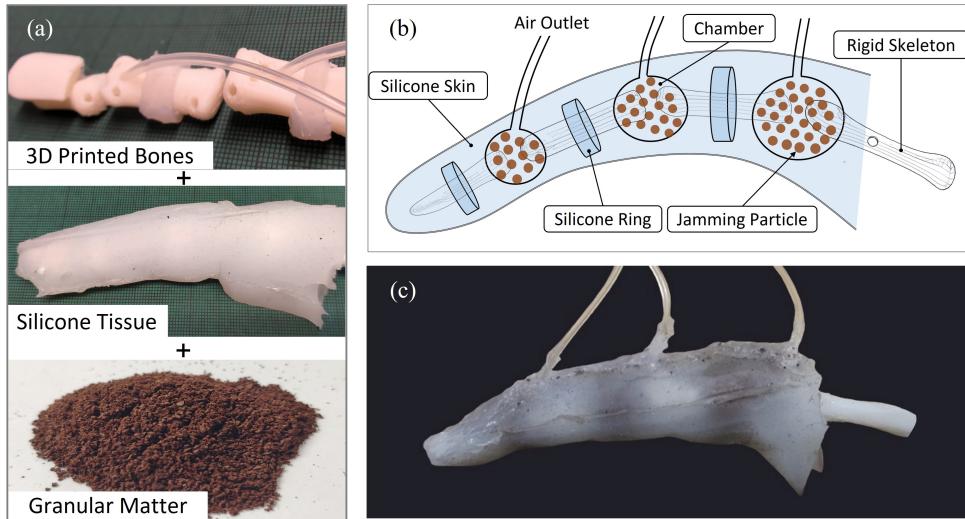


Figure 3.2: The fabrication of the variable stiffness finger. (a) Materials include the 3D printed skeleton with silicone ring and tubing, silicone skin, and granular ground coffee; (b) Schematic of the particle jamming; (c) Fabricated soft finger.

Stiffness Control

PVC tubes are installed in holes left on skeleton to connect the particle jammed joints to a pneumatic regulator (SMC IRV10/20)². A tiny piece of cloth is stuck to the end of each air tube to prevent the coffee powder from being inhaled in and blocking the tube. The pneumatic regulator is connected to a compressed air valve via a vacuum generator (SMC ZH10B)³. By tuning the pneumatic regulator, we can change the stiffness of the soft finger by manipulating the vacuum pressure ($-80 \sim 0 \text{ kPa}$) in the particle chambers.

As a comparison to the soft particle jamming finger, a 3D-printed rigid Thermoplastic Polyether-Polyurethane elastomer (TPE) finger with shore hardness 82A⁴ is also used to

¹A detailed guide on how to build the soft particle jamming finger can be found here: *How to build a particle jammed soft finger*

²<https://www.smc.eu/en-eu/products/irv10-20-vacuum-regulator~54222~cfg>

³<https://www.farnell.com/datasheets/309118.pdf>

⁴<https://recreus.com/gb/filaments/9-filaflex-82a.html>

perform piano keystroke experiments. The rigid finger has a cylinder structure and serves as a simple end-effector of UR5 robot arm.

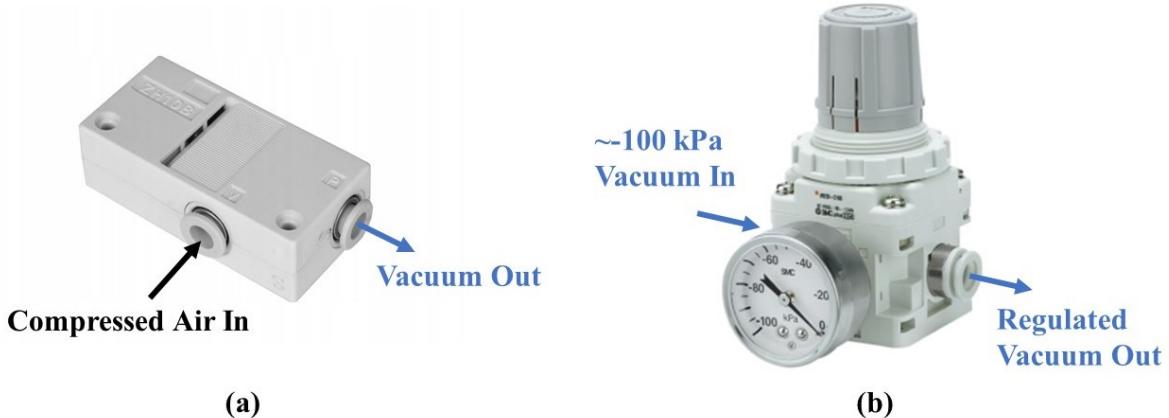


Figure 3.3: Stiffness changing tool of the soft particle jamming finger: (a)Vacuum generator; (b)Pneumatic regulator.

3.1.2 Musical Instrument

The music instrument serves as a specific environment with which the robotic finger interacts actively and dynamically in our project. It also provides a set of sensors, which detect audio signals generated by the robot or a human player's keystroke actions [29].

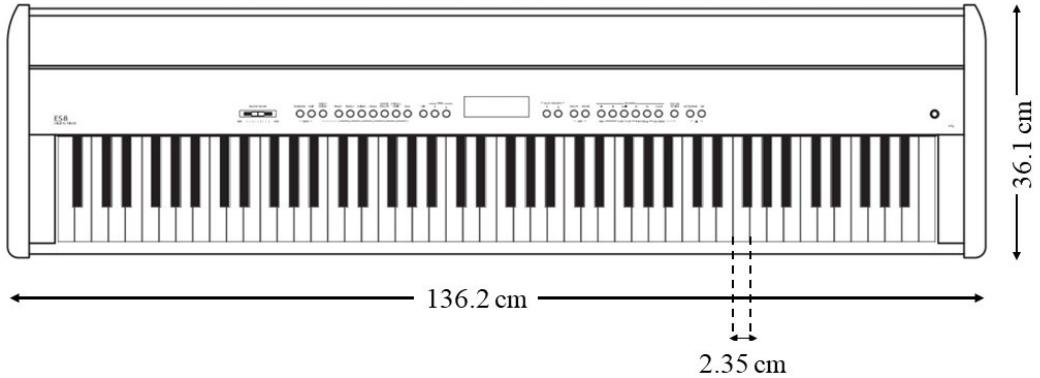


Figure 3.4: Physical dimensions of the Kawai ES8 Digital Piano.

We chose to use the Kawai ES8 Digital Piano⁵ with 88 weighted keys for keystroke detection⁶. It features a hammer that accurately reproduces the dynamic sounds produced by the mechanical hammer-string action in its acoustic equivalent, which is accomplished by the Responsive Hammer Action design. In addition, the hammer weights are 'graded', with larger left-hand bass hammers and lighter right-hand treble hammers to mimic the

⁵<https://www.kawai.co.uk/products/digitalpianos/esseries/es8/>

⁶https://www.bhphotovideo.com/c/product/1211144-REG/kawai_es8b_es8_88_key_triple_pedal.html/specs

key feature of the acoustic piano⁷. The inbuilt counterweights are likewise finely calibrated to lessen the touch of the keyboard during pianissimo passages and deliver a stronger touch while playing strongly⁸. When playing the same key repeatedly, the triple-sensor key detection complexity enables for a high level of touch sensitive precision and responsiveness [29].

Musical Instrument Digital Interface (MIDI)

In digital pianos, MIDI acts as a widely agreed data communication protocol which defines the information exchange between speaker control signals and the piano instrument. The digital piano generates MIDI message from the mechanical strike on the keyboard. The internal electrical circuit then play the recorded piano sound according to the MIDI. One valid keystroke will generate a MIDI message: [[[144, note, pressing velocity,0], pressing timestamp],[[128, note, releasing velocity,0], releasing time stamp]]]. The digital piano captures the changing emotion, the various interpretations and varying playing techniques of players in a MIDI message sequence.

The most important MIDI message for piano sound is the pressing velocity and the gap between two time stamps (hold time). However, MIDI generation is not always stable due to the unknown properties of the digital piano. Thus in this project, we define the pressing velocity of piano key at a threshold to be *on_velocity*, and the time the piano key stay below the threshold to be *hold_time*. We use *on_velocity* and *hold_time* to evaluate the keystroke output, as a substitution of MIDI message when MIDI recording becomes unstable for the soft finger.

3.1.3 Motion Capture System

We applied a 3D infrared motion capture systems (OptiTrack) to record the detailed movement of soft finger in a keystroke action. The system is composed of three OptiTrack S250e cameras⁹ and the Motive optical motion capture software¹⁰. Before usage, the cameras are calibrated by rotating the calibration rod to add at least 5000 sample points for each camera in the software. The horizontal surface is located by another metal calibration frame with a spirit level. By adding reflective markers to the UR5 end-effector, the soft finger's tip and the piano key surface, their real-time positions are captured in 3D Cartesian coordinate at a rate of 120 FPS with a precision of 0.001m and a latency less than 9ms during keystroke actions. The markers' real-time position changing is visualized in the Motive software. We export the time-position data into excel file for further analysis.

⁷<https://kawaius.com/technology/wooden-key-actions/>

⁸<https://www.kawai-global.com/product/es8/>

⁹<https://optitrack.com/support/hardware/s250e.html>

¹⁰<https://optitrack.com/software/motive/>

3.2 Robot Control

For the experiments we connect the fabricated particle jammed soft finger to a UR5 robotic arm¹¹. By setting the tool center point (TCP) to be the tip of soft finger, we move the finger up and down to press the piano key by controlling the vertical acceleration, velocity and displacement of UR5 TCP in Cartesian coordinate at 125Hz. We control the UR5's TCP movement to affect the keystroke actions. The control parameters might include pressing distance, pressing velocity, pressing acceleration, releasing velocity, releasing acceleration, hold time, pressing depth. For sequential midi data, we include moving velocity and moving acceleration for the finger movement between two adjacent pressings. For soft finger, the vacuum pressure of the particle jamming capsules is one of the control parameters too. The UR5 TCP moving velocity under a single command follows the trapezoidal shape. The robot arm first accelerate to the velocity by the specified acceleration, then maintain the velocity set, then decelerate to zero velocity when reaching the specified distance, given the pressing distance is long enough for the whole acceleration and deceleration process.

3.2.1 Robot Calibration

At the start of experiments, we manually calibrate the soft finger tip to be touching but not pressing the B piano key on the left of central C. The fingertip is placed 1.5 cm away from the key edge with the skeleton connected to UR5 kept parallel to the horizontal ground and vertical to the piano key edge. The X axis of TCP movement is set to be parallel to the piano key edge. The Y axis is along the piano key. The Z axis is orthogonal to piano key surface. The keystroke action on a single piano key includes only the movement of robot TCP in Z direction. For music piece demo where multiple piano keys are used, we locate the piano key by adding *number of keys in between* \times *key width* to the Y coordinate.



Figure 3.5: Robot calibration.

¹¹<https://www.universal-robots.com/products/ur5-robot/>

Chapter 4

Theoretical Model

4.1 Learning-based Model of Keystroke

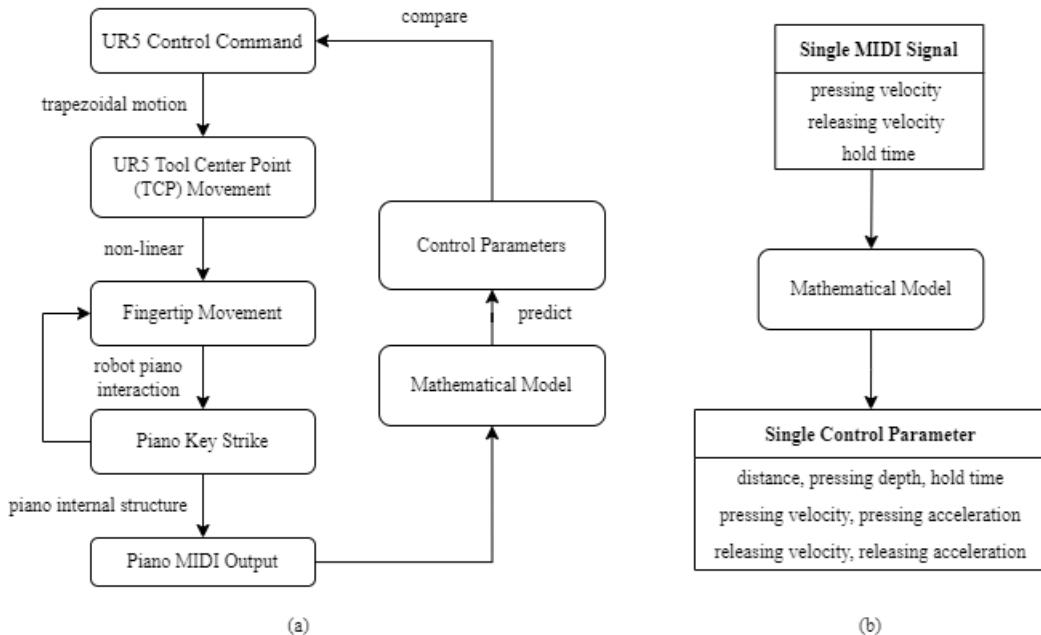


Figure 4.1: Piano playing robot system diagram, including (a) the model relationship between robot key-press and piano sound outputs and (b) the mathematical model calculate control parameters from a single MIDI signal

The robot-piano set-up combines to be a non-linear system (Figure 4.1). The velocity of the fingertip movement is not the same as the UR5 TCP velocity due to the deformation of robotic finger when interacting with the piano. The non-linear bending in the robotics finger, as well as non-linear relationship between fingertip pressing velocity and MIDI message due to the internal mechanic of the digital piano, together formed a non-linear system we wish to model using a mathematical mapping.

4.1.1 Pre-experiment

Before starting to do data collection with control parameters, we first did pre-experiments by changing one control parameter and fixing the others. We observed whether the changing control parameter will cause the pressing/releasing velocity or hold time(gap between pressing and releasing timestamp) to vary in a linear or non-linear manner. In addition to selecting valid control parameters, we also located the proper range of control parameters in the pre-experiment stage.

```
press(down_vel=0.1, up_vel=0.1, distance=0.10, hold_time=0, acc=2,  
      press_depth=press_depth)
```

Listing 4.1: Example code controlling the UR5 robot to press the piano key with down velocity $0.1m/s$, accelerate with $2m/s^2$ from $0.10m$ above the target key surface. After finger touching the key surface, the UR5 will continue going down for distance specified in *press depth*. The UR5 will stay at the bottom for time length specified by *hold time*, which is 0s in this case. Then the UR5 will release the piano key with acceleration $2m/s^2$ and up velocity $0.1m/s$

4.1.2 Data collection

We collected the MIDI messages of rigid finger single note pressing and soft finger single note pressing with grid-searching control parameters within proper ranges located by pre-experiments. By inputting the control parameters to UR5 by `press()` function, we drive the robotic finger to press the central C piano key. Each pressing, if successfully stroke the key, will generate a MIDI message in the form of `[[[144, note, pressing velocity,0], pressing timestamp],[[128, note, releasing velocity,0], releasing time stamp]]`. The control parameters together with the generated MIDI message are recorded by Python program to make one pressing data.

We also collected the sequential MIDI messages from rigid finger sequential notes pressing, in which the control parameters are randomly drawn from uniform distributions over the selected range. The sequences have a fixed length of 12 (12 control parameter sets and their 12 generated MIDIs). If the data collecting program is interrupted in the middle due to connection problem with UR5, the program will start with a new sequence when the connection is reestablished. We collected 100 control and MIDI sequential data with 1151 key pressings. The sequences with length less than 12 will be padded with zeros to make it the same length as others in the modeling process. We then use RNN and LSTM to map the sequential MIDI to its corresponding sequential control parameters. The timestamp of key pressing and releasing, which is neglected in the single note experiment, is also included in the modelling process.

4.1.3 Data modeling

We use several mathematical tools (smoothing spline, multilayer perceptron, Gaussian Process, LSTM) to model the mapping relationship between the UR5 control parameters and the MIDI pressing velocity. The mathematical models take a MIDI message as input, and output the predicted control parameters. For the single MIDI to single control parameter set mapping, we used Gaussian process and densely connected neural network. For the sequential MIDI to sequential control parameter sets mapping, we used RNN and LSTM with a fully connected output layer.

4.2 Mass-spring-damper Model of Keystroke

The piano keystroke with the variable stiffness finger is mechanically represented by a 2-DoF mass-spring-damper system. The stiffness and damping coefficient fundamentally govern the states of the dynamic system. We proposed a state-space model to represent the piano keystroke based on the dynamic system's second-order ordinary differential equations (ODEs).

Piano playing is a sequential procedure that necessitates repeated key-press actions throughout a range of notes. In this context, we are concerned in the delicate modeling of a single-note piano keystroke movement, in which the finger is always in touch with one piano key. Because both the soft finger and the piano key exhibit passive behaviors in the presence of external force. To represent the keystroke mechanism, we employed a linear 2-DoF spring-damping oscillator.

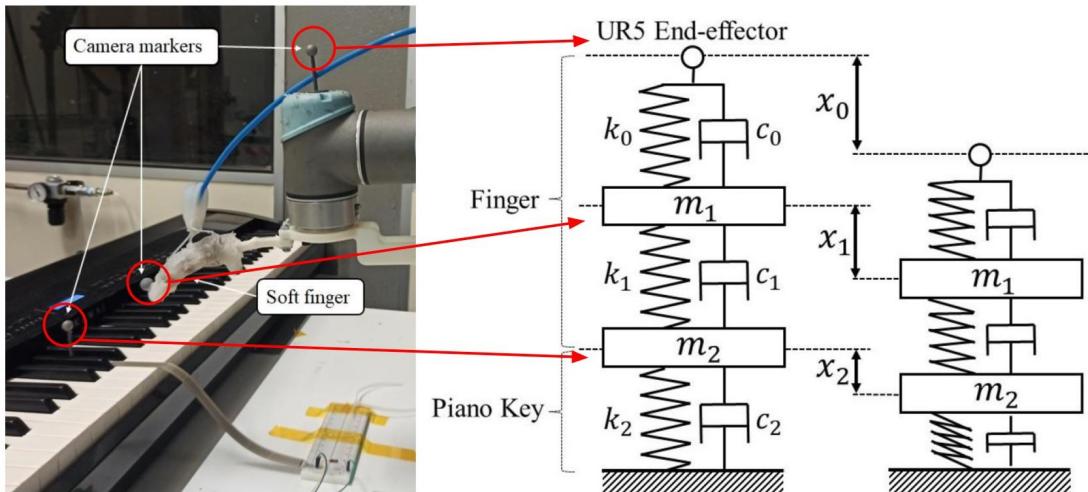


Figure 4.2: Spring-damping representation of the piano keystroke action.

The keystroke process is characterized by a series of springs and dampers. As demonstrated in Figure 4.2, x_0 , x_1 and x_2 denote the displacement of the UR5, soft finger and

piano key respect to their initial positions, respectively. The masses of the finger and piano key are given as m_1 and m_2 . As a result, the mechanical model's dynamics equations are as follows:

$$\begin{cases} m_1\ddot{x}_1 = k_0(x_0 - x_1) - k_1(x_1 - x_2) + c_0(\dot{x}_0 - \dot{x}_1) - c_1(\dot{x}_1 - \dot{x}_2) \\ m_2\ddot{x}_2 = k_1(x_1 - x_2) - k_2x_2 + c_1(\dot{x}_1 - \dot{x}_2) - c_2\dot{x}_2 \end{cases} \quad (4.1)$$

4.2.1 Forward State-space Representation

Based on the dynamic model, we characterized the soft-finger piano keystroke as a second-order mass-spring-damper system. In order to investigate the mechanical behaviors of both the soft finger and piano key within a keystroke action. For the forward case, we chose displacement of UR5 end-effector x_0 as the entire system's control input and aim to examine the states of both the finger and the piano key (i.e., x_1 and x_2) while taking variable stiffness conditions into account, since we are controlling the UR5 robot arm in the experiment. We proposed a state-space model to represent the dynamic system as

$$\begin{aligned} \dot{\mathbf{z}}(\mathbf{t}) &= \mathbf{Az}(\mathbf{t}) + \mathbf{Bu}(\mathbf{t}) \\ \mathbf{y}(\mathbf{t}) &= \mathbf{Cz}(\mathbf{t}) + \mathbf{Du}(\mathbf{t}) \end{aligned} \quad (4.2)$$

where the coefficient matrices are denoted as: state matrix, $A \in R^{n \times n}$; input matrix, $B \in R^{n \times r}$; output matrix, $C \in R^{m \times n}$ and feedforward matrix, $D \in R^{m \times r}$.

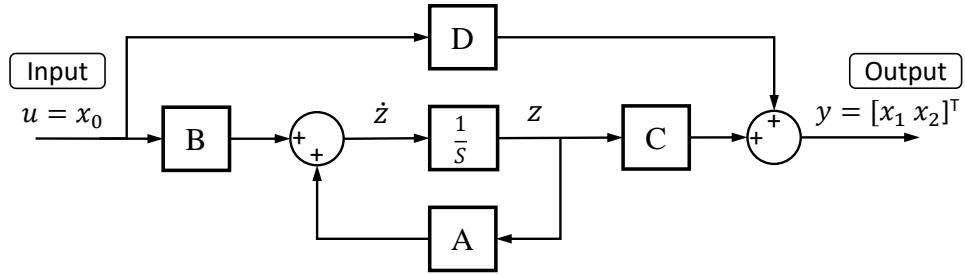


Figure 4.3: Forward state-space model of the system.

The coefficient matrices in this state-space model can be time-variant if it is meant to represent soft interactions with time-variant qualities. In the state-space model, we chose x_0 as the input and x_1 and x_2 as the system's output such that the keystroke action is represented as a single-input multiple-outputs (SIMO) system with $n = 4, r = 1, m = 2$.

$$\begin{cases} u(t) = x_0(t) \\ \mathbf{y}(\mathbf{t}) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \end{cases} \quad (4.3)$$

To investigate the dynamic behavior of the $y(t)$ using UR5 motion control as input,

we modeled the state vectors as follows:

$$\mathbf{z}(\mathbf{t}) = \begin{bmatrix} x_1 & \dot{x}_1 - \frac{c_0}{m_1}x_0 & x_2 & \dot{x}_2 \end{bmatrix}^T. \quad (4.4)$$

The coefficient matrices of the state-space representation are formed by substituting the terms in Eq. 4.5 with Eq. 4.7 and Eq. 4.1. As a consequence, we formulated the four coefficients matrices as

$$\left\{ \begin{array}{l} \mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_0+k_1}{m_1} & -\frac{c_0+c_1}{m_1} & \frac{k_1}{m_1} & \frac{c_1}{m_1} \\ 0 & 0 & 0 & 1 \\ \frac{k_1}{m_2} & \frac{c_1}{m_2} & -\frac{k_1+k_2}{m_2} & -\frac{c_1+c_2}{m_2} \end{bmatrix} \\ \mathbf{B} = \begin{bmatrix} \frac{c_0}{m_1} \\ \frac{k_0}{m_1} - \frac{c_0c_1+c_0^2}{m_1^2} \\ 0 \\ \frac{c_0c_1}{m_1m_2} \end{bmatrix} \\ \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \\ \mathbf{D} = \begin{bmatrix} 0 & 0 \end{bmatrix}^T \end{array} \right.$$

4.2.2 Inverse State-space Representation

For the inverse case, we use the displacement of piano key x_2 as input and the movement of UR5 end effector x_0 as output. The inverse keystroke action is represented as a multiple-input multiple-outputs (MIMO) system with $n = 4, r = 2, m = 2$. The inverse state-space model can serve as a control model to determine the required UR5 movement for target piano keystroke.

$$\begin{aligned} \dot{\mathbf{z}}(\mathbf{t}) &= \mathbf{Az}(\mathbf{t}) + \mathbf{Bu}(\mathbf{t}) \\ \mathbf{y}(\mathbf{t}) &= \mathbf{Cz}(\mathbf{t}) + \mathbf{Du}(\mathbf{t}) \end{aligned} \quad (4.5)$$

where the coefficient matrices are denoted as: state matrix, $A \in R^{n \times n}$; input matrix, $B \in R^{n \times r}$; output matrix, $C \in R^{m \times n}$ and feedforward matrix, $D \in R^{m \times r}$.

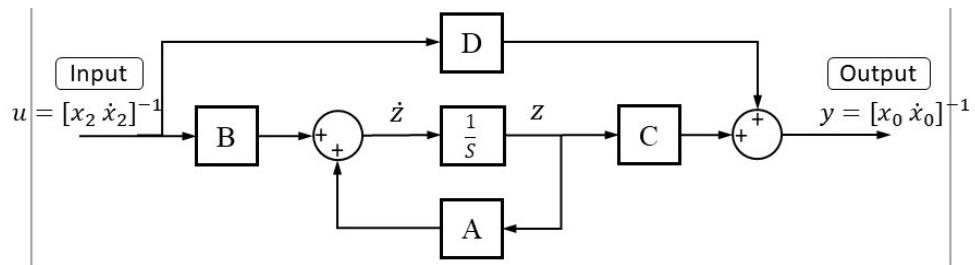


Figure 4.4: Inverse state-space model of the system.

$$\begin{cases} u(t) = x_0(t) \\ \mathbf{y}(\mathbf{t}) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \end{cases}. \quad (4.6)$$

To investigate the dynamic behavior of the $y(t)$ using desired keystroke movement as input, we modeled the state vectors as follows:

$$\mathbf{z}(\mathbf{t}) = \begin{bmatrix} x_0 & x_1 & \dot{x}_0 & \dot{x}_1 \end{bmatrix}^T. \quad (4.7)$$

The coefficient matrices of the state-space representation are formed by substituting the terms in Eq. 4.5 with Eq. 4.7 and Eq. 4.1. As a consequence, we formulated the four coefficients matrices as

$$\left\{ \begin{array}{l} \mathbf{A} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ \frac{k_0}{m_1} & -\frac{k_0+k_1}{m_1} & \frac{c_0}{m_1} & -\frac{c_0+c_1}{m_1} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{k_1}{m_1} & \frac{c_1}{m_1} \end{bmatrix} \\ \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \\ \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \\ \mathbf{D} = \begin{bmatrix} 0 & 0 \end{bmatrix}^T \end{array} \right.$$

4.3 Definition of Multi-phase Keystroke

Based on the displacement of the piano key, we separated a typical keystroke motion into five stages. As depicted in Figure 4.5, there is no displacement before or after the keystroke. When the *press* stage arrives, the finger begins to push the key until they reach the same speed and touch the bottom. When the key passes the triggering threshold, acoustic signals are generated and recorded in MIDI format. The *hold* stage is defined based on the time span during which the velocity is zero. After the *release* stage, which generates another MIDI event, the piano key bounces back to its initial location, while the soft finger recovers the initial pose. Various keystroke patterns can be obtained by altering the motion control of the UR5 end-effector.

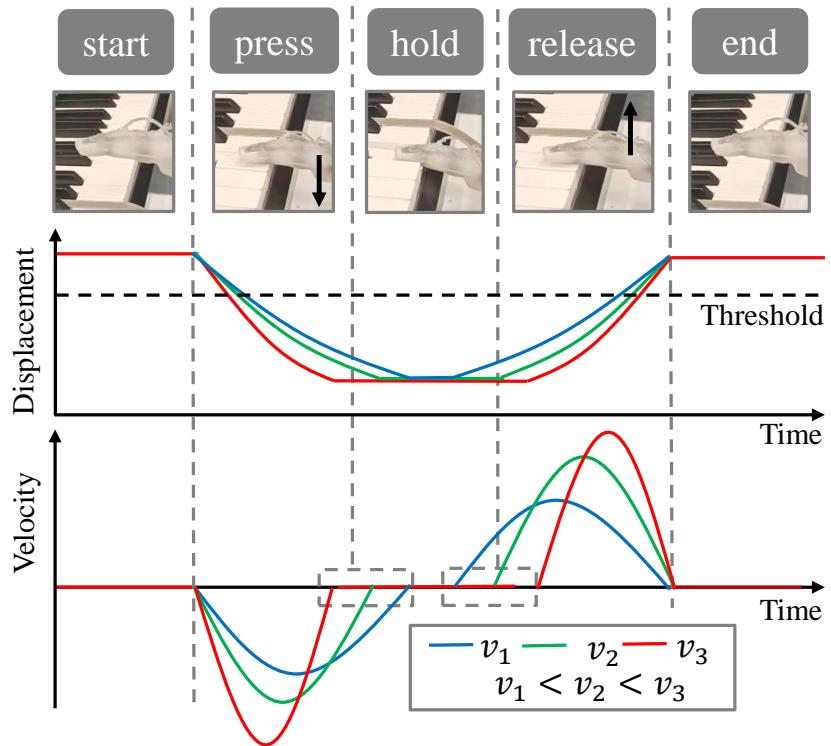


Figure 4.5: Schematic of the 5 stages of a single-note piano keystroke.

Chapter 5

Results

The details of experiments and results are shown in this chapter. In the first part, we present the result about the learning-based model. Our output is MIDI message from the digital piano and no motion capture data is included at this stage. In the second part, we observe the 3D motion of soft finger and piano key during a keystroke captured by cameras and compare the ground truth with predictions made by mass-spring-damper model of the piano key movement.

5.1 Learning-based Model

5.1.1 Pre-experiments

Before official data collecting experiments, we did some pre experiments to identify the triggering threshold and suitable range of keystroke control parameters.

The triggering threshold refers to the shortest moving distance for a rigid finger resting on the key surface without interaction to produce a keystroke with valid MIDI message. The contact between rigid finger and the piano key is assumed to be rigid during the keystroke. Thus the measured finger movement is essential the same as piano key movement. This measured triggering threshold (9mm with precision $\pm 1\text{mm}$, as shown in Figure 5.1 (a)(b)(c)) is used as triggering threshold in following piano key movement analysis as well.

When the velocity or acceleration of UR5 is set too large, the table base of robot arm starts to shake violently and interferes the keystroke experiment. Due to the compliance and deformability of soft material, the MIDI message produced by soft finger does not increase with UR5 down velocity when it is over 0.06 m/s. Small UR5 pressing depth might result in no valid MIDI generated, while large depth will potentially cause damage to the finger and the piano. The MIDI hold time is strictly proportional to the UR5 hold time (Figure 5.1(d)). In the first few rounds of keystroke data collection, we ended up with large number of repeated or invalid MIDI messages due to the poor choice of control parameter

range. In Table 5.1, we identify the proper range of keystroke control parameters for rigid and soft finger considering the experiment safety, data variety, and time efficiency. The following keystroke experiments are performed with control parameters selected within the range specified in Table 5.1.

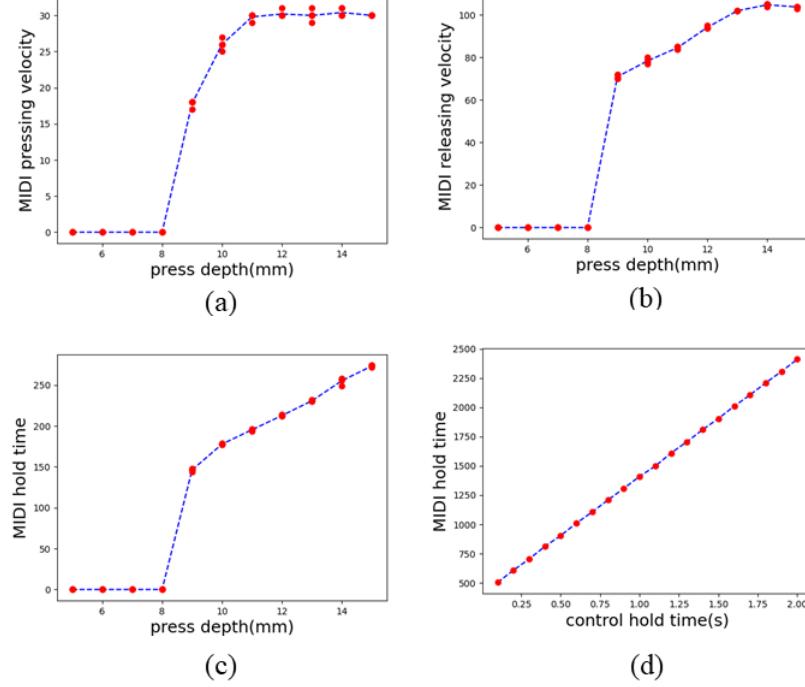


Figure 5.1: Pre-experiment results: (a,b,c) rigid finger experiments with changing pressing depth and fixed finger angle 0° , UR5 down velocity $0.1m/s$, UR5 up velocity $0.1m/s$, key-finger distance $0.1m$, hold time $0s$, UR5 acceleration $1m/s^2$ (a) MIDI pressing velocity vs UR5 pressing depth; (b) MIDI releasing velocity vs UR5 pressing depth; (c) MIDI hold time vs UR5 pressing depth; (d) MIDI hold time vs UR5 hold time.

Control parameter	Rigid finger	Soft finger
UR5 down velocity	$0.01 \sim 0.5 m/s$	$0 \sim 0.06 m/s$
UR5 up velocity	$0.01 \sim 0.7 m/s$	$0 \sim 0.7 m/s$
UR5 acceleration	$0.05 \sim 2 m/s^2$	$0.05 \sim 2 m/s^2$
Key-finger distance	$0 \sim 0.05m$	$0 \sim 0.05m$
Key-finger angle	$0 \sim 90^\circ$	$0 \sim 90^\circ$
UR5 pressing depth	$0.008 \sim 0.015m$	$0.020 \sim 0.035m$
UR5 hold time	$0 \sim 2s$	$0 \sim 2s$
Soft finger vacuum pressure	N/A	$-80 \sim 0 kPa$

Table 5.1: Proper control parameter ranges located in pre-experiments. In data collection step, we will use control parameters within these ranges.

5.1.2 Rigid finger single note

Holding all other control parameters to be constant, the smoothing spline fits well the relationship between UR5 pressing velocity and the MIDI pressing velocity (Figure 5.2(a)). The range of possible MIDI pressing velocity is 0~127, but the maximum reached MIDI velocity is no higher than 121.

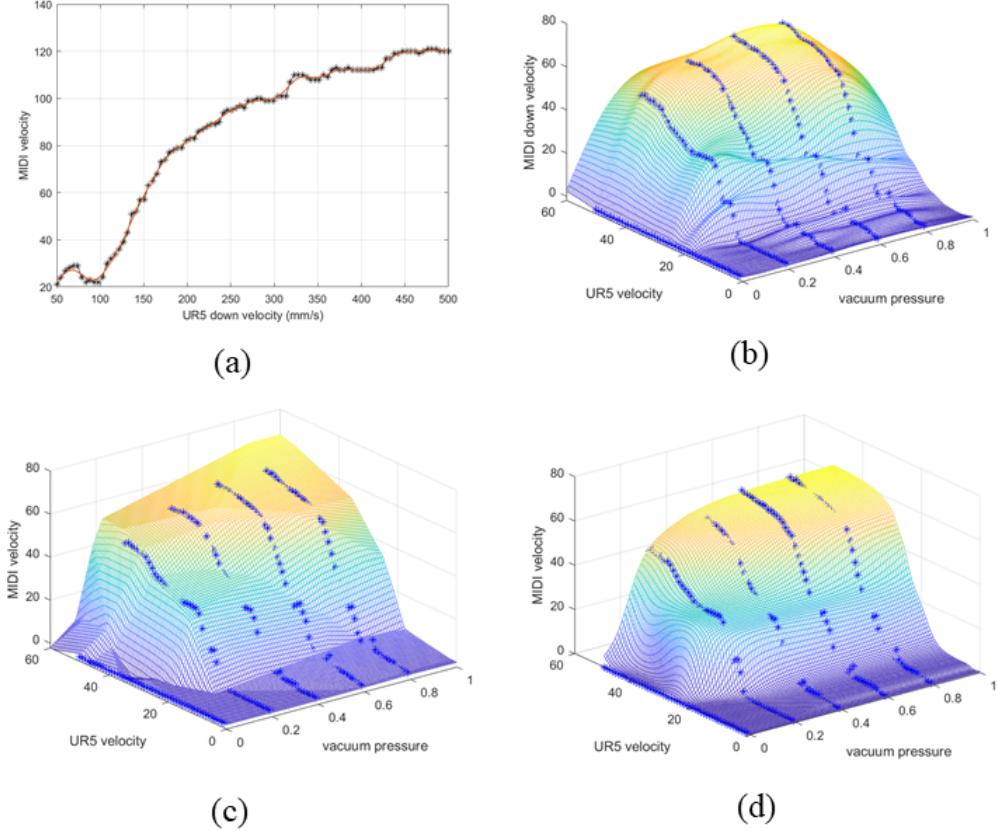


Figure 5.2: Modelling results: (a) rigid finger UR5 pressing velocity to MIDI pressing velocity; (b)(c)(d) soft finger vacuum pressure and UR5 pressing velocity to MIDI pressing velocity with (b) Gaussian Process (c) Non optimized densely connected neural network (d) Bayesian optimized densely connected neural network. UR5 velocity in unit mm/s. Vacuum pressure normalized by -100 kPa.

5.1.3 Soft finger single note

We used Gaussian Process and densely connect neural network (DCNN) to map the soft finger vacuum pressure and UR5 down velocity to MIDI pressing velocity. The models' detailed structure and their fitting performance are shown in Table 5.2.

Gaussian Process model yields the most accurate fitting with the detailed fluctuation in MIDI pressing velocity taken into account, resulting in a non-smooth model with several ridges parallel to vacuum pressure axis (Figure 5.2(b)). Models formed by DCNN shows a smoother fitting surface. Instead of trying to fit every single data point, the optimized DCNN focuses on the general changing trend of MIDI pressing velocity on UR5 down

velocity and vacuum pressure (Figure 5.2(d)). After Bayesian optimization on network layer size and activation function, the dimension of DCNN (3×291) exceeds the dimension of training data (3×250), yet overfitting does not occur and test loss decreases.

Model	Structure	Evaluate
Gaussian Process	$CovFunc = covSEard_1 + covSEard_2$	-loglikelihood=469.17
DCNN	[10 10], relu, linear	test loss=6.8415
Optimized DCNN	[3 291], tanh, linear	test loss=1.843

Table 5.2: Model performances on the soft finger. The structure column shows covariance function for Gaussian Process and fully connected layer size, activation function, output activation function for the Densely connected neural network

Fitting model formed by polynomial functions also shows interesting results. Figure 5.3 shows a polynomial model:

$$f = a + bx + cy + dx^2 + exy + fy^2 + gx^2y + hxy^2 + iy^3$$

,where f is MIDI pressing velocity, x is UR5 down velocity in mm/s, y is vacuum pressure normalized by -100 kPa, a, b, c, \dots are constants to be optimized using least square error. The polynomial fitting surface is the smoothest among all models in this section with degree 2 in x and degree 3 in y . The plotted residual of polynomial model prediction has a similar shape to sine function (Figure 5.4). The period of residual over different vacuum pressure appears to be similar ($P = \sim 20\text{mm/s}$).

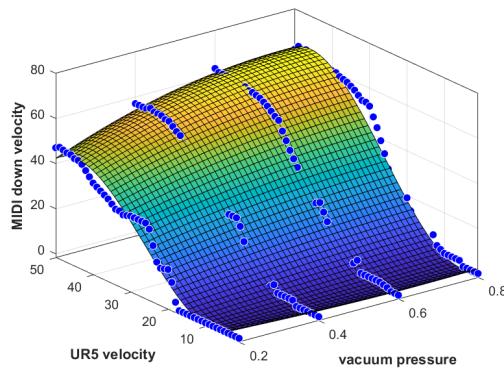


Figure 5.3: Polynomial model fit.

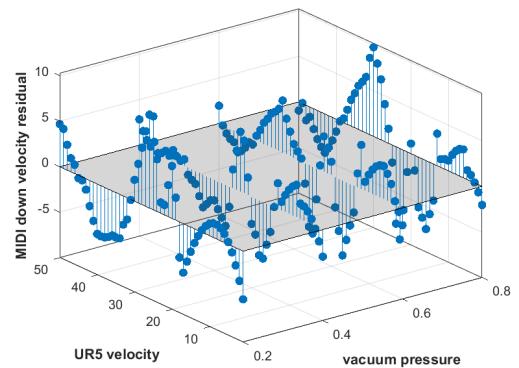


Figure 5.4: Residual of polynomial fit.

5.1.4 Rigid finger sequential notes

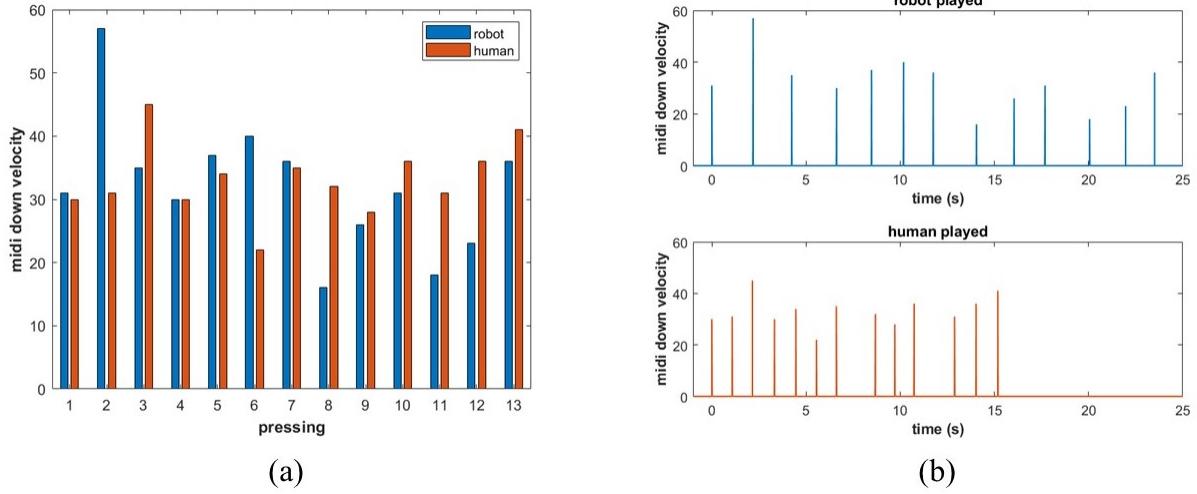


Figure 5.5: Rigid finger piano playing demo with individual keystrokes controlled by smoothing spline model.

To make robot piano playing demo, we first record the human played piece using one finger by MIDI message sequence. Then we find the control parameters the robot needs to replicate the recorded piece by inverting the fitting models above. We record the robot played music and compare it with the human played one (the target).

In our first demo, the time gap between notes played was not considered in robots' control parameter. Among the 13 keystrokes, the robot was able to replicate most of MIDI pressing velocities (Figure 5.5(a)). However, the robot's playing speed was significantly lower than human (Figure 5.5(b)).

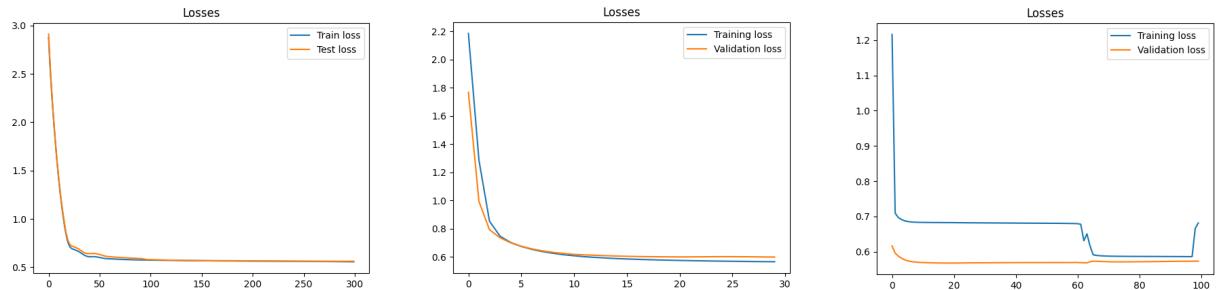


Figure 5.6: 3 layer RNN network

Figure 5.7: 1 layer LSTM with 12 hidden states

Figure 5.8: 10 layer LSTM with 64 hidden states

In the following few demos, we calculated the time gap between two pressings by subtracting their MIDI timestamps. We then set the moving time between two pressings to be the time gap in UR5 control. It showed good result in demonstrations, but was still slower than human playing since the time for keystrokes is not considered. As in demo we essentially map a sequence of MIDI to a sequence of control parameters, we

trained a LSTM neural network to manage the pressing timings together with other control parameters. Significant drop in losses is observed in training (Figure 5.6, 5.7, 5.8).

5.2 Mass-spring-damper Model

In this section, we look into the mass-spring-damper model’s performance in predicting soft finger keystroke. We first show how the model parameters (stiffness and damping coefficients) are identified, then we compare the model predicted keystroke in terms of stages, displacement and MIDI with the ground truth recorded by motion capture system.

5.2.1 System Identification

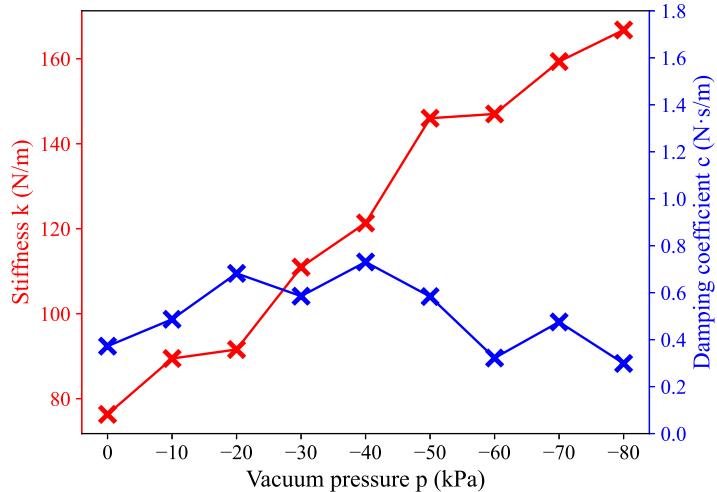


Figure 5.9: System identified parameters vs. varying vacuum pressure.

Soft finger	mass m_1	70 g
	stiffness k_1	$k_1 = -1.19p + 75.41 \text{ N/m}$
	damping coefficient c_1	0.50 $\text{N} \cdot \text{s/m}$
	mass m_2	34.05 g
Piano key	stiffness k_2	0.30 N/mm
	damping coefficient c_2	0.30 $\text{N} \cdot \text{s/m}$

Table 5.3: System parameters

To bridge the simulation-to-reality (sim2real) gap in predicting real-world piano keystrokes. It is critical to have the model parameters (i.e., spring stiffness k and damping coefficient c) as close to the ground truth as possible. Therefore, we used system identification approach [12] to capture the stiffness of the particle jamming finger under different vacuum conditions, and the stiffness of the piano key. The finger stiffness is estimated by plotting the displacement of fingertip under different applied forces. The piano key stiffness

is measured similarly by plotting the displacement of piano key surface under different pressing force applied by increasing weights. The damping coefficients of finger and piano key are estimated by examining the captured keystroke waveform using the logarithmic decrement method [59] and natural frequency. The results of the system identification are shown in Table 5.3 and Figure 5.9.

5.2.2 Frequency Response

Two metrics are commonly used to evaluate robotic piano playing performance: articulation and dynamics, which describe the timing and loudness of sequential keystrokes. Tempo control necessitates the robot performing keystrokes with varied velocity and frequency. As a result, analyzing the frequency response of the finger-piano interaction is significant since piano playing with a resonant frequency will make the robot extremely difficult to control.

The frequency response can be investigated by feeding system-identified parameters into the proposed state-space model. The frequency response of the system with finger parameters under a vacuum pressure of -60kPa is shown in Fig. 5.10, where x_2 is the output and x_0 is the input. It can be shown that the system's resonant frequency is 38.5Hz, which makes controlling the soft finger problematic. Instead, the robot performs well with a keystroke frequency ranging from 10^{-1} Hz to nearly 20Hz because there is no noticeable lead or lag between the key displacement and the input of UR5 displacement. It demonstrates that the proposed state-space model is promising for investigating the available piano playing hitting frequency and system limitations prior to real-world design, construction, and experimentation of a piano playing robot.

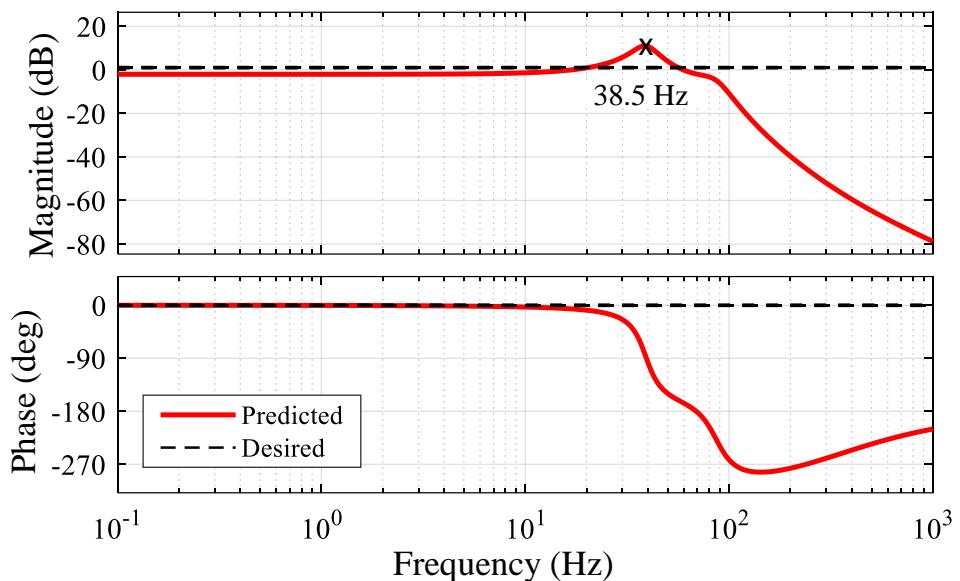


Figure 5.10: Frequency response of the state-space model.

5.2.3 Keystroke Stage Prediction

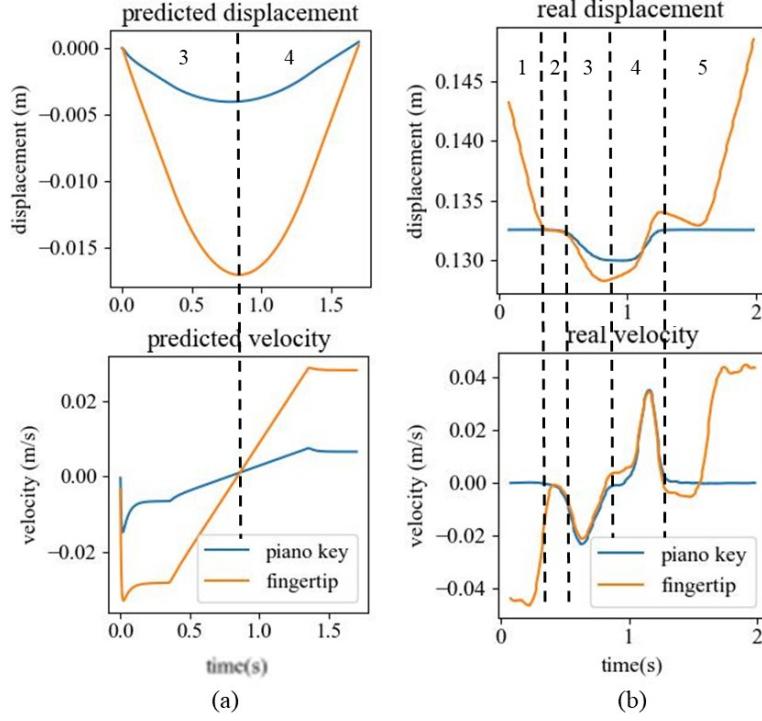


Figure 5.11: Keystroke stages comparison of (a) model predicted and (b) ground truth.

Figure 5.11 shows the displacement and velocity of piano key and soft fingertip during a keystroke. We divides the process of real keystroke into five different stages according to the relative movement of soft finger and piano key:

1. Soft finger is approaching the piano key but they have not been in contact yet. The soft finger moves downwards in a constant velocity controlled by UR5 robot arm. The piano key is stationary.
2. Soft finger gets in contact with the piano key. The finger decelerates sharply and the key accelerates slightly to move in a synchronous manner. The vertical displacement shows little change as the fingertip slip forwards in this stage.
3. Soft finger has slipped to its limit and starts to move downwards together with the piano key. They experience an accelerate and decelerate process to reach the maximum pressing depth.
4. Soft finger has reached the bottom of key or the lowest limit of pressing depth set in UR5 control command. The finger-key system bounces up back by first accelerating and then decelerating.
5. Piano key is back to its original position. Soft finger detaches from the contact with piano key and moves upwards at the releasing velocity controlled by UR5 robot arm.

The mismatch of displacement and velocity during contact phase (stage 2,3,4) might be attributed to the deformation of soft finger under interaction with the piano key. The soft finger is also observed to be stick to the key surface for a short time during the releasing phase (stage 4,5) due to the property of skin material silicone, which might disturb the keystroke observation as well.

The model predicted keystroke includes two stage only, which corresponds to the stage 3 and 4 in real keystroke: Soft finger moves down and up together with the piano key. This is expected as the mass-spring-damper system models the finger-key system as a whole. So only the contact phase is included in our model.

Mass-spring-damper model performs well in predicting the displacement in a keystroke. The accelerate and decelerate process of the soft finger and piano key is simulated as well. Though the final velocity of piano key at the end of stage 4 is not back to zero, it does not affect our prediction on MIDI using piano key movement, as demonstrated in section 5.3.5.

5.2.4 Displacement Prediction

In this subsection, we compare the model predicted displacement with the ground truth when the pressing depth is set small (7 mm). To validate the proposed model, we employed the same control parameters in the model-based simulation as in the ground truth. The x_0 and UR5 displacements, respectively, provide input to the model-based and real-world keystroke systems. Two groups of vacuum pressure were tested for a single-note keystroke action. The UR5 end-effector descends 7mm at a speed of 10mm/s for the keystroke. When it reaches the designed position, it remains stationary for 1 second, during which time we analyzed the steady states of the two-dimensional output.

As illustrated in Fig. 5.12, it can be seen that the model is able to simulate all the 5 stages of the keystroke action. It is noticeable that the damped behavior of the model matches with the ground truth. Both the simulation and ground truth reaches their steady states before the release stage. In addition, we can see from the image that as finger stiffness increased, both the key and soft finger displayed a bigger displacement in their vertical direction. The similar stiffness-varying behavior have been seen in the ground-truth results.

To determine how much of the original mechanical qualities are maintained following reduced-order modeling of the real-world keystroke action. We assess the performance of two models in representing vertical movement with varying stiffness quantitatively by defining the error rate as

$$\sigma = \frac{|r_{gt} - r^*|}{r^*} \times 100\% \quad (5.1)$$

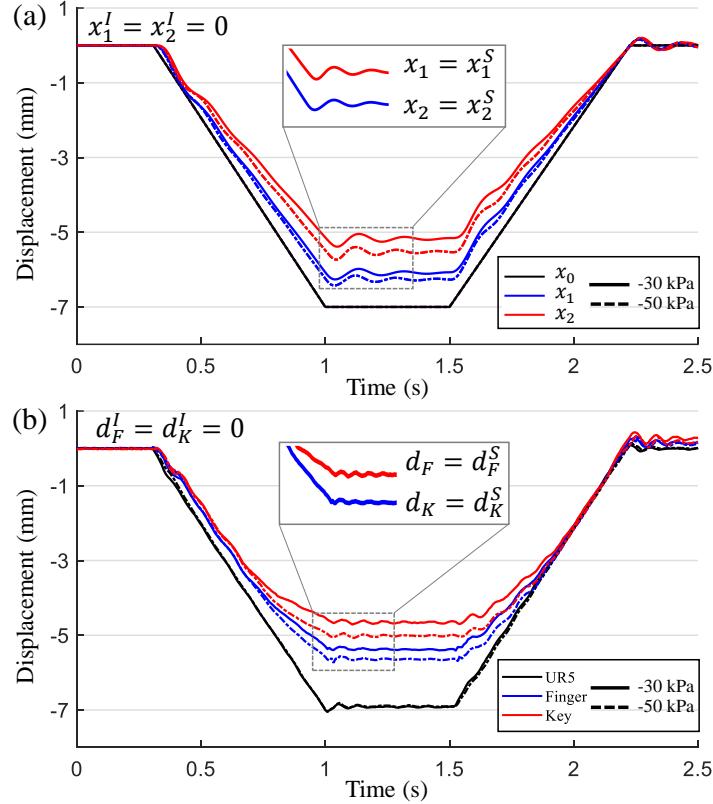


Figure 5.12: Comparison of steady states between the (a) analytical model and (b) ground truth.

where

$$\left\{ \begin{array}{l} r_{gt} = \frac{d_F^S - d_F^I}{d_K^S - d_K^I} \\ r^* = \frac{x_1^S - x_1^I}{x_2^S - x_2^I} \end{array} \right. \quad (5.2)$$

In Eq. 5.2, d_F and d_K denote the states of the finger and key, respectively, whereas the superscripts S and I denote the steady and initial states. As a result, the deformation ratio r represents the relative displacement of the finger and key.

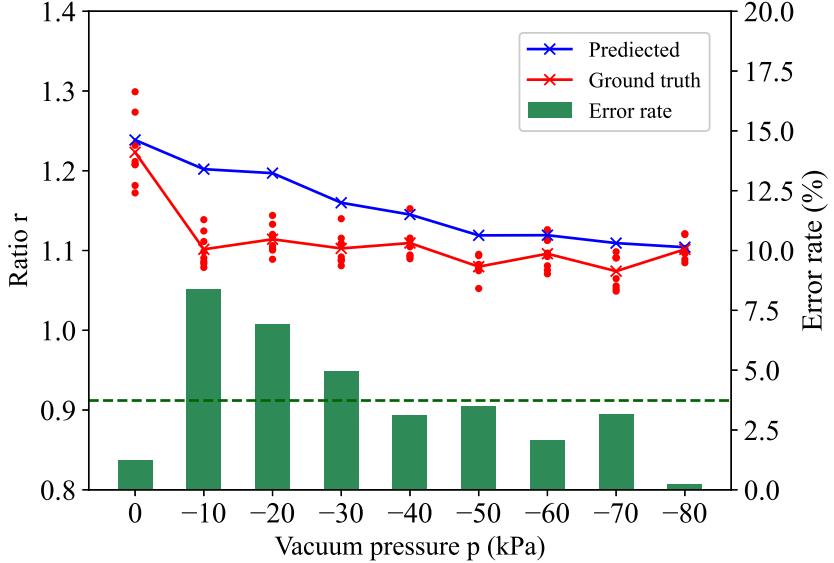


Figure 5.13: Deformation ratio and error rate under varying stiffness conditions.

To validate the analytical reduced-order model in representing real-world keystroke action under various stiffness circumstances, we employed the same input as in Fig. 5.12 and compared the results of both scenarios. The predicted deformation ratio r^* , ground truth r_{gt} , and error rate σ under 9 stiffness groups are shown in Fig. 5.13. As can be observed, the model's predicted deformation ratio is close to the ground truth. The overall average error rate in characterizing the deformation behavior of the finger and the piano key is 3.73%, suggesting that the model achieves good accuracy in describing the steady-states of the non-linear finger-key interaction system.

5.2.5 MIDI Message Prediction

In this subsection, we aim to use the analytical mass-spring-damper model to predict the MIDI messages as this will directly influence the quality of the produced music (loudness and tempo). We exploited the motion control of UR5 robot arm (x_0) with a range of pressing velocity ($10 \sim 80 \text{mm/s}$) as the model input and analyzed the piano key displacement x_2 . As MIDI message generation is directly related to the piano key movement at triggering threshold, we use the piano key velocity at triggering threshold to represent predicted MIDI pressing velocity, and the time gap between triggering threshold in pressing and releasing to represent predicted MIDI hold time (Figure 4.5). Due to the unstable MIDI generation by digital piano, we compare the predicted results directly with the measured piano key movement at triggering threshold in real world experiments.

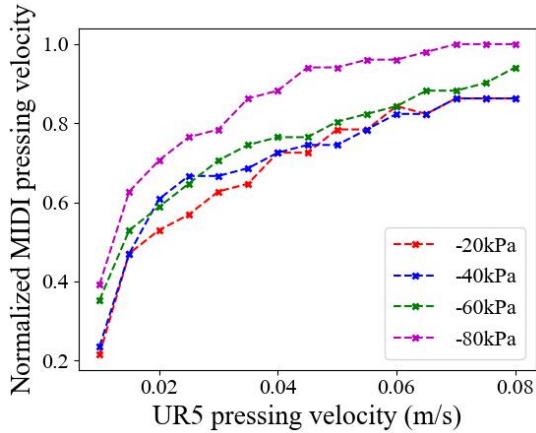


Figure 5.14: Measured MIDI pressing velocity under changing finger stiffness.

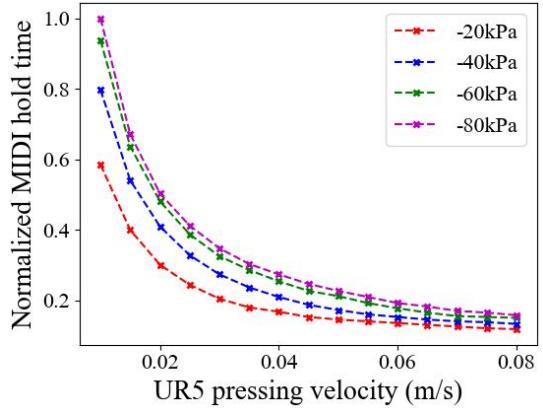


Figure 5.15: Measured MIDI hold time under changing finger stiffness.

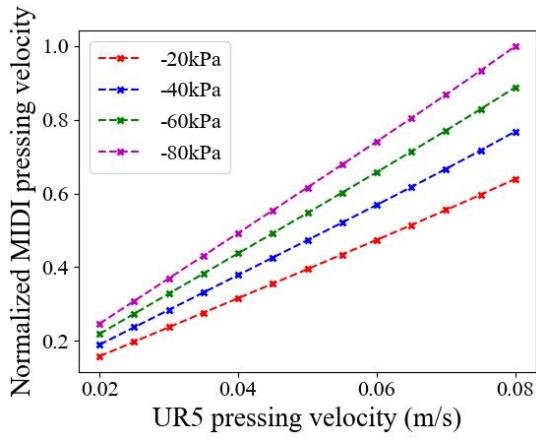


Figure 5.16: Model predicted MIDI pressing velocity.

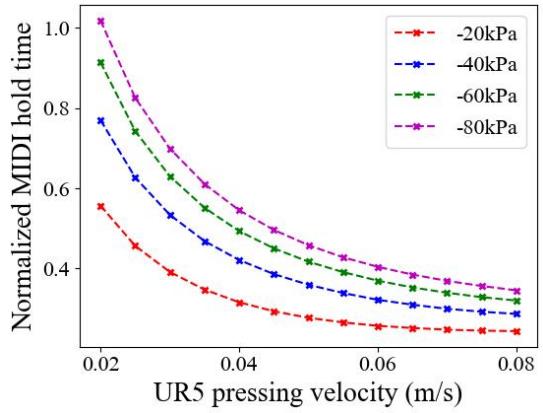


Figure 5.17: Model predicted MIDI hold time under changing finger stiffness.

The proposed model was validated to represent the music production with reasonable fidelity. Let Vel_{on} denote the normalized MIDI pressing velocity, Figure 5.14 reveals that Vel_{on} increases if the UR5 end-effector descends with a higher speed. It is noticeable that when the pressing velocity is fixed, results from both the model and ground truth witnessed a rise of Vel_{on} under a higher vacuum pressure (Figure 5.14, 5.16). This is because the keystroke action becomes more powerful as stiffness increases. A stiff finger is able to provide the piano key with a higher acceleration since the non-linearity and hysteresis raised by soft materials are reduced. The mass-spring-damper model also succeeds in predicting the increasing trend of normalized MIDI hold time with the increasing vacuum pressure. The feature that MIDI hold time increasing at a lower speed as vacuum pressure approaches -80 kPa is captured by the model, shown as the decreasing gap between hold time lines. (Figure 5.15, 5.17)

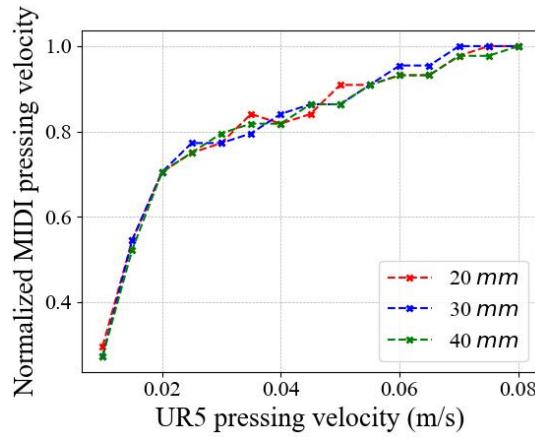


Figure 5.18: Measured MIDI pressing velocity under changing pressing depth.

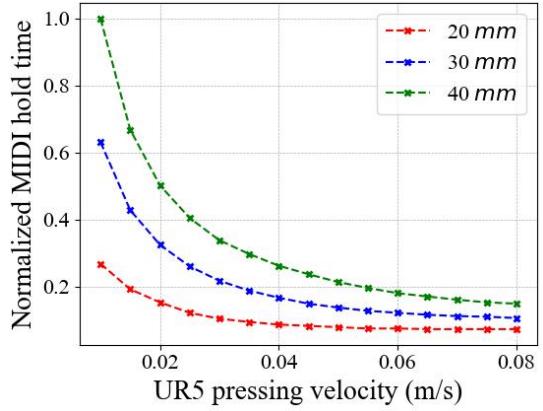


Figure 5.19: Measured MIDI hold time under changing pressing depth.

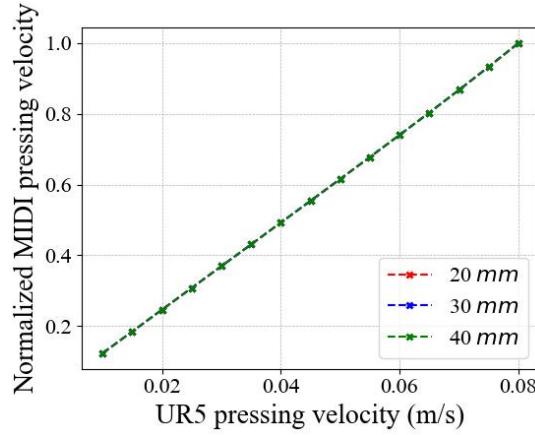


Figure 5.20: Model predicted MIDI pressing velocity.

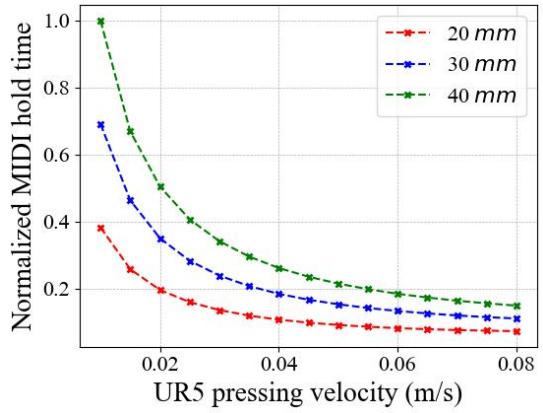


Figure 5.21: Model predicted MIDI hold time under changing pressing depth.

With fixed pressing velocity, normalized MIDI pressing velocity does not show much difference for various pressing depth in both the model and ground truth (Figure 5.18, 5.20). The model predicts the increasing trend of normalized MIDI hold time with the increasing pressing depth (Figure 5.19, 5.21).

Chapter 6

Discussion and Conclusion

In this chapter, we start with a comparison of rigid and soft robotic finger in piano playing. We then discuss about the model selecting process based on keystroke motion analysis. Next we analyse the strengths and drawbacks of the mass-spring-damper model serving as a reduced-order internal model in soft finger keystroke control. At last, we give a summary of the project work and talk about the possible future improvement to the soft robotic hand, the model and the control strategy.

6.1 Rigid vs Soft Finger

The control of rigid finger is by far more straightforward and simpler than the soft particle jamming finger, due to the rigid transformation of movement from the UR5 robot arm to the fingertip. However, rigid fingers are easier to be broken during the data collection process when thousand of keystrokes are performed. The sound of collision between the rigid finger and the piano key persists throughout the experiments, which is a huge drawback for music playing. Rigid finger also fails to give us variation in robotic properties, limiting the model's adaptability for robot body changes. Soft finger provides the variation in stiffness achieved by pneumatic methods, which serves as a robotic simulation to human body stiffness changes and leads to the development of more advanced adaptive models.

6.2 Model Selection

We proposed several different models for soft particle jamming finger piano keystroke, including learning based models and a variety of physical models as elaborated in below.

Learning-based models such as DCNN and GP gives us a simple solution for control model when there is enough keystroke data for training. While achieving accurate description of the relationship between control parameters and generated MIDI, the learning-

based model fails to capture the stochastic nature of soft finger in piano keystroke process. The adaptation of model to soft finger or environment change (like a new piano) is possible but requires long-term training on newly collected keystroke data. The complexity structure of neural network models makes them non-analytical, although no obvious effect of computation delay is observed in robot control using learning-based models in this project.

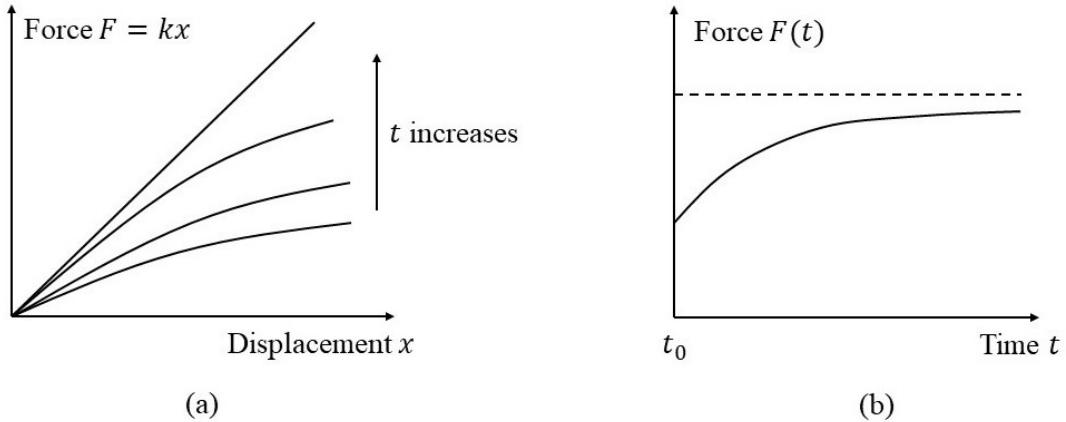


Figure 6.1: Hysteresis spring model. (a) Force-displacement plot of a hysteresis spring. Spring stiffness k is a function of both displacement and time, i.e. $k(x, t)$. (b) Force-time plot of a hysteresis spring compressed to x_0 at time t_0 and held at x_0 afterwards.

A hysteresis in soft finger and piano key interaction is observed in almost every single keystroke recorded by 3D motion capture cameras: when the soft finger moves downwards to be in contact with the piano key, the piano key is not pressed down immediately but only after a short moment, as shown by keystroke stage 2 in Figure 5.11). We wonder if this phenomenon is caused by the slow release of elastic potential energy stored in soft silicone hand skin. We proposed the hysteresis spring model (Figure 6.1) to describe this phenomenon. However, after a verification experiment in which we measured the force change over time when pressing the soft finger onto a rigid surface, we found that the hysteresis in soft finger force reaction is negligible. The hysteresis spring model is thus not applicable to our soft particle jamming finger. However, it might be interesting in modelling the behaviour of other soft porous materials with better ability in absorbing and releasing elastic energy, such as memory foam.

We also considered modelling the soft robotic finger in an anthropomorphic way with each rigid bone represented as a stick and connected by springs and dampers at joints (Figure 6.2 (a)) [12]. While this model is able to capture the dynamics of soft finger skeleton and joints, it neglects the piano key which is an essential part of the interaction system. In addition, the equations of motion become tricky to derive with multiple rotational springs and dampers, which might lead to over complicated state-space matrices.

At last, we decided to use mass-spring-damper system as our soft finger keystroke

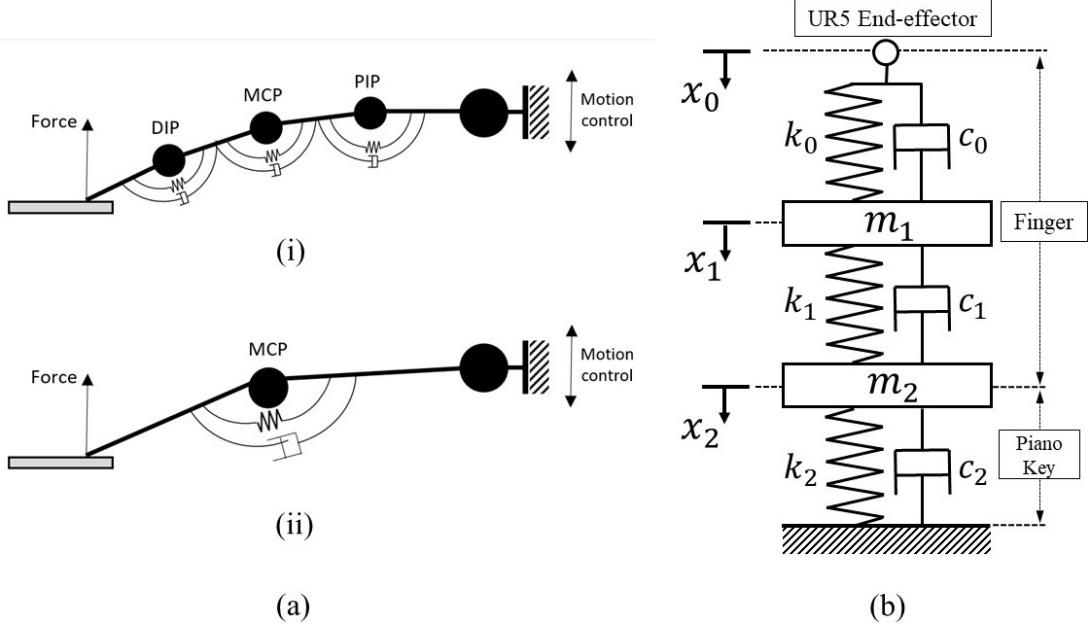


Figure 6.2: Reduced-order physical models. (a) (i) Anthropomorphic finger model from [12] and (ii) simplified anthropomorphic finger model. (b) Mass-spring-damper model.

model (Figure 6.2 (b)). It neglects the internal deformation of soft finger during the interaction with piano keys, while keeping the UR5 robot arm movement and the piano key movement which are fundamentally important for keystroke control. We successfully derived the state-space model for mass-spring-damper system, and used it as the analytical solution to soft robotic finger piano keystroke.

6.3 Mass-spring-damper Model Analysis

The mass-spring-damper model can describe the contact phases of soft particle jamming finger piano keystroke. It can also accurately predict the changing trend of piano key movement with UR5 pressing velocity, vacuum pressure and UR5 pressing depth in terms of normalized MIDI pressing velocity and hold time as demonstrated in section 5.2.5. While the mass-spring-damper system succeeds in modelling the exponential decay trend of normalized MIDI hold time, it fails in describing the deceleration of normalized MIDI pressing velocity increase with UR5 pressing velocity and shows a linear relationship of them. This might be due to the over simplification of model regarding the complex soft finger structure and hybrid material.

6.4 Summary of Findings

In this project, we proposed a reduced-order, analytical mass-spring-damper model which can depict real-world keystroke action of soft particle jamming finger in piano

playing scenario.

We started by fabricating an anthropomorphic robotic finger with a hybrid structure: rigid skeleton, soft silicone skin and granule particle chambers at joints. We controlled the stiffness of the soft finger by varying the vacuum pressure and performed 2160 keystrokes with UR5 control command, digital piano generated MIDI message, and 3D motion of finger-key system recorded.

We first tried applying a learning-based model to relate the control parameters (vacuum pressure and UR5 control command) to audio MIDI message. Methods used include densely connected neural network, Gaussian Process and polynomial fit. While succeeded in acquiring accurate modelling performance from control input to MIDI output, the data-driven models failed to capture the stochastic nature of soft robot. They are complex in structural and computationally expensive. Adaption to change in soft robot body or the external environment requires training by large amount of extra keystroke data. The computation cost and slow adaption limit the usage of learning-based model in feedforward soft finger keystroke control.

Next we turned to look into a reduced-order physical model, a second-order mass-spring-damper system, as a representation of soft-bodied finger-piano interaction. The spring stiffness and damping coefficient in the mass-spring-damper model were acquired from direct measurement of finger-key system. A state-space model derived from the mass-spring-damper model, which is lightweight and mathematically analytical, is used to represent the dynamic behaviors of piano keystrokes under varying stiffness conditions. The experimental results reveals that the presented model accurately depicts the passive movements of the finger-piano interaction in terms of keystroke stage and MIDI message prediction. The model has advantages in computation efficiency and adaptability, and thus serve as an ideal internal model for soft robotic finger piano keystroke feedforward control.

In conclusion, this work demonstrated that the behaviours of high-order, non-linear soft finger-piano key interactions can be modelled as a linear mass-spring-damper system. The computation efficiency and high adaptability to finger stiffness and piano key stiffness changes makes the mass-spring-damper model an ideal choice as the internal model for model-based feedforward keystroke control. By involving the external environment (piano key) as a part of adaptive model, this project creatively migrate the idea of feedforward control in neuroscience to soft robotic realisation.

6.5 Limitation and Future Work

The major limitations of this project lie in three aspects: robot, model and control designs, which are elaborated in following three subsections.

6.5.1 Beyond Monophonic Piano Playing

The experiments done in this project focus on single keystroke completed by one soft finger. While the soft particle jamming finger is able to play monotonic songs with accurate keystroke, it performs poorly in terms of timing since the finger needs to lift up from the current key in order to press the next one. The single finger design also makes it impossible to press two keys simultaneously. To enable better timing and polyphonic piano playing, we need a soft robotic hand with multiple fingers and independent actuation implemented for each finger. Implementation of particle jamming in each separate finger will add difficulty in the hand design and fabrication. The control model needs to consider the influence of adjacent finger movements. Collision avoidance and trajectory planning needs to be taken account for. Different fingers, such as index and little finger, might have different strength in performing keystrokes, which needs to be considered in motion planning too.

6.5.2 Adaptive Feedforward Internal Model

The adaptability of the mass-spring-damper model is shown by the easily tunable stiffness of the finger spring (adapt to the soft robot change) and the piano key (adapt to the environmental change). However, the variable stiffness is not likely to cover all aspects of property changes in either soft finger or the piano key. We might add damping coefficient as another changeable feature, improving the model accuracy at a cost of computational complexity.

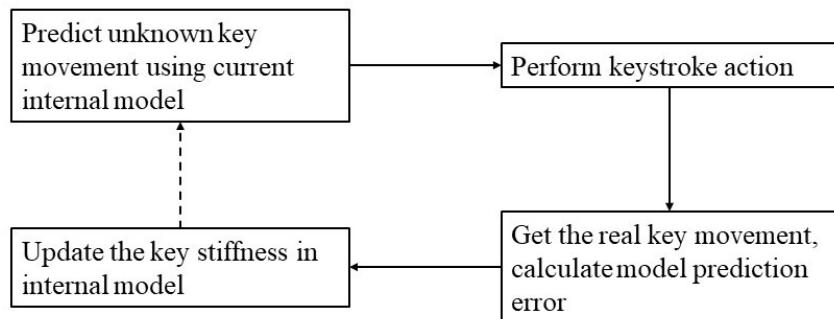


Figure 6.3: How the piano key stiffness gets updated in an adaptive mass-spring-damper model.

In this project, we acquire the stiffness of piano key and soft finger under different vacuum pressure by measurement. Due to the feature of soft materials, the property of soft finger is observed to be highly unstable. The stiffness under the same vacuum pressure is usually not fixed after an inflation and deflation process. In this case, it is not accurate to use a constant measured stiffness for all keystrokes represented by the model. In the future, we could add a learning mechanism to allow self-adjustment of stiffness

according to the force and position feedback information. Ideally, when the soft robot encounter with a new piano key with unknown stiffness, it will be able to perform desired keystroke actions by deducing the key stiffness after a few trials and error. When the soft finger is pressing a structure with known stiffness, it will be able to adjust its own stiffness value according to various forms of feedback.

6.5.3 Feedback Control

A controller may achieve better keystroke performance via a combination of feedback and feedforward mechanisms: usually, the feedforward component provides a prediction of the movement's outcome, while the feedback component compensates for the estimation inaccuracies and stabilizes the system about the desired behavior [60]. In this project, only the feedforward component is used for control. The error in MIDI prediction might partially be attributed to the absence of feedback mechanism.

For future experiments, we can integrate stress sensor to the soft finger to provide real-time force feedback. The real-time position of soft finger, robot arm and piano key captured by cameras may also be used as a form of visual feedback. The digital MIDI message output will serve as an equivalent to audio feedback in the piano keystroke task. However, considering that MIDI is only produced at the exact moment of key pressing, the audio feedback may not directly affect the current keystroke action, but rather influence the following keystrokes, which is essentially a part of the training process.

Chapter 7

Bibliography

- [1] Angelica Lim, Tetsuya Ogata, and Hiroshi G Okuno. Towards expressive musical robots: a cross-modal framework for emotional gesture, voice and music. *EURASIP Journal on Audio, Speech, and Music Processing*, 2012(1):1–12, 2012.
- [2] Shinichi Furuya and Eckart Altenmüller. Flexibility of movement organization in piano performance. *Frontiers in human neuroscience*, 7:173, 2013.
- [3] Reza Shadmehr and Ferdinando A Mussa-Ivaldi. Adaptive representation of dynamics during learning of a motor task. *Journal of neuroscience*, 14(5):3208–3224, 1994.
- [4] Chin-Shyurng Fahn, Cheng-Feng Tsai, and Yung-Wei Lin. Development of a novel two-hand playing piano robot. In *Proceedings of the 4th International Conference of Control, Dynamic Systems, and Robotics (CDSR'17)*. CDSR, 2017.
- [5] Dan Zhang, Jianhe Lei, Beizhi Li, Daniel Lau, and Craig Cameron. Design and analysis of a piano playing robot. In *2009 International Conference on Information and Automation*. IEEE, 2009.
- [6] Anthony Topper, Thomas Maloney, Scott Barton, and Xiangnan Kong. Piano-playing robotic arm. 2019.
- [7] Ichiro Kato, Sadamu Ohtera, Katsuhiko Shirai, Toshiaki Matsushima, Seinosuke Narita, Shigeki Sugano, Tetsunori Kobayashi, and Eizo Fujisawa. The robot musician ‘wabot-2’ (waseda robot-2). *Robotics*, 3(2):143–155, 1987. Special Issue: Sensors.
- [8] Yen-Fang Li and Chi-Yi Lai. Intelligent algorithm for music playing robot—applied to the anthropomorphic piano robot control. In *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, pages 1538–1543. IEEE, 2014.
- [9] Yen-Fang Li and Chi-Yi Lai. Intelligent algorithm for music playing robot — applied to the anthropomorphic piano robot control. In *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, pages 1538–1543, 2014.
- [10] Luca Scimeca, Cheryn Ng, and Fumiya Iida. Gaussian process inference modelling of dynamic robot control for expressive piano playing. *Plos one*, 15(8):e0237826, 2020.
- [11] JAE Hughes, P Maiolino, and Fumiya Iida. An anthropomorphic soft skeleton hand exploiting conditional models for piano playing. *Science Robotics*, 3(25):eaau3098, 2018.
- [12] Huijiang Wang, Toby Howison, Josie Hughes, Arsen Abdulali, and Fumiya Iida. Data-driven simulation framework for expressive piano playing by anthropomorphic hand with variable passive properties. *2022 5th IEEE-RAS International Conference on Soft Robotics (RoboSoft2022)*, In press, 2022. <https://doi.org/10.17863/CAM.81805>.

- [13] Tao Du, Josie Hughes, Sebastien Wah, Wojciech Matusik, and Daniela Rus. Underwater soft robot modeling and control with differentiable simulation. *IEEE Robotics and Automation Letters*, 6(3):4994–5001, 2021.
- [14] Nathaniel N Goldberg, Xiaonan Huang, Carmel Majidi, Alyssa Novelia, Oliver M O'Reilly, Derek A Paley, and William L Scott. On planar discrete elastic rod models for the locomotion of soft robots. *Soft robotics*, 6(5):595–610, 2019.
- [15] Josie Hughes, Utku Culha, Fabio Giardina, Fabian Guenther, Andre Rosendo, and Fumiya Iida. Soft manipulators and grippers: a review. *Frontiers in Robotics and AI*, 3:69, 2016.
- [16] James M Bern, Yannick Schnider, Pol Banzet, Nitish Kumar, and Stelian Coros. Soft robot control with a learned differentiable model. In *2020 3rd IEEE International Conference on Soft Robotics (RoboSoft)*, pages 417–423. IEEE, 2020.
- [17] Federico Renda, Michele Giorelli, Marcello Calisti, Matteo Cianchetti, and Cecilia Laschi. Dynamic model of a multibending soft robot arm driven by cables. *IEEE Transactions on Robotics*, 30(5):1109–1122, 2014.
- [18] Mahdi Ilami, Hosain Bagheri, Reza Ahmed, E Olga Skowronek, and Hamid Marvi. Materials, actuators, and sensors for soft bioinspired robots. *Advanced Materials*, 33(19):2003139, 2021.
- [19] Jorge Solis, Kenichiro Ozawa, Maasaki Takeuchi, Takafumi Kusano, Shimpei Ishikawa, Klaus Petersen, and Atsuo Takanishi. Biologically-inspired control architecture for musical performance robots. *International Journal of Advanced Robotic Systems*, 11(10):172, 2014.
- [20] S Furuya and H Kinoshita. Expertise-dependent modulation of muscular and non-muscular torques in multi-joint arm movements during piano keystroke. *Neuroscience*, 156(2):390–402, 2008.
- [21] Werner Goebel. Movement and touch in piano performance. *Handbook of human motion*, pages 1–18, 2017.
- [22] Masaya Hirashima, Kazutoshi Kudo, Koji Watarai, and Tatsuyuki Ohtsuki. Control of 3d limb dynamics in unconstrained overarm throws of different speeds performed by skilled baseball players. *Journal of neurophysiology*, 97(1):680–691, 2007.
- [23] Masaya Hirashima, Kazutoshi Kudo, and Tatsuyuki Ohtsuki. Utilization and compensation of interaction torques during ball-throwing movements. *Journal of Neurophysiology*, 89(4):1784–1796, 2003.
- [24] Paul L Gribble and David J Ostry. Compensation for interaction torques during single-and multijoint limb movement. *Journal of neurophysiology*, 82(5):2310–2326, 1999.
- [25] Klaus Schneider, Ronald F Zernicke, Richard A Schmidt, and Timothy J Hart. Changes in limb dynamics during the practice of rapid arm movements. *Journal of biomechanics*, 22(8-9):805–817, 1989.
- [26] NF Marconi and GL Almeida. Principles for learning horizontal-planar arm movements with reversal. *Journal of Electromyography and Kinesiology*, 18(5):771–779, 2008.
- [27] RL Sainburg, C Ghez, and D Kalakanis. Intersegmental dynamics are controlled by sequential anticipatory, error correction, and postural mechanisms. *Journal of neurophysiology*, 81(3):1045–1056, 1999.
- [28] Nikolai Bernstein. The co-ordination and regulation of movements. *The co-ordination and regulation of movements*, 1966.

- [29] Cheryn Ng. Gaussian process inference modelling of dynamic robot control for expressive piano playing, 2020.
- [30] G Runge, M Wiese, L Günther, and A Raatz. A framework for the kinematic modeling of soft material robots combining finite element analysis and piecewise constant curvature kinematics. In *2017 3rd International Conference on Control, Automation and Robotics (ICCAR)*, pages 7–14. IEEE, 2017.
- [31] Abhishek Gupta, Clemens Eppner, Sergey Levine, and Pieter Abbeel. Learning dexterous manipulation for a soft robotic hand from human demonstrations. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3786–3793. IEEE, 2016.
- [32] Yu She, Chang Li, Jonathon Cleary, and Hai-Jun Su. Design and fabrication of a soft robotic hand with embedded actuators and sensors. *Journal of Mechanisms and Robotics*, 7(2), 2015.
- [33] Sylvain Abondance, Clark B Teeple, and Robert J Wood. A dexterous soft robotic hand for delicate in-hand manipulation. *IEEE Robotics and Automation Letters*, 5(4):5502–5509, 2020.
- [34] Daniela Rus and Michael T Tolley. Design, fabrication and control of soft robots. *Nature*, 521(7553):467–475, 2015.
- [35] Nina R Sinatra, Clark B Teeple, Daniel M Vogt, Kevin Kit Parker, David F Gruber, and Robert J Wood. Ultragentle manipulation of delicate structures using a soft robotic gripper. *Science Robotics*, 4(33):eaax5425, 2019.
- [36] Glenn K Klute, Joseph M Czerniecki, and Blake Hannaford. Mckibben artificial muscles: pneumatic actuators with biomechanical intelligence. In *1999 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (Cat. No. 99TH8399)*, pages 221–226. IEEE, 1999.
- [37] Kevin C Galloway, Panagiotis Polygerinos, Conor J Walsh, and Robert J Wood. Mechanically programmable bend radius for fiber-reinforced soft actuators. In *2013 16th International Conference on Advanced Robotics (ICAR)*, pages 1–6. IEEE, 2013.
- [38] Benjamin Shih, Dylan Drotman, Caleb Christianson, Zhaoyuan Huo, Ruffin White, Henrik I Christensen, and Michael T Tolley. Custom soft robotic gripper sensor skins for haptic object visualization. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 494–501. IEEE, 2017.
- [39] John Morrow, Hee-Sup Shin, Calder Phillips-Grafflin, Sung-Hwan Jang, Jacob Torrey, Riley Larkins, Steven Dang, Yong-Lae Park, and Dmitry Berenson. Improving soft pneumatic actuator fingers through integration of soft sensors, position and force control, and rigid fingernails. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5024–5031. IEEE, 2016.
- [40] Mariangela Manti, Vito Cacucciolo, and Matteo Cianchetti. Stiffening in soft robotics: A review of the state of the art. *IEEE Robotics & Automation Magazine*, 23(3):93–106, 2016.
- [41] Federico Carpi, Gabriele Frediani, Carlo Gerboni, Jessica Gemignani, and Danilo De Rossi. Enabling variable-stiffness hand rehabilitation orthoses with dielectric elastomer transducers. *Medical engineering & physics*, 36(2):205–211, 2014.
- [42] Koichi Suzumori, Shuichi Wakimoto, Kenta Miyoshi, and Kazuhiro Iwata. Long bending rubber mechanism combined contracting and extending fluidic actuators. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4454–4459. IEEE, 2013.
- [43] M Henke and G Gerlach. A multi-layered variable stiffness device based on smart form closure actuators. *Journal of Intelligent Material Systems and Structures*, 27(3):375–383, 2016.

- [44] ME Cates, JP Wittmer, J-P Bouchaud, and Ph Claudin. Jamming, force chains, and fragile matter. *Physical review letters*, 81(9):1841, 1998.
- [45] Yang Yang, Yingtian Li, and Yonghua Chen. Principles and methods for stiffness modulation in soft robot design and development. *Bio-Design and Manufacturing*, 1(1):14–25, 2018.
- [46] Eric Brown, Nicholas Rodenberg, John Amend, Annan Mozeika, Erik Steltz, Mitchell R Zakin, Hod Lipson, and Heinrich M Jaeger. Universal robotic gripper based on the jamming of granular material. *Proceedings of the National Academy of Sciences*, 107(44):18809–18814, 2010.
- [47] Nadia G Cheng, Maxim B Lobovsky, Steven J Keating, Adam M Setapen, Katy I Gero, Anette E Hosoi, and Karl D Iagnemma. Design and analysis of a robust, low-cost, highly articulated manipulator enabled by jamming of granular media. In *2012 IEEE international conference on robotics and automation*, pages 4328–4333. IEEE, 2012.
- [48] Eulalie Coevoet, Adrien Escande, and Christian Duriez. Optimization-based inverse model of soft robots with contact handling. *IEEE Robotics and Automation Letters*, 2(3):1413–1419, 2017.
- [49] Yahya Elsayed, Augusto Vincensi, Constantina Lekakou, Tao Geng, CM Saaj, Tommaso Ranzani, Matteo Cianchetti, and Arianna Menciassi. Finite element analysis and design optimization of a pneumatically actuating silicone module for robotic surgery applications. *Soft Robotics*, 1(4):255–262, 2014.
- [50] Frederick Largilliere, Valerian Verona, Eulalie Coevoet, Mario Sanz-Lopez, Jeremie Dequidt, and Christian Duriez. Real-time control of soft-robots using asynchronous finite element modeling. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2550–2555. IEEE, 2015.
- [51] Sander Tonkens, Joseph Lorenzetti, and Marco Pavone. Soft robot optimal control via reduced order finite element models. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 12010–12016. IEEE, 2021.
- [52] Wen-Bo Li, Wen-Ming Zhang, Hong-Xiang Zou, Zhi-Ke Peng, and Guang Meng. A fast rolling soft robot driven by dielectric elastomer. *IEEE/ASME Transactions on Mechatronics*, 23(4):1630–1640, 2018.
- [53] Michael W Hannan and Ian D Walker. Kinematics and the implementation of an elephant’s trunk manipulator and other continuum style robots. *Journal of robotic systems*, 20(2):45–63, 2003.
- [54] Thomas George Thuruthel, Yasmin Ansari, Egidio Falotico, and Cecilia Laschi. Control strategies for soft robotic manipulators: A survey. *Soft robotics*, 5(2):149–163, 2018.
- [55] Bryan A Jones and Ian D Walker. Kinematics for multisection continuum robots. *IEEE Transactions on Robotics*, 22(1):43–55, 2006.
- [56] Kaspar Althoefer, Jelizaveta Konstantinova, and Ketao Zhang. *Towards autonomous robotic systems*. Springer International Publishing, 2019.
- [57] Michele Giorelli, Federico Renda, Gabriele Ferri, and Cecilia Laschi. A feed-forward neural network learning the inverse kinetics of a soft cable-driven manipulator moving in three-dimensional space. In *2013 IEEE/RSJ international conference on intelligent robots and systems*, pages 5033–5039. IEEE, 2013.
- [58] Jorge Solis, Kei Suefuji, Koichi Taniguchi, Takeshi Ninomiya, Maki Maeda, and Atsuo Takanishi. Implementation of expressive performance rules on the wf-4riii by modeling a professional flutist performance using nn. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 2552–2557. IEEE, 2007.

- [59] D. J. Inman. *Upper Saddle*, page 43–48. Pearson Education, 2009.
- [60] Jean-Jacques E Slotine. The robust control of robot manipulators. *The International Journal of Robotics Research*, 4(2):49–64, 1985.

Appendix A

Risk Assessment Retrospective

The risk assessments completed at the start of the project discussed various potential hazards, which were safely avoided:

1. Physical injury caused by inappropriate robot operations: UR5 robot arm is set to work under safety mode with limited velocity and acceleration. Emergency stop is applied when UR5 is blocked by an obstacle. UR5 control codes are pre run before loading the robotic finger on UR5 to avoid potential damage caused by faulty codes.
2. Chemical hazards in soft material moulding: Latex gloves are used to avoid direct contact between skin and silicone liquid. Masks are worn to reduce inhalation of toxic gas. Soft finger fabrication is done under the supervision of PhD student for safety guarantee.
3. Electric shock or burn from soldering: No soldering operation is used in this project. Electric circuit for force sensing was powered by computer USB with low voltage, which ensures safety in electricity use.
4. Mechanical hazard in power tool usage: Tasks including power tools are carried out with proper clothing and eye protection. Safety inductions are given by senior lab members before students are involved in such tasks.
5. Fatigue from excessive computer use: Regular break are taken during the data analysing or paper working with long screen time. Lone work is avoided to maintain a collaborative and supportive working environment.

Appendix B

Additional Resource

B.1 Project Repository

The project repository includes full code for keystroke data collection and analysis, as well as documents including project reports, posters and a soft finger fabrication manual: https://github.com/zhangyuyi99/soft_robot_piano_playing.git

Part of keystroke data (UR5 control parameters and corresponding MIDI message) is included in *data* folder in Github repository. The 3D motion data captured by tracking cameras can be found *here*, including both original *.tak* files and the extracted *.csv* files.

B.2 Video Demonstrations

Link to keystroke showcase videos is *here*.

B.3 Log Book

A weekly record of progress can be found *here*, which includes summaries of weekly work, problems discussed at meetings and plans for the following week.

Acknowledgement

I would like to thank Huijiang Wang for patiently mentoring a research newbie through project planning, experiment design, and article writing. My special thanks goes to Prof Fumiya Iida for giving timely feedback and helpful advice at every Monday meeting. Thank you Fumiya for being a supportive and encouraging lab leader. Thank you also to all of BIRL lab members for the fruitful discussions and amazing projects that they exposed me to. I enjoyed a fantastic fourth year, thanks to your kind support and warm friendship.