A TRIDENT SCHOLAR
PROJECT REPORT

NO. 478

Design of a hand exoskeleton system actuated via linear and adaptive control for rehabilitation

by

Midshipman 1/C Anderson S. Camp, USN

UNITED STATES NAVAL ACADEMY
ANNAPOLIS, MARYLAND

This document has been approved for public release and sale; its distribution is unlimited.
A hand exoskeleton is designed and constructed to achieve five hand positions: (1) fully extended, (2) hook fist, (3) right angle to the palm, (4) straight fist, and (5) fully flexed. These hand orientations comprise the five positions defining a rehabilitation exercise known as tendon glide. The device is significant in its ability to move the two joints distal to the palm independently of the joint adjoining the palm, without requiring bulky, rigid hardware located on the finger. Movement of the finger is achieved through hydraulically activated fluidic artificial muscles (FAMs). FAMs show improved force-to-weight ratios, cost, and alignment strategies over traditional, rigid hydraulic cylinders and allow forces to be applied across a flexed joint of the finger as it straightens. A direct model of the relationship between the volume transferred to the FAM by the hydraulic cylinder and the strain of the FAM is developed and validated through experiment. The strain-volume relationship remains constant regardless of load, enabling streamlined models and control algorithms. Position-based control of the FAMs is achieved, in both simulation and experiment, with a Proportional Integral (PI) controller and a Model Reference Adaptive Controller (MRAC). The PI controller is a linear algorithm characterized by constant controller gains. Alternatively, MRAC is an adaptive control algorithm characterized by time-varying controller gains, which can guarantee convergence of the actual system to a defined reference system. The resultant device is a wearable exoskeleton actuated by FAMs and governed by novel control architecture. The exoskeleton is capable of guiding a finger through all five positions of tendon glide. The exoskeleton aims to assist patients with at-home rehabilitation, particularly targeting patients who are typically unable to conduct their exercises without assistance from an occupational therapist.
Design of a hand exoskeleton system actuated via linear and adaptive control for rehabilitation

by

Midshipman 1/C Anderson S. Camp
United States Naval Academy
Annapolis, Maryland

___________________________
(signature)

Certification of Adviser(s) Approval

Assistant Professor Paola Jaramillo Cienfuegos
Weapons, Robotics and Control Department

___________________________
(signature)

___________________________
(date)

LCDR Edward M. Chapman, USN
Physics Department

___________________________
(signature)

___________________________
(date)

Acceptance for the Trident Scholar Committee

Professor Maria J. Schroeder
Associate Director of Midshipman Research

___________________________
(signature)

___________________________
(date)

USNA-1531-2
Abstract

A hand exoskeleton is designed and constructed to achieve five hand positions: (1) fully extended, (2) hook fist, (3) right angle to the palm, (4) straight fist, and (5) fully flexed. These hand orientations comprise the five positions defining a rehabilitation exercise known as tendon glide. The device is significant in its ability to move the two joints distal to the palm independently of the joint adjoining the palm, without requiring bulky, rigid hardware located on the finger. Movement of the finger is achieved through hydraulically activated fluidic artificial muscles (FAMs). FAMs are soft, biomimetic actuators consisting of an expandable bladder encased in a braided sheath. FAMs show improved force-to-weight ratios, cost, and alignment strategies over traditional, rigid hydraulic cylinders and allow forces to be applied across a flexed joint of the finger as it straightens. A direct model of the relationship between the volume transferred to the FAM by the hydraulic cylinder and the strain of the FAM is developed and validated through experiment. The strain-volume relationship remains constant regardless of load, enabling streamlined models and control algorithms. Position-based control of the FAMs is achieved, in both simulation and experiment, with a Proportional Integral (PI) controller and a Model Reference Adaptive Controller (MRAC). The PI controller is a linear algorithm characterized by constant controller gains. Alternatively, MRAC is an adaptive control algorithm characterized by time-varying controller gains, which can guarantee convergence of the actual system to a defined reference system. The resultant device is a wearable exoskeleton actuated by FAMs and governed by novel control architecture. The exoskeleton is capable of guiding a finger through all five positions of tendon glide. The exoskeleton aims to assist patients with at-home rehabilitation, particularly targeting patients who are typically unable to conduct their exercises without assistance from an occupational therapist.

Keywords: McKibben, FAM, hydraulics, rehabilitation, exoskeleton

1 Acknowledgements

The authors would like to acknowledge Professor Bishop for his support with 3D manufacturing. Additionally, the authors like to thank the Technical Support Division of the Weapons, Robotics, and Control Engineering Department, particularly Dan Rhodes and Joe Bradshaw, for their unfailing support and guidance. Finally, the authors would like to thank Annabelle Fichtner for her editorial assistance.

Contents

1 Acknowledgements 1

2 Introduction 2

3 Fluidic Artificial Muscle Model 5
   3.1 Model Nomenclature ........................................... 6
   3.2 Materials and Methods ....................................... 7
      3.2.1 Fluidic Artificial Muscle Construction .................... 7
      3.2.2 Piston-FAM Hydraulic System ............................ 7
      3.2.3 Syringe Pump for FAM Activation .......................... 8
   3.3 Piston-FAM Model Derivation and Quasi-static Validation .... 8
      3.3.1 Test Methods ........................................... 9
      3.3.2 Ideal FAM Model ...................................... 9
      3.3.3 Tondu Cylindrico-conical Model .......................... 11
Modern rehabilitation techniques have demonstrated the advantageous role technology plays in rehabilitation, particularly for victims of stroke [Thielbar et al., 2014, Volpe et al., 1999, Borboni et al., 2017]. Between 2000 and 2008 the estimated stroke incidence rate in high-income countries was 94 per 100,000 person-year, and approximately 80% of victims survived their stroke in the short term [Feigin et al., 2009]. At least 35% of stroke survivors undergoing rehabilitation who
suffered from some form of mild paresis, paresis, or paralysis recovered motor function to some degree at periods longer than two weeks post-stroke [Hendricks et al., 2002]. Therefore, there is a large population of stroke victims who have the potential to benefit from assistive-rehabilitative technology to aid in their recovery. This research aims to construct an exoskeleton governed by controllers and driven by soft robotic actuators.

Soft robotic actuators vary in construction and method of activation [Laschi and Cianchetti, 2014]. As a group, they are often lightweight, inexpensive, and can be utilized in multiple system alignments. Such actuators have been shown to be effective in lightweight climbing [Chapman et al., 2017], walking [Asbeck et al., 2015], and prosthetic/orthotic devices [Polygerinos et al., 2015, McConnell et al., 2017]. Traditional rigid robotics have been seen in devices for applications ranging from military [Wooden et al., 2010] to clinical [Ruszkowski et al., 2015], but often face limitations when interfacing with the human body. In rehabilitation devices, the use of rigid links in exoskeletons requires the highly sensitive alignment of joints [Polygerinos et al., 2015] and inefficient mechanical stops to prevent injury. Additionally, rigid robots tend to be expensive and heavier than soft robotic systems [Polygerinos et al., 2015]. Soft robots are able to take advantage of the morphology of the body due to their compliant nature. The inherent advantage allows for simplified construction of a robotic system [Polygerinos et al., 2015]. In exoskeleton applications, soft actuators can often apply force across the biological joint without the need for alignment [Asbeck et al., 2015, Chapman and Bryant, 2018].

**Figure 1: Contraction of a fluidic artificial muscle.** Pressure from the fluid causes the internal bladder to expand, leading the muscle to contract.

Soft, tensile actuators are often akin to variable stiffness springs and can replicate the mechanical function of biological skeletal muscle, as skeletal muscle employs similar actuation techniques to cause motion [Pfeifer et al., 2012]. One such type of variable stiffness artificial muscle is the fluidic artificial muscle (FAM). There are a wide range of artificial muscles categorized as FAMs. The FAM shown in Figure 1 is considered here. This early-stage artificial muscle was developed for use as an arm-and-hand orthotic for polio patients and is often named for the designer of its control valve, Joseph McKibben, whose daughter suffered from polio. The McKibben FAM design dates to the 1950’s [Tondu, 2012] and is distinctive in its high force to weight ratio. The muscle is constructed of an inner bladder, made of an elastic material such as latex, surrounded by a braided sheath, often constructed of a nylon or Kevlar® mesh. When an internal source pressure is applied to the FAM, the inner bladder is forced to expand, transmitting the force to the braided sheath. The sheath prevents the bladder from lengthening and acts to translate radial expansion to axial contraction. Recent improvements in computer modeling capabilities have led to a resurgence in the study of these actuators [Meller et al., 2014, Tondu and Lopez, 2000, Tondu, 2012]. Although FAMs are traditionally pneumatically activated, the use of hydraulic artificial muscles may greatly improve the overall system efficiency, in comparison to pneumatic FAMs [Tiwari et al., 2012]. Hydraulic FAMs have therefore been the subject of a variety of recent studies [Meller et al., 2014, Kothera
Hydraulic FAMs, also known as hydraulic artificial muscles (HAMs), pose a number of potential advantages compared to traditional, pneumatic FAMs. HAMs have been observed to increase pressure-force bandwidths and exhibit less sensitivity to load [Focchi et al., 2010]. Additionally, hydraulic systems can achieve greater degrees of efficiency and exhibit a quicker response time than pneumatic systems [Focchi et al., 2010, Meller et al., 2014]. For both hydraulic and pneumatic systems, mathematical models have been developed to describe the contraction force of a FAM [Tondu and Lopez, 2000, Meller et al., 2014, Chou and Hannaford, 1996, Kothera et al., 2009]. FAMs are often assumed to be thin-walled cylinders throughout their contraction [Tondu, 2012, Chou and Hannaford, 1996]. To account for inherent nonlinearities of the FAM as it deforms, including end-effects and elasticity, geometric and empirical correction factors have been introduced [Tondu and Lopez, 2000, Tondu, 2012, Meller et al., 2014].

**Figure 2: Tendon glide pattern of exercise.** From the starting position (straight hand), there are four different hand positions- an L-shape, a hook fist, a straight fist, and a full fist- which are completed cyclically to improve or preserve a patient’s range of motion. Adapted from ”Open Carpal Tunnel Release Post-Op Guidelines” by UVA Hand Center.

One of the frontiers of FAM use in the human-robot interaction is rehabilitation. A number of prosthetic and orthotic devices have been developed which employ robotics [Gupta et al., 2008, Polygerinos et al., 2015, Agarwal et al., 2015, Kim and Deshpande, 2017]. The beneficial role robotic systems can play in rehabilitation following stroke, particularly for the hand and arm, has been demonstrated through clinical comparisons [Kutner et al., 2010, Volpe et al., 1999, Takahashi et al., 2008]. An important physical therapy exercise for the recovery and maintenance of hand dexterity is known as tendon glide and can be performed actively or passively [Wehbe, 1987]. Tendon glide, shown in Figure 2, is comprised of five distinct hand positions dictated by the orientation of the joints of the fingers. In position one, considered here as the starting point of the exercise, the fingers are fully extended and all joints are aligned. In order to achieve position two, the proximal interphalangeal (PIP) joint and the distal interphalangeal (DIP) joint are both flexed as much as possible to form what is called a hook fist. In order to achieve position three, the PIP and DIP joints extend and the metacarpophalangeal (MCP) joint flexes to form an L-shape. In position four, the PIP joint flexes from position three to form a straight fist. To reach position five, known as a full fist, the DIP joint flexes. The patient then starts over with position one to continue the exercise.

Robotic exoskeleton devices have the potential to improve a patient’s physical therapy, particularly tendon glide, through both active and passive rehabilitation methods, while also preserving the quality of the exercise over time [Takahashi et al., 2008]. Active motion refers to the pa-
patient’s participation in the movement. Passive motion refers to an outside force guiding the patient through a motion without their participation. Both classes of exercise have been clinically associated with enhanced patient recovery through the improvement of joint function and range of motion [Wolf et al., 2011, Faso and Stills, 1975, Dent, 1993, Volpe et al., 1999]. Passive tendon glide is well-suited to be implemented with a soft exoskeleton, as it consists of achieving five well-defined positions cyclically. Soft robot actuators have the potential to be used in a rehabilitation exoskeleton to achieve the positions of tendon glide. A number of soft robotic exoskeleton gloves have been proposed [Agarwal et al., 2015, Takahashi et al., 2008, Polygerinos et al., 2015, McConnell et al., 2017], but none provide a suitable actuation methodology or model for this task. The use of a soft exoskeleton actuated by hydraulically driven FAMs is therefore proposed. The advent of FAMs actuated hydraulically (rather than pneumatically) presents an opportunity to remove the bulky pressure reservoir systems that have previously driven FAM actuation in labs. By taking advantage of the relative incompressibility of hydraulic fluid, FAMs can be driven with a simple hydraulic cylinder such as a syringe. Ryu et al. [2008] introduced a haptic feedback device driven by FAMs and actuated with a small hydraulic cylinder, but their control and actuation model was empirical rather than predictive. This work establishes a method of FAM actuation capable of cycling a FAM through the positions required for a soft robotic exoskeleton to achieve the positions of tendon glide.

This work also aims to establish a control regimen capable of optimizing the motion of a FAM-actuated soft exoskeleton through the positions of tendon glide. Control of FAMs has been achieved in a variety of fashions. Several labs have employed sliding mode controllers to help overcome unmodelled aspects of the FAM’s contraction [Cai, 2000, Jouppila et al., 2014, Shen, 2010, Braikia et al., 2011]. The majority of researchers employ some form of Proportional Integral Derivative (PID) control, often using neural networks or fuzzy controllers to adapt the gains of the controller to achieve the desired result [Chan et al., 2004, Anh and Ahn, 2011, Thanh and Ahn, 2006, Zhu et al., 2008, Tondu, 2014, Hesselroth et al., 1994, De Volder et al., 2011]. Nouri et al. [2002] were able to implement an adaptive controller for a FAM. Notably, all published examples of control algorithms for FAMs employ pneumatic FAMs rather than hydraulic FAMs. This work builds upon previous research in FAM control by developing and testing a linear and adaptive controller in simulation and experimentation - more specifically, a Proportional Integral (PI) controller and a Model Reference Adaptive Controller. Both controllers will actuate hydraulic FAMs. Through performance testing for three reference inputs, a step, ramp, and sinusoid, a controller will be selected based on minimal error tracking. The best performing controller will then be used with the soft exoskeleton.

This paper outlines the development of a rehabilitation exoskeleton for the hand. A novel model to describe the FAMs is introduced. The PI and MRAC controllers, which optimize the motion of the exoskeleton, are applied and evaluated. Finally, a functional exoskeleton, significant in its ability to achieve multiple hand orientations, is presented. The project is therefore threefold - the work flow and the relationship between the three components is illustrated in Figure 3.

3 Fluidic Artificial Muscle Model

A model is developed to accurately describe the FAMs used in this work. The nomenclature used to describe the models is defined. The system test bed, the construction of the FAMs, and the FAM inputs from the syringe pump are described. Next, the process by which the coupled pump-FAM model is generated, beginning with the idealized FAM model, is explained. A previously suggested cylindrico-conical FAM model [Tondu, 2012] is adapted for the coupled model and a
Figure 3: Scope of investigation. The investigation has three phases - modeling the FAMs, designing and testing a controller for the FAMs in simulation and experimentation, and designing the soft exoskeleton to achieve the positions of tendon glide.

A novel fixed-end cylindrical model is introduced. The fixed-end model is shown to have an improved volume-strain relationship over other models for the studied system. The mathematical model is validated through the dynamic responses of the experimental setup via open loop.

3.1 Model Nomenclature

- $a$: Geometric constant
- $\alpha_0$ and $\alpha$: Initial and current braid angle of FAM
- $b$: Geometric constant
- $d_e$: Experimental displacement of cylinder
- $d_s$: Displacement of cylinder
- $e_m$: Model error
- $\varepsilon$: Strain of FAM
- $F_m$: Force applied by FAM
- $k_c$: Constant to define cylindrical portion of FAM
- $l_c$: Length of conical end of FAM
- $l_e$: Length of cylindrical portion of FAM
- $l_{m,0}$ and $l_m$: Initial and current length of FAM
- $l_n$: Length of nonpantographic region
- $l_{p,0}$ and $l_p$: Initial and current length of center of FAM
- $n$: Length of displacement arrays
- $P_{app}$: Pressure of system
- $r_c$: Radius of cylindrical portion of FAM
- $r_{m,0}$ and $r_m$: Radius of FAM and initial radius of FAM
- $r_s$: Radius of hydraulic cylinder
- $V_c$: Volume of the hydraulic cylinder
- $V_m$: Volume of FAM
- $V_p$: Volume required to initially fill FAM
3.2 Materials and Methods

3.2.1 Fluidic Artificial Muscle Construction

The FAMs modeled in this paper, as shown in Figure 4, are constructed from silicone tubing: the bladder, over-expanded Techflex braided cable sleeving: the outer sheath, and barbed end fittings. The FAMs are made in-house. The dimensions of the FAMs used in this paper are described in Table 1. These dimensions are selected specifically for the scale of the hand exoskeleton application. Two 54-mm FAMs can fit on the dorsal side of the hand to actuate a finger.

Figure 4: Fluidic artificial muscle construction. Braided cable sleeving encloses a silicon bladder. Barbed end fittings are attached to either end of the bladder and sleeving.

Table 1: Dimensions of the fluidic artificial muscles. The dimensions are selected such that the muscles could be incorporated in a hand exoskeleton.

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Measurement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Braid Angle</td>
<td>$26.5^\circ \pm 1.66^\circ$</td>
</tr>
<tr>
<td>Effective Length</td>
<td>54.0 mm</td>
</tr>
<tr>
<td>Central Radius</td>
<td>4.00 mm</td>
</tr>
<tr>
<td>Bladder Inner Diameter</td>
<td>4.76 mm</td>
</tr>
<tr>
<td>Bladder Outer Diameter</td>
<td>6.35 mm</td>
</tr>
<tr>
<td>Braided Sleeving Diameter</td>
<td>6.35 mm</td>
</tr>
</tbody>
</table>

3.2.2 Piston-FAM Hydraulic System

A test bed is designed to activate a FAM using a hydraulic cylinder - in this case a syringe pump filled with water. A schematic with the hydraulic system is depicted in Figure 5. Sensors are installed to measure the strain of the FAM, displacement of the syringe pump, and pressure of the system.

Figure 5: Schematic of the coupled hydraulic system. The position of the syringe pump, the strain of the FAM, and the gauge pressure of the system are recorded using electronic sensors. The syringe pump is driven by a threaded rod spun by a stepper motor.

The displacement of both the FAM and the syringe pump are recorded using 10 kΩ linear potentiometers (ETI Systems, California, LCP12A-50-10K), with a resolution of $1.27 \times 10^{-5}$ m,
under fixed-free boundary conditions. The potentiometers are wired to determine the position of the divider. Gauge pressure is recorded using a pressure sensor (NXP Semiconductors, Netherlands, MPX2050GP) in parallel with the FAM. The pressure is transduced from the voltage across a Wheatstone bridge within the sensor, which is amplified using an instrumentation amplifier (INA118P), resulting in a resolution of $9.0 \times 10^{-2}$ kPa. All data are collected using an mBed microcontroller (ARM, United Kingdom, LPC1768). Load is applied and varied through hanging masses, which ensure a constant and known load throughout the contraction of the FAM.

### 3.2.3 Syringe Pump for FAM Activation

A syringe pump is constructed, as shown in Figure 6, to control and measure the volumetric flow rate of the system. The plunger of the syringe is driven by the shuttle. The shuttle is threaded into a rod, which is spun by a Nema 17 stepper motor (Beauty Star, China, 17HS4401). As the threaded rod spins, the shuttle translates forward or back at a speed proportional to the angular velocity of the rod. The stepper motor is controlled by the mBed microcontroller. The displacement of the shuttle is measured using a 10 kΩ linear potentiometer.

![Figure 6: In-house syringe pump designed for FAM actuation.](Image)

A stepper motor spins a threaded bar, which translates the shuttle and the syringe plunger. The position of the shuttle is measured using a linear potentiometer.

### 3.3 Piston-FAM Model Derivation and Quasi-static Validation

In order to provide a predictive model for the position of the FAM in the piston-FAM actuation system, the ideal cylindrical model, a thoroughly validated and widely used model [Tondu and Lopez, 2000, Meller et al., 2014], is first considered. The ideal cylindrical model relates the pressure, load, and strain of the FAM. The load on a FAM in a rehabilitation device would vary throughout the contraction, complicating the use of the ideal cylindrical model. However, models that are based on the volume of the FAM predict strain regardless of load, which circumvents the varying load in the exoskeleton application. As a result, a volume-based model is considered and compared to experimental data. First, the cylindric-conical model, originally developed by Tondu [2012], is applied and found to be not fully predictive for the volume-strain relationship of the FAMs considered in this work. Tondu’s model is then modified as a “fixed-end” cylindrical model, which is a novel geometric consideration of the FAMs. This section first describes the testing process. Then, the derivation and application of the ideal FAM model is discussed. The volume-based method is introduced and the two models are derived. Finally, the models are compared to experimental data to assess their validity.
3.3.1 Test Methods
Quasi-static validation testing is performed under four loads, 0.0 N, 2.93 N, 5.26 N, and 7.58 N, which are similar to the range of loads expected in the tendon glide application during rehabilitation. The FAM is cycled through three full contractions under each of the loads at a volumetric flow rate of approximately $1.1 \times 10^{-3}$ mL/min. The pressure of the system, the displacement of the syringe pump, and the strain of the FAM are recorded. The starting point for the stroke of the FAM is the unloaded resting length, in this case 54 mm. The deformation of the FAM is measured using Cauchy, or engineering, strain. Strain is calculated as the ratio of the change of length of the FAM to the original length of the FAM. The volume ejected from the pump is calculated using the cross-sectional area of the syringe and the displacement of the plunger, which is measured. The displacement of the syringe is defined as the distance from the position where the first deflection of the FAM occurred. Data is sampled at a frequency of approximately 0.2 Hz.

3.3.2 Ideal FAM Model
In order to model the strain of a FAM as a function of load and pressure as was done by Tondu and Lopez [2000], the virtual work principle is first identified as,

$$F_m dl_m = -P_{app} dV_m,$$

where $F_m$ is the force applied by the FAM, $P_{app}$ is the applied pressure, $dl_m$ is the change in length of the FAM, and $dV_m$ is the change in volume of the FAM. The FAM is assumed to remain perfectly cylindrical and to expand according to the pantographic opening principle [Tondu and Lopez, 2000] such that,

$$\frac{l_m}{l_{m,0}} = \frac{\cos \alpha}{\cos \alpha_0} \quad \text{and} \quad \frac{r_m}{r_{m,0}} = \frac{\sin \alpha}{\sin \alpha_0},$$

where $l_{m,0}$ and $l_m$ are the initial length and current length, $r_{m,0}$ and $r_m$ are the initial radius and current radius, and $\alpha_0$ and $\alpha$ are the initial and current braid angle of the FAM, respectively. The pantographic opening principle is illustrated in Figure 7, which shows the opening of braids and the resultant change in braid angle. Additionally, the variables used to describe the FAM are labeled.

Combining the two equations in eqn. (2) through trigonometric substitution yields the radius of the FAM defined as a function of FAM length,

$$r_m = \frac{r_{m,0}}{\sin \alpha_0} \sqrt{1 - \left( \frac{l_m}{l_{m,0}} \right)^2 \cos^2 \alpha_0}.$$
The strain of the FAM is defined as \( \varepsilon = \frac{l_{m,0} - l_m}{l_{m,0}} \). Through substitution into the equation for the volume of a cylinder, the volume of the FAM, \( V_m \), is defined as,

\[
V_m = \pi r_m^2 l_{m,0} \left( \frac{(1 - \varepsilon)}{\sin^2 \alpha_0} - \frac{(1 - \varepsilon)^3}{\tan^2 \alpha_0} \right).
\]

(4)

Applying the ideal cylindrical definition of FAM volume, eqn. (4), to the virtual work principle, eqn. (1), yields the force applied by the FAM defined as,

\[
F_m = \pi r_m^2 P_{app} \left( a (1 - \varepsilon)^2 - b \right),
\]

(5)

where \( a \) and \( b \) are geometric constants defined as \( a = 3/\tan^2 \alpha_0 \) and \( b = 1/\sin^2 \alpha_0 \).

Figure 8: (a) Relationship between the pressure of the system, the force applied to the system, and the strain of the FAM. (b) Relationship between the volume transferred to the FAM, the force applied to the system, and the strain of the FAM. Throughout the FAM stroke, the pressure required to reach a contraction increases as the load increases. No such effect is observed for the relationship between the volume and the load.

The resultant model, due in part to the end-effects of the FAM, tends to overpredict the strain of the system [Meller et al., 2014], which has lead to the development of empirical correction factors [Tondu and Lopez, 2000, Meller et al., 2014]. In order to predict the strain of the FAM, both the load of the system and the pressure of the system must be known, as eqn. (5) demonstrates. The variance seen between these three variables is shown in Figure 8(a). However, because the geometry of the FAM remains constant regardless of the load, the volume of the FAM can be used to predict the strain of the FAM irrespective of load, as shown in 8(b).
3.3.3 Tondu Cylindrico-conical Model

The Tondu cylindrico-conical FAM model describes the volume of a FAM as a cylinder with two symmetric, conical frustums on either end, as depicted in Figure 9. The length of the conical ends is defined as,

\[ l_c = \varepsilon l_m k_c, \]  

where \( l_c \) is the length of the one conical end, and \( k_c \) is any constant greater than 0, such that \( k_c \leq \frac{1 - \varepsilon_{\text{max}}}{2 \varepsilon_{\text{max}}} \) [Tondu, 2012]. The total volume of both conical ends is therefore,

\[ V_c = \frac{2\pi l_c}{3} \left( r_m + r_{m,0} \right)^2, \]

where \( V_c \) is the volume of both frustums.

![Initial condition](image)

**Figure 9: Tondu cylindrico-conical FAM.** As the length of the FAM decreases, the radius of the cylindrical portion increases. The larger radius of the cones is determined by the current radius of the center. The height of the cones is determined as a function of the strain of the FAM.

The pantographic opening principle, as shown in eqn. (2), is applied to the cylindrical portion of the muscle with length, \( l_e \). The cylindrical portion of the muscle is defined as, \( l_e = l_m - 2l_c \). The total volume of the FAM is then defined as \( V_m = V_c + V_e \), where \( V_e \) is the volume of the cylindrical portion. The virtual work theorem can again be applied to this model, as was done by Tondu [2012], resulting in,

\[ F_m = \pi r_{m,0}^2 P_{\text{app}} \left( -1 + \frac{2(1-\varepsilon)}{\tan^2 \alpha_0} \left( 1 - \varepsilon + \frac{k_c \varepsilon (\sin \alpha_0)}{3\sqrt{1-(1-\varepsilon)^2 \cos^2 \alpha_0}} \right) \right). \]  

The cylindrico-conical model as described by Tondu [2012] is adapted to predict the strain of a FAM as a function of the displacement of a hydraulic cylinder, rather than as a function of load and applied pressure as in eqn. (8). The volume ejected from a hydraulic cylinder, \( \Delta V_d \) is defined as,

\[ \Delta V_d = \pi r_s^2 d_s, \]

where \( r_s \) is the internal radius of hydraulic cylinder and \( d_s \) is the displacement of the plunger of the cylinder.
The change in volume of the FAM is defined as,

\[
\Delta V_m = \pi r_m^2 l_e + \frac{2\pi l_c}{3} \left( r_m + r_{m,0} \right)^2 - \pi r_{m,0}^2 l_{m,0} + V_p,
\]

(10)

where \( \Delta V_m \) is the change in volume of the FAM and \( V_p \) is the initial volume of the fluid required to completely fill the FAM, as was done by Meller et al. [2014].

Assuming a closed hydraulic system, \( \Delta V_m \) can be equated to \( \Delta V_c \) such that

\[
\pi r_s^2 d_s = \pi r_{m,l}^2 l_e + \frac{2\pi l_c}{3} \left( r_m + r_{m,0} \right)^2 - \pi r_{m,0}^2 l_{m,0} + V_p.
\]

(11)

The resultant model is one way of relating the displacement of a hydraulic cylinder to the strain of a FAM. However, as shown in Figure 10, it fails to accurately predict the strain of the FAMs used in this paper. For this model, \( k_c = 0.5 \), as it most closely matches the experimental results. The data shown is for the middle load tested, 5.26 N, and is representative of the other loads tested. Data for all the loads can be seen in the Supplemental material section in Figure S1. As the model overpredicts the strain of the FAMs made for a rehabilitation exoskeleton, a “fixed-end” cylindrical model is introduced.

![Figure 10: Discrepancy between adapted Tondu cylindrico-conical model and experimental data for a representative load of 5.26 N. The cylindrico-conical model overpredicts the strain of the FAM for a given volume.](image)

### 3.3.4 Fixed-end Cylindrical Model

The adaptation of the Tondu cylindrico-conical model discussed in the previous section attempts to relate the volume of the FAM to the strain of the FAM. However, due to the relatively small
bladder diameter and low strand-count of the braided mesh used, the Tondu cylindrico-conical model does not accurately predict the contraction of FAMs in this study. To better represent the position of the FAMs with respect to volume, a modified, fixed-end cylindrical model is introduced. As was originally observed by Kothera et al. [2009], the end-effects of the FAM make it useful to consider the modeled length of the FAM as shorter than the actual length. The novel fixed-end cylindrical model combines the approaches of Kothera et al. [2009] and Tondu [2012]. In the fixed-end cylindrical model, the extremities of the FAM expand radially without contracting axially. The ends therefore consume volume without contributing to strain. As in the Tondu cylindrico-conical model, only the center portion of the FAM is considered to contract according to the pantographic opening principle. A labeled diagram of the contraction of the FAM according to the fixed-end model is shown in Figure 11. The length of the ends, $l_n$, is treated as a constant where, $l_n = 2.5r_{m,0}$.

![Initial condition](image1)

![FAM pressurized](image2)

**Figure 11: Fixed-end FAM.** The length of the ends is constant. The radius of the FAM is determined by the length of the center portion. The volume is determined by the radius and total length, including the ends.

The length of the center region, $l_p$, is therefore, $l_p = l_m - 2l_n$. As only the central length, $l_p$, affects the radius of the FAM, $r_m$ is redefined as a function of $l_p$. The relationship yields,

$$r_m = \frac{r_{m,0}}{\sin\alpha_0} \sqrt{1 - \left(\frac{l_p}{l_{p,0}}\right)^2 \cos^2\alpha_0},$$

(12)

where $l_{p,0}$ is the initial length of the pantographic region, given by $l_{p,0} = l_{m,0} - 2l_n$. The FAM is then assumed to remain cylindrical throughout its contraction. The radius of the cylinder is given by eqn. (12), which defines the radius of the center of the FAM, and the length of the cylinder is the total length of the FAM, $l_m$.

The volume of the FAM described in eqn. (4) is reformulated as,

$$V_m = \pi r_{m,0}^2 l_{m,0} \left( \frac{1 - \varepsilon}{\sin^2\alpha_0} \right) \left( \frac{1 - \varepsilon}{\tan^2\alpha_0} \left( \frac{l_{m,0}(1 - \varepsilon) - 5r_{m,0}}{l_{m,0} - 5r_{m,0}} \right)^2 \right).$$

(13)

Consequently, the change in volume of a FAM is related to the displacement of a coupled hydraulic
cylinder by,

\[ \pi r_s^2 s = V_p - \pi r_{m,0}^2 l_{m,0} + \pi r_{m,0}^2 l_{m,0} \left( \frac{1 - \varepsilon}{\sin^2 \alpha_0} - \frac{1 - \varepsilon}{\tan^2 \alpha_0} \left( \frac{l_{m,0}(1 - \varepsilon) - 5r_{m,0}}{l_{m,0} - 5r_{m,0}} \right) \right). \]  \hspace{1cm} (14)

The predicted strain and the experimental strain are plotted as a function of the transferred volume. The result is shown in Figure 12. Again, the data shown is for a representative load of 5.26 N. Data for all the loads is shown in Figure S2 in the Supplemental Material section. By assuming only the central portion of the FAM acts as a pantograph, the model accurately predicts the strain of a FAM as shown in Figure 12.

![Figure 12: Validation of fixed-end cylindrical model for a representative load of 5.26 N. There is less than 0.02 strain error between the experimental and predicted contractions from 0.0 to 0.2 strain.](image)

### 3.3.5 Model Comparison

The Tondu cylindrico-conical model and the fixed-end model both relate the volume ejected from cylinder via the displacement of a cylinder. The Tondu model in this case overpredicts the strain of the FAM for a given volume. The overprediction may be attributed to the magnified end-effects of small-sized FAMs, which limit the applicability of the pantographic opening principle to a smaller region. Moreover, the ends of the FAMs are approximated as frustums in the Tondu model, although visual inspection of the FAMs used in this paper reveals the ends are more cylindrical than conical when the muscle is pressurized. The fixed-end model, which assumes cylindrical end-effects, is able...
Figure 13: (a) Comparison of cylindrico-conical and fixed-end model. (b) Error of the two models. The models are both plotted as a function of syringe displacement. The fixed-end model more accurately predicts the strain of the FAM.

to more accurately predict the strain of a FAM given the displacement of a syringe pump, as seen in Figure 13(a).

The error seen in Figure 13(b) is defined as,

$$e_{m} = | d_{s} - d_{e} |$$

(15)

where $e_{m}$ is the error between the modeled syringe displacement, $d_{s}$, and the experimental syringe displacement, $d_{e}$, for a given strain. The mean of the error is defined as,

$$\bar{e}_{m} = \frac{\sum e_{m}}{n}$$

(16)

where $n$ is the length of the displacement arrays. The average error for the Tondu cylindrico-conical model is 3.43 mm, while the average error of the fixed-end cylindrical model is just 0.50 mm.

3.4 Dynamic Experimental Validation

In order to assess the ability of the coupled piston-FAM system to actuate FAMs dynamically and to evaluate the performance of the fixed-end model under dynamic conditions, a series of open-loop, dynamic tests are conducted and compared to model predictions. In this section, the dynamic test procedure is explained. Then, the performance of the quasi-static model in dynamics conditions is discussed. The model is scaled to account for the differences between dynamic and quasi-static testing. Finally, the scaled model is validated against further dynamic data.

3.4.1 Dynamic Testing Procedure

In order to observe the differences between quasi-static and dynamic actuation, the FAM is driven to three different strains, 0.20, 0.15, and 0.10, at three different syringe pump speeds for each, 7.4, 4.5, and 2.8 cm/min. The syringe pump used in quasi-static testing is again used in dynamic testing. Additionally, the FAMs are configured in the same set up with the same sensors. The displacement of the syringe is defined as the distance from the position where the first deflection of the FAM occurred. Dynamic testing is done under one load, 2.57 N, which is chosen arbitrarily from the previously tested loads. The FAM holds the contracted position for five seconds. The muscle then expands at the same rate at which it contracted. The cycle is meant to mimic the performance of tendon glide.
3.4.2 Dynamic Results

Figure 14: (a) Input of approximately 2.8 cm/min to model and experimental system for dynamic testing. (b) Strain of model and experimental system for amplitude of 0.20 strain. The simulated and experimental input of the system match well. The model is able to replicate the shape of the experimental results without lag. However, the model underpredicts the strain of the FAM throughout the contraction.

For the largest stroke of 0.20 strain, the system and the model have very similar input, yet the model underpredicts the strain significantly throughout the stroke, as shown in Figure 14. The experimental input does, however, degrade slightly at the far end of the stroke, rather than hold its plateau. The degradation is due to the syringe pump slipping, as it does not actively hold its position when stopped. The dynamic conditions clearly have an effect on the relationship between the strain and the volume of the FAM. The difference between the dynamic and quasi-static response could arise from friction. Given that the movement of the braided sheath surrounding the FAM largely determines the shape of the FAM, the interweave friction that Tondu [2012] discusses could affect the shape of the FAM. In quasi-static testing, the friction is static and likely greater than the dynamic friction that would be expected during dynamic testing. Less friction may allow the braids to open more freely, causing the muscle to shorten more with the same volume when compared to higher friction scenarios. With the possible effects of friction in mind, the model is scaled linearly by a factor of 1.35, which is determined empirically, leading to much better prediction as shown in Figure 15. The same scaled model is expanded to varied rates for the same amplitude, as well as two other amplitudes.

Figure 15: (a) Input of approximately 2.8 cm/min to model and experimental system for dynamic testing. (b) Strain of scaled model and experimental system for amplitude of 0.20 strain. After adjusting for the effects of dynamic versus static friction, the model accurately predicts the experimental strain.
Figure 16: (a) Input to model and experimental system for dynamic testing: (i) 4.5 cm/min (ii) 2.6 cm/min. (b) Strain of scaled model and experimental system for amplitude of 0.20 strain. After adjusting for the effects of dynamic versus static friction, the model accurately predicts the experimental strain.

The scaled model reasonably predicts the contraction of the FAM for all amplitudes and loads, as shown in Figures 15 - 18. The scaled model least accurately predicts the strain of the FAM for

Figure 17: (a) Input to model and experimental system for dynamic testing: (i) 7.4 cm/min (ii) 4.5 cm/min (iii) 2.6 cm/min. (b) Strain of scaled model and experimental system for amplitude of 0.15 strain. After adjusting for the effects of dynamic versus static friction, the model accurately predicts the experimental strain. The discrepancy at the peaks is due in part to a slight mismatch inherent to the model.
the amplitude of 0.15 strain. The error for the 0.15 amplitude stroke can be attributed in part to the error seen in the fixed-end model under quasi-static conditions, which is evident in Figure 12 and 13(b). The scaled model, for all amplitudes and rates, generally underpredicts the strain of the FAM. The experimental and modeled input correspond better for the low amplitude tests shown in Figures 17(a) and 18(a). As the system is under less pressure at the lower strains, less slipping occurs while the pump idles at the peak of the contraction.

Figure 18: (a) Input to model and experimental system for dynamic testing: (i) 7.4 cm/min (ii) 4.5 cm/min (iii) 2.6 cm/min. (b) Strain of scaled model and experimental system for amplitude of 0.10 strain. After adjusting for the effects of dynamic versus static friction, the model accurately predicts the experimental strain.
3.5 Discussion

This section introduces a fully coupled piston-FAM model. The novelty of the model is the ability to determine the strain of the FAM, knowing only the volume transferred to the FAM via the displacement of a hydraulic cylinder. The use of a coupled piston-FAM model eliminated the need to consider the additional pressure required to actuate a FAM under greater load, as was demonstrated by the consistent relationship between FAM strain and FAM volume shown in Figure 8(b). The pressure required to command a strain appeared to increase linearly with load, as validated in Figure 8(a). The syringe pump accounted for this increased pressure by applying more force to the plunger during contraction. By innately applying more force when required, the model was able to predict the strain of the FAM without knowledge of the load.

Since traditional models were not to be suitable for the FAMs used in this work, a novel fixed-end cylindrical model was introduced. The model built upon previous models by providing a functional way to describe the volume of small FAMs, and relate the FAM’s volume and strain. The model can be tuned to fit other small FAMs by measuring the length of the fixed-ends where no appreciable contraction is occurring. In this way, the model is predictive, yet potentially compatible with a number of FAM designs.

The novel, fixed-end model was found to replicate the shape of a dynamic FAM response well, but consistently underpredict the strain. This was hypothesized to be the result of frictional differences. When the model was scaled, it was able to predict the dynamic response of a FAM with reasonable accuracy. The differences in the dynamic and quasi-static responses warrants further investigation.

By employing a coupled piston-FAM system, the traditional, bulky pressure reservoirs required to actuate traditional FAMs were eliminated. The system only required a method for actuating the plunger of the hydraulic cylinder. In other systems, this could be achieved through a number of linear actuators or rotational actuators with gearing. The size of the hydraulic cylinder could be as small as the maximum volume of the FAM. The syringe pump design, therefore, offers improved portability and utility, in comparison with the traditional, bulky tanks required to supply pressure.

Additionally, the model employs a direct relationship between the position of the FAM and the position of the hydraulic cylinder. Therefore, control can be achieved with less information about the system state. Other FAM models require both the load and pressure of the system to be known in order to predict the position of a FAM [Tondu and Lopez, 2000, Meller et al., 2014]. An exoskeleton employing FAMs actuated with a hydraulic cylinder could be used to articulate a patient’s hand through the positions of tendon glide for rehabilitation with minimal hardware and sensory information to increase portability for the patient. The system would not require pressure sensors in order to control the position of the FAM, nor would the load of the system need to be known. The model proposed therefore increases the utility of FAMs in robotic applications, in particular, the area of rehabilitation robotics.

4 Linear and Adaptive Control Algorithms

To effectively control the FAMs used in this paper, a control system which governs the amount of fluid injected to the FAM is introduced. A simulation for the dynamics of the whole system, including the pump and the controller, is designed. Both a Proportional Integral (PI) controller and a Model Reference Adaptive Controller (MRAC) are implemented in simulation and experiment. The fidelity of these controllers when responding to step, ramp, and sinusoid inputs is evaluated. Ultimately, the controller best able to track the reference inputs is considered for the rehabilitation device application.
4.1 Control Nomenclature

- $a$: System state constant
- $a_r$: Reference system state constant
- $b$: System state constant
- $b_r$: Reference system state constant
- $d_s$: Displacement of cylinder
- $e_{ad}(t)$: MRAC error
- $e_{PI}(t)$: PI error
- $\gamma_r$: Adaptive tuning parameter
- $\gamma_x$: Adaptive tuning parameter
- $k_P$: Proportional gain
- $k_I$: Integral gain
- $r$: MRAC desired strain
- $t$: Time
- $\theta_t(t)$: Adaptive gain
- $\theta_x(t)$: Adaptive gain
- $u(t)$: Generic control effort
- $u_{ad}(t)$: MRAC control effort
- $u_{exp}(t)$: Experimental control effort
- $u_{id}(t)$: MRAC ideal control effort
- $u_{PI}(t)$: PI control effort
- $u_{sim}(t)$: Simulated control effort
- $\sigma$: Standard deviation
- $x(t)$: FAM strain
- $x_d(t)$: Desired FAM strain
- $x_{exp}(t)$: Experimental FAM strain
- $x_m(t)$: Measured FAM strain
- $x_r(t)$: Reference FAM strain
- $x_{sim}(t)$: Simulated FAM strain

4.2 Control Simulation

A Simulink block diagram is developed to simulate the system. The major components of the simulation, as labelled in Figure 19, are the FAM, the pump, and the controller, whether a PI or and MRAC. The pump is modelled as a motor where rotation results directly in fluid flow. The system is assumed to be free of the effects of static and dynamic friction. The simulation accounts for damping through a damping coefficient, which is tuned based on experimental data. The load on the pump is approximated through a linear relationship between the strain of the FAM and the torque on the motor. The FAM contracts as the pressure increases and can therefore be used to estimate the torque-load of the pump.

4.3 Control Architecture

The control algorithms described in this section are implemented through an mBed microcontroller (ARM, United Kingdom, LPC1768). Positional feedback is achieved via the linear potentiometers described in section 3.2. The mBed microcontroller updates the control effort at 100 Hz. The syringe pump described in section 3.2.3 is redesigned with a DC motor (Pololu, Nevada, 2825) rather than a stepper motor. The use of a DC motor allows for control of the pump to be achieved
by simply varying the voltage across the motor, rather than by varying the step size. The control effort is a pulse width modulation (PWM) signal controlling the voltage applied across the DC motor. The DC motor translates the shuttle, which results in fluid flow. The control architecture is illustrated in Figure 20.

4.4 Control Test Methods

Both the PI controller and the MRAC are tested using the linear test bed described in section 3.2. Data is sampled at 10 Hz using an mBed microcontroller (ARM, United Kingdom, LPC1768). A step, ramp, and sinusoid is supplied as the desired input signal, \( x_d(t) \), for the system. 30 trials are performed for each signal and controller. Each of the signals is shown in Figure 21. The step is an instantaneous step at \( t = 0.01 \) from 0.0 to 0.2 strain. The ramp signal has a slope of 0.02 and a final value of 0.2 strain. The equation of the sinusoidal wave is \( x_d(t) = .1(1 - \cos(.1\pi t)) \). The period of the signal is 20 seconds in order to approximate the frequency of a typical rehabilitation exercise. The mean of each of the 30 trials is analyzed for response characteristics. The standard deviation for each signal is displayed as a shaded region on each graph. To counteract the effects of friction in experimentation, a constant voltage of 6.36 V is added to the signal produced by the
controller. The constant voltage, determined experimentally, is equivalent to the minimum voltage that enables the pump to actuate in the unloaded configuration.

Figure 21: Desired response signals. (a) Step (b) Ramp (c) Sinusoidal. The signals are sent to the system under both PI controller and MRAC in order to gauge tracking error.

4.5 Proportional Integral Controller

4.5.1 Derivation

For the purpose of FAM actuation, a PI controller is used as described in Ogata [2009] and Dorf and Bishop [2010]. The block diagram of this controller is shown in Figure 22. In this model, the user specifies the desired strain, $x_d(t)$, as the system input, which is then compared to the current measured state, $x(t)$. The system error, $e_{PI}(t)$, is defined as follows,

$$e_{PI}(t) = x_d(t) - x(t).$$  \hspace{1cm} (17)

Figure 22: Block diagram of the closed-loop system using a Proportional Integral controller. For PI control, the error $e_{PI}(t)$ is determined from the state $x(t)$ and desired state $x_d(t)$. A control effort, $u_{PI}(t)$, based on $e_{PI}(t)$ is then input into the system.

Based on this definition of the error, the PI controller then determines the control input, $u_{PI}(t)$, through the following law,

$$u_{PI}(t) = k_P e_{PI}(t) + k_I \int_0^t e_{PI}(\tau) d\tau,$$  \hspace{1cm} (18)
where $k_P$ is the proportional gain and $k_I$ is the integral gain to minimize steady state error.

To tune the controller, a procedure similar to the Ziegler-Nichols Critical Gain method presented in Ogata [2009] is applied. For a desired strain, $x_d(t) = 0.2$, and with the integral gain, $k_I$, set to 0, the proportional gain, $k_P$, is steadily increased until the control input, $u_{PI}(t)$, peaks at the upper saturation limit. The resultant value is considered to be the optimal proportional gain. Using this value, the $k_I$ term is gradually increased until the system has minimal overshoot to ensure a proper transient response.

### 4.5.2 Results

The gains used for the PI controller in both experimentation and simulation for the step, ramp, and sinusoidal input are $k_P = 2.35$ and $k_I = 0.28$. The system successfully responded to the step input as shown in Figure 23(a). The mean shown is across 30 trials, and average standard deviation of the group is $\bar{\sigma} = 0.0028$, indicating reproducibility. The simulation lagged the experimental mean in the transient portion of the response. Similarly, the simulated control effort in Figure 23(b) lags the experimental effort. Both the simulation and the experimental system achieve the desired strain of $x_d(t) = 0.2$.

![Figure 23: System response under PI control to a step input of magnitude 0.2. (a) Strain of FAM $x(t)$ (b) Control Effort $u_{ad}(t)$. The group has an average standard deviation of $\bar{\sigma} = 0.0028$. Simulation lags the experimental data in the transient portion of the response. Both the simulation and the experimental system achieve the desired strain of $x_d(t) = 0.2$.](image)

As shown in Figure 24(a), the experimental system lagged the ramp input by approximately 0.8 seconds throughout the contraction. The system ultimately settled lower than the desired final value of $x_d(t) = 0.2$. The simulation also lagged the reference system, more so than in experimentation, but settled above 0.2. The shape of the response mirrors that of the desired response. Figure 24(b) shows remarkable similarity between the simulated and experimental control effort, $u_{PI}(t)$. 

![Figure 24:](image)
Figure 24: System response under PI control to a ramp input of slope 0.02 with a final value of 0.2. (a) Strain of FAM $x(t)$ (b) Control Effort $u_{ad}(t)$. The group has an average standard deviation of $\bar{\sigma} = 0.0013$, indicating reproducibility. Both the simulation and the experimental strain lag the reference system by approximately 0.8 seconds. The simulated and experimental control efforts are similar.

Figure 25: System response under PI control to a sinusoidal input. (a) Strain of FAM $x(t)$ (b) Control Effort $u_{ad}(t)$. The group has an average standard deviation of $\bar{\sigma} = 0.0008$. Both the simulation and the experimental mean lag the desired response, $x_d(t)$. The simulated and experimental control efforts are similar in shape, with the simulated effort out of phase with the experimental effort.
For the sinusoidal input, the system again lagged the desired response by approximately 0.8 seconds, as shown in Figure 25(a). The shape of the experimental response mirrors that of the desired response, with the exception of some sticking at the maximum contraction of the FAM, possibly due to the increased load on the motor and friction. In Figure 25(b), the experimental control effort corresponds to the simulated control effort, although it is out of phase with the simulation.

4.6 Model Reference Adaptive Controller

4.6.1 Derivation

In addition to the PI controller, a MRAC algorithm based on the theory presented in Ioannou and Sun [1996] and Narendra and Annasway [1989] is used. The block diagram, seen in Figure 26, shows the control effort as a function of the adaptive gains, \( \theta_r(t) \) and \( \theta_x(t) \), reference input, \( x_r(t) \), and current state, \( x(t) \). The adaptive gains, likewise, depend on the reference input, \( x_r(t) \), current state, \( x(t) \), and error between the unknown plant and reference system, \( e_{ad}(t) \). Within Figure 26, \( x_d(t) \) is the desired strain, \( x_r(t) \) is the strain of the reference system, and \( x(t) \) is the strain of the FAM. The following derivation was done by Jaramillo Cienfuegos et al. [2017] and is repeated here for the FAM system.

![Figure 26: Block diagram of the closed loop system using a Model Reference Adaptive Controller.](image)

The error, \( e_{ad}(t) \), state, \( x(t) \), and desired strain, \( x_d(t) \), are used to define the dynamics for adaptive gains, \( \theta_r(t) \) and \( \theta_x(t) \). The control effort is computed using these gains, which then drives the experimental system to the reference system.

The following linear structure is implemented to describe the FAM behavior,

\[
\dot{x}(t) = ax(t) + bu_{ad}(t),
\]

where \( x(t) \in \mathbb{R} \) is the system state, \( a \in \mathbb{R} \) is an unknown system state constant, \( b > 0 \) is the system input constant with known sign and unknown magnitude, and \( u_{ad}(t) \in \mathbb{R} \) is the control effort. The MRAC algorithm ultimately forces the system given in eqn. (19) to converge to the following reference model,

\[
\dot{x}_r(t) = a_r x_r(t) + b_r r(t),
\]

where \( x_r(t) \in \mathbb{R} \) is the reference model state, \( a_r < 0 \) is a negative (stable) system state constant, \( b_r \in \mathbb{R} \) is a known input constant, and \( r(t) \in \mathbb{R} \) is the reference input. The reference input, \( r(t) \), is then chosen so that the reference system \( x_r(t) \) tracks the desired trajectory, \( x_d(t) \in \mathbb{R} \).

It is assumed that ideal gains, \( \theta^*_x \in \mathbb{R} \) and \( \theta^*_r \in \mathbb{R} \), exist to drive the system to the reference model through an ideal control law, \( u_{id}(t) \doteq \theta^*_x x(t) + \theta^*_r r(t) \). By substituting \( u_{id}(t) \) into eqn. (19),
the following relationship is obtained,

\[
\dot{x}(t) = (a + b\theta^*_x)x(t) + (b\theta^*_r)r(t).
\]  

(21)

The error is given by \( e_{ad}(t) \triangleq x(t) - x_r(t) \). Choosing \( \theta^*_x \triangleq \frac{a_r - a}{b} \) and \( \theta^*_r \triangleq \frac{b_r}{b} \), the closed-loop system expression simplifies to the reference model and the error dynamics are given by

\[
\dot{e}_{ad}(t) = \dot{x}(t) - \dot{x}_r(t) = a_r x(t) - b_r r(t) - (a_r x_r(t) + b_r r(t)) = a_r e_{ad}(t).
\]  

(22)

Given that the reference model is chosen to be stable \((a_r < 0)\), the error dynamics are also stable, and the system state converges to the reference state. However, since \( a \) and \( b \) are unknown, the ideal gains \( \theta^*_x \) and \( \theta^*_r \) cannot be computed and the ideal control law \( u_{id}(t) \) cannot be implemented. Hence, the ideal control law needs to be modified into an adaptive one which, instead of the ideal gains, implements adaptive gains, \( \theta^*_x(t) \) and \( \theta^*_r(t) \).

**Theorem 1** Consider the system eqn. (19), the reference system eqn. (20) and adaptation laws given by

\[
\dot{\theta}_x(t) \triangleq -\gamma_x e_{ad}(t)x(t)\text{sign}(b),
\]

(23)

\[
\dot{\theta}_r(t) \triangleq -\gamma_r e_{ad}(t)r(t)\text{sign}(b),
\]

(24)

where \( \gamma_x > 0 \) and \( \gamma_r > 0 \) are the tuning parameters. Then, the closed-loop system given by (19), (20), (23), (24) with the adaptive control law

\[
u_{ad}(t) = \theta_x(t)x(t) + \theta_r(t)r(t),
\]

(25)

is Lyapunov stable, and the tracking error \( e_{ad}(t) \) converges to zero. \( \Box \)

Proof of Theorem 1 is explained in detail in the work of Jaramillo Cienfuegos et al. [2017].

The controller is tuned through the tuning constants \( \gamma_x \) and \( \gamma_r \), and the reference system parameters \( a_r \) and \( b_r \).

4.6.2 Results

The parameters used to define the MRAC are shown in Table 2. As shown in Figure 27(a), the system initially overshoots the reference trajectory significantly. The controller quickly corrects and is generally able to track the reference system, with slight oscillation. The average standard deviation of the group is \( \bar{\sigma} = 0.0031 \), indicating reproducibility. The control effort of the FAM,

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Experimental</th>
<th>Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \gamma_r )</td>
<td>1000</td>
<td>1000</td>
</tr>
<tr>
<td>( \gamma_x )</td>
<td>900.0</td>
<td>900.0</td>
</tr>
<tr>
<td>( \theta_x(0) )</td>
<td>-7.000</td>
<td>2.000</td>
</tr>
<tr>
<td>( \theta_r(0) )</td>
<td>7.000</td>
<td>-2.000</td>
</tr>
<tr>
<td>( a_r )</td>
<td>-0.5000</td>
<td>-0.5000</td>
</tr>
<tr>
<td>( b_r )</td>
<td>0.5000</td>
<td>0.5000</td>
</tr>
</tbody>
</table>
Figure 27: System response under MRAC to a step input of magnitude 0.2. (a) Strain of FAM $x(t)$ (b) Control Effort $u_{ad}(t)$. The experimental and simulated strains and control efforts generally agree, with the exception of the region surrounding the strain overshoot.

Figure 28: System response under MRAC to a ramp input of slope 0.02 with a final value of 0.2. (a) Strain of FAM $x(t)$ (b) Control Effort $u_{ad}(t)$. Both the simulation and the experimental strain achieve the desired response. The simulated and experimental control efforts are similar up to $t = 13s$.

$u_{ad}(t)$, is generally similar, with the exception of the large dip at $t = 1s$, as shown in Figure 27(b). The standard deviation of the control effort increases around $t = 7s$ in Figure 27(b), due to an oscillation between positive and negative control effort to maintain steady state conditions.

For the ramp case, shown in Figure 28(a), the system matched the desired response throughout
the contraction. The greatest discrepancy occurred at the beginning of the motion, exhibiting an overshoot similar to that shown in Figure 27(a), although less pronounced. The system begins to oscillate slightly as it approaches steady state at $t = 15s$. Figure 24(b) shows remarkable similarity between the simulated and experimental control effort from $t = [0 \ 13]s$. A similar oscillatory behavior to that shown in 27(b) increases the standard deviation to about $\sigma = 0.5$ after $t = 13s$.

The system achieves the desired response, a sinusoidal wave, as shown in Figure 29(a). The system again exhibits overshoot at the onset of the input of the signal command, with a more pronounced overshoot in the second period. After the overshoot, the system tracks the desired response very closely. The experimental and simulated control effort in Figure 29(b) are similar, with the experimental effort exhibiting more oscillation.

![Figure 29: System response under MRAC to a sinusoidal input. (a) Strain of FAM $x(t)$ (b) Control Effort $u_{ad}(t)$. Both the simulation and the experimental response achieve the reference system. The simulated and experimental control efforts are similar, with the exception of some oscillation in the experimental effort.](image)

### 4.7 Performance Comparison

Both the PI controller and the MRAC were successful at tracking a variety of reference signals. The MRAC, however, was more successful according to analysis of tracking error due to the persistent lag seen in the PI control. The MRAC, despite initial overshoot, was able to better track the desired signal. Side by side comparison for the step, ramp, and sinusoidal inputs are shown in Figure 30. The PI controller exhibited more tracking error, as shown in Figure 30 and Table 3. The MRAC controller, however, exhibited oscillatory behaviour. Given the intended application of a rehabilitation device, oscillations pose a safety risk, as overshoot could result in the over-bending of a joint. As the PI controller behaved without oscillation, albeit with more tracking error, the PI controller is selected for implementation in a rehabilitation device.
Figure 30: Absolute tracking error for each reference and controller (a) PI (b) MRAC (i) Step (ii) Ramp (iii) Sinusoidal. The PI controller lags the desired response, but imitates the shape of the reference. The MRAC exhibits oscillatory behavior, but does not lag.

Table 3: Error of System Responses. The average error of the MRAC is lower than the error of the PI controller for every reference signal.

<table>
<thead>
<tr>
<th>Signal</th>
<th>$\bar{e}_{PI}$</th>
<th>$\bar{e}_{ad}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ramp</td>
<td>0.0137</td>
<td>0.0045</td>
</tr>
<tr>
<td>Step</td>
<td>0.0102</td>
<td>0.0034</td>
</tr>
<tr>
<td>Sinusoidal</td>
<td>0.0197</td>
<td>0.0106</td>
</tr>
</tbody>
</table>

5 Exoskeleton Design

To prove the feasibility of a wearable device capable of achieving the positions of tendon glide, a model finger outfitted with the components of the exoskeleton is constructed as seen in Figure 31. First, the design specifications of the finger and exoskeleton are described. Then, testing methods
and results are discussed. Finally, the design of the actual wearable device is illustrated.

5.1 Materials and Specifications

In order to achieve the positions of tendon glide, a wearable exoskeleton must be able to rotate each joint from fully flexed to fully extended. Additionally, the exoskeleton must be able to move as joints independently in order to achieve as many hand positions as possible. Furthermore, the device must be sufficiently light-weight to be worn comfortably. The model finger pictured in Figure 31 consists of three links equivalent to the phalanges, and a base equivalent to the metacarpal. The dimension of the phalanges are scaled to be anatomically representative of those of an average-sized adult male. The finger is 3D printed using a Stratasys 3D printer (Stratasys, Minnesota, Eden260VS™) and VeroClear RGD810. The exoskeleton consists of five main components: the distal plug, the proximal plug, medial hydraulic connection, the restoring force sheath, and the muscles, as described in Figure 31. Both plugs and the medial hydraulic connection are 3D printed using VeroClear RGD810, while the restoring force sheath is 3D printed using polylactic acid (PLA). The restoring force itself is proved by a 4.58 N constant force spring (McMaster Carr, New Jersey, 9293K44) which servers as a restoring force to return the muscles to the fully extended position. The muscles are FAMs, constructed of braided sheathing and an expandable bladder, as previously discussed. The exoskeleton also features resistive flex sensors (SparkFun, Colorado, SEN-10264 ROHS) and 3D printed guides for those sensors to determine the orientation of the finger. Polyvinyl

Figure 31: Finger used for testing. The finger is 3D printed and outfitted with FAMs, a restoring force, and sensors.
chloride (PVC) tubing (ATP, Ohio, PVS316-516ANA) connects the medial hydraulic connection to a ball valve which allows control of each muscle individually. Pressure is supplied via the syringe pump shown in Figure 32.

![Figure 32: Pump for control testing. The DC motor drives the shuttle, which in turn depresses the syringe.](image)

### 5.2 Performance Testing

Using human-in-the-loop control, the finger was able to achieve the positions of tendon glide, as shown in Figure 33. Using just two FAMs, the finger could be orientated to achieve the five hand positions required for tendon glide. The distal FAM extended the joint equivalent to the DIP prior to the PIP joint, allowing for the achievement of position (2). The achievement of position (2) was not originally sought, but discovered during testing. The maximum strain of the FAM limited the degree of extension achieved by the finger, particularly in positions (1) and (3). A single restoring force enabled the achievement of all five positions, simplifying design.

![Figure 33: Test finger achieving positions of tendon glide with biological comparison. All five positions were achieved.](image)
As previously mentioned, a PI controller is used to govern the flexion and extension of the joint. Control is applied for the distal FAM, with feedback coming from the flex sensor across the PIP joint. Figure 34 shows the mean of ten trials with a sinusoidal reference system. For the first period, the finger began from the fully relaxed position shown as position (4) in Figure 33. As the reference system flexed, the finger flexed until the PIP joint was at 90 degrees, which corresponds to position (2) in Figure 33. As the finger started with the PIP joint on the cusp of motion, there is less delay in the second period. The controller requires notably more effort when controlling the FAMs in the finger assembly, versus the linear test bed. The increased control effort is expected, given that there is a restoring force acting against the contraction of the FAMs.

![Figure 34: Test finger under PI control (a) Sensor Position (b) Control Effort.](image)

There is delay within the system, but the joint is able to flex and extend.

5.3 Wearable Exoskeleton Design

The components used to actuate the finger model are transferable to a wearable exoskeleton. The distal and proximal plug, the medial hydraulic connection, and the restoring force sheath can be fitted to a plastic wrist splint, cut to cover only the palm, and a glove. The wearable exoskeleton would therefore have the same functionality as that achieved by the exoskeleton fitted to the model finger. Once prototypes of the wearable exoskeleton are completed, preliminary testing in healthy human subjects can begin with the approval of the Human Research Protection Program (HRPP).
6 Conclusions

To create a device actuated by FAMs and capable of guiding the hand through the positions of tendon glide, a novel model was developed and linear and adaptive controls were applied to FAMs in both simulation and experimentation. The FAMs discussed in this paper were not well described by existing models. Therefore, the fixed-end cylindrical model was developed, which accurately represents the contraction of small FAMs well suited for rehabilitation devices. The fixed-end cylindrical model assumes the center portion of the FAM contracts according to the pantographic opening principle, while the ends remain a fixed length but expand and consume fluid. The model enabled the accurate simulation of FAM dynamics, which aided in the development and tuning of linear and adaptive controllers.

MRAC and PI controllers were simulated, tuned, and tested in experimentation to govern the contraction of the FAM. Both controllers enabled the system to track three kinds of input signals: a step, ramp, and sinusoid. The MRAC demonstrated consistently less tracking error than the PI controller, but also exhibited oscillatory behavior. The PI controller, despite lag, mimicked the shape of the reference system. In order to ensure that a patient’s hand is moved smoothly and with minimal oscillations, the PI controller was chosen for implementation in a rehabilitation device.

A rehabilitation exoskeleton was capable of achieving all five positions of tendon glide for a test finger. The exoskeleton applied to the test finger is transferable to a wearable device. Implementation of a PI controller to track a sinusoidal reference system enabled the joint to flex and extend in a controllable manner. The device did not accurately track the reference system, indicating an adaptive, rather than a linear, controller may be necessary. The MRAC exhibited oscillatory behavior. Therefore, a control scheme that incorporates linear and adaptive algorithms could achieve the error tracking of the MRAC with the smooth response of the PI controller. One potential option is the Augmented Adaptive PI Controller [Jaramillo Cienfuegos et al., 2017].

7 Project Impact and Contributions

This project originally proposed to meet a series of goals in pursuit of a wearable exoskeleton capable of achieving the positions of tendon glide. Those goals and the corresponding deliverables are shown below, along with the results of the project.

Goal 1: Build, model, and control McKibben muscle for exoskeleton application

**Deliverable:** McKibben muscles capable of contracting 1 cm under minimal pressure.
**Result:** McKibben muscles capable of achieving greater than 0.2 strain.

**Deliverable:** A mathematical model representing the dynamics of the McKibben muscle based on measured input and output information across the frequency spectrum.
**Result:** A mathematical model, called the fixed-end cylindrical model, relating the volume change of a FAM to the position of the FAM, regardless of load.

**Deliverable:** Linear Proportional Integral and nonlinear Model Reference Adaptive algorithms for tracking McKibben muscle contraction with minimal error.
**Result:** Linear Proportional Integral and nonlinear Model Reference Adaptive algorithms capable of tracking McKibben muscle contraction with minimal tracking error.

**Deliverable:** Submit findings and present results of the open-loop and closed-loop simulations and experiments at a conference.

Goal 2: Adapt and model a plunger-syringe system for actuating McKibben muscles
Deliverable: A hydraulic system capable of supplying 300 KPa of pressure and 2 mL of fluid, while minimizing cross-sectional area.
Result: A hydraulic system capable of supplying well over 300 KPa of pressure and 25 mL of fluid.

Deliverable: A mathematical model representing the dynamics of the hydraulic system based on measured input and output data.
Result: A pump model based on first principles.

Goal 3: Design, build, and control an exoskeleton system for one finger
Deliverable: A wearable exoskeleton driven for one finger for open-loop hand motion.
Result: An exoskeleton fitted to a test finger for human-in-the-loop achievement of the positions of tendon glide.

Deliverable: A wearable exoskeleton to be guided by a linear and a nonlinear controller for best tracking performance. The system is capable of tracking the motions based on the tendon glide exercise routine.
Result: An exoskeleton fitted to a test finger able to flex and extend a joint when governed by linear control.

Deliverable: A manuscript for submission to a journal, such as IEEE Journal of Robotics and Automation, Journal of Mechanisms and Robotics, or the Journal of Rehabilitation and Assistive Technologies Engineering, etc.
Result: A manuscript to disseminate the full results of the project in progress. A publication accepted with revisions to the International Journal of Robotics Research, Special Issue on Soft Robotics Modeling and Control: Bringing Together Articulated Soft Robotics and Soft-Bodied Robots.

8 Dissemination of Current Research Work

Below is a bibliography of articles, posters, and presentations done in conjunction with this work.


References


9 Supplemental Material

Figure S1 shows the experimental data for all the loads tested with the Tondu cylindrico-conical strain predictions. For the FAMs tested in this paper, the model does not accurately predict the strain.

![Figure S1: Discrepancy between adapted Tondu cylindrico-conical model and experimental data. The cylindrico-conical model overpredicts the strain of the FAM for a given volume.](image1)

Figure S2 shows the experimental data for all the loads tested with the fixed-end strain predictions. The fixed-end model successfully predicts the strain for all loads.

![Figure S2: Validation of fixed-end cylindrical model. The model generally matches the experimental data.](image2)
A Appendix

A.1 Code used for MRAC and PI control in the linear test bed

```c
#include "mbed.h"
#include "SDFileSystem.h"

SDFileSystem sd(p5, p6, p7, p8, "sd"); // the pinout on the mbed Cool Components workshop board
Ticker c; //control
Ticker d; //was for data
Serial pc(USBTX, USBRX);
DigitalOut en(p18); // Motor Control
DigitalOut dr(p19); // Motor control
PwmOut st(p21); // Motor
AnalogIn distance(p15); // FAM pos
AnalogIn pressure(p16); // use for pressure sensor
AnalogIn touch(p17); // used for syringe pump pressure detection
DigitalOut ct(p11); // for oscilloscope flag
DigitalOut dt(p12); // for oscilloscope flag

float t = 0; // time
float desired; // for controller
float dist0; // for strain calc
float strain;
float er = 0; // for running error
float er0;
float PWM; // for running error
float period = .01; // control update period
float eri = 0; // for integral of error
float kp = 2.35; // proportional gain
float ki = .28; // integral gain
int signal; // Control Effort
float t0; // Pot position
int e; // for data export
FILE *fp; // for SPI export
char filename[30]; // to call SPI file
float timei; // for integral
float r = 0; // mrac reference
float rd; // reference derivative
float rd0 = 0; // for running integral
float theta_xd0 = 0; // for running integral
float theta_rd0 = 0; // for running integral
float theta_xd; // gain derivative
float theta_rd; // gain derivative
float tx = 0; // gain
float tr = 0; // gain
float ar = -.5; // reference model
float br = .5; // reference model
float gr = 1000; // tuning parameter
float gx = 900; // tuning parameter
int loopcount = 0; // for data export
int dout; // for data export
int txout; // for data export
int trout; // for data export

void PIStep() // PI control for step
{
    ct = 1; // Triggers for OScope
    ct = 0;
    ct = 1;
    strain = (distance*3.3 - dist0)/2.12598425; // strain based on 54 mm
    er0 - desired - strain; // error
    eri - eri + ((er0+er)*period)/2.0; // integral of error
    er - er0; // stores last value
    PWM = (eri * ki) + (er0 * kp); // PI control law
    if(PWM>0) { // assigns direction
dr = 1;
    } else {
dr = 0;
    }
    st = abs(PWM)+.53; // addition to beat friction
    ct = 0; // for data export
}
```
loopcount = loopcount + 1;
if( loopcount == 10) {
  if(PWM>0) {
    signal = (PWM + .53)*1000;
  } else {
    signal = (PWM - .53)*1000;
  }
  e = strain*1000.0;
  dout = desired*1000.0;
  fprintf(fp, "%d,%d,%d\n",e,signal,dout);
  loopcount = 0;
}
}

void FIRamp()
{
  strain = (distance*3.3 - dist0)/2.12598425; //strain based on 53 mm
  desired = t*.02;    //ramp
  if(desired>.2) {    //max value of ramp
    desired = .2;
  }
  er0 = desired - strain; //error
  eri = eri + ((er0+er)*period)/2.0; //integral
  er = er0;    //stores
  PWM = (eri * ki) + (er0 * kp); //control law
  if(PWM>0) { //assigns direction
    dr = 1;
  } else {
    dr = 0;
  }
  st = abs(PWM)+.53; //boost for friction
  t = t + period; //time
  loopcount = loopcount + 1; //for data export
  if( loopcount == 10) {
    if(PWM>0) {
      signal = (PWM + .53)*1000;
    } else {
      signal = (PWM - .53)*1000;
    }
    e = strain*1000.0;
    dout = desired*1000.0;
    fprintf(fp, "%d,%d,%d\n",e,signal,dout);
    loopcount = 0;
  }
}

void FISin()
{
  strain = (distance*3.3 - dist0)/2.12598425; //strain based on 54 mm
  desired = -.1*cos(.314159265*t)+.1; //sin signal
  er0 = desired - strain; //error
  eri = eri + ((er0+er)*period)/2.0; //integral
  er = er0; //stores
  PWM = (eri * ki) + (er0 * kp); //control law
  if(PWM>0) { //assigns direction
    dr = 1;
  } else {
    dr = 0;
  }
  st = abs(PWM)+.53; //boost for friction
  t = t + period; //time
  loopcount = loopcount + 1; //for data export
  if( loopcount == 10) {
    if(PWM>0) {
      signal = (PWM + .53)*1000;
    } else {
      signal = (PWM - .53)*1000;
    }
    e = strain*1000.0;
dout = desired*1000.0;
fprintf(fp, "%d,%d,%d\n",e,signal,dout);
loopcount = 0;
}

void MStep()
{
  ct = 1; //flag for oscpoe
  ct = 0;
  ct = 1;
  strain = (distance*3.3 - dist0)/2.12598425; //strain defined based on 54 mm
  rd = ar*r + br*desired; //d ofreference signal
  r = r + ((rd+rd0)*period)/2.0; //integral
  rd0 = rd; //stores value
  er = r - strain; //error
  theta_xd = gx*er*strain; //derivative of gains
  theta_rd = -gr*er*r; //integral
  tr = tr + ((theta_rd+theta_rd0)*period)/2.0; //integrate gains
  theta_xd0 = theta_xd; //stores values
  theta_rd0 = theta_rd;
PWM = tx*strain + tr*desired; //control law
  PWM = -1.0*PWM; //flips sign
  if(PWM>0) {//assign directions
    dr = 1;
  } else {
    dr = 0;
  }
  st = abs(PWM)+.53;//bosts for fritction
  ct = 0; //flag for oscpoe
  loopcount = loopcount + 1; //for data export
  if( loopcount == 10) {
    if(PWM>0) {
      signal = (PWM + .53)*1000;
    } else {
      signal = (PWM - .53)*1000;
    }
    e = strain*1000.0;
    dout = r*1000.0;
    txout = tx*1000.0;
    trout = tr*1000.0;
    fprintf(fp, "%d,%d,%d,%d,%d\n",e,signal,dout,txout,trout);
    loopcount = 0;
  }
}

void MRamp()
{

desired = t*.02;
strain = (distance*3.3 - dist0)/2.12598425; //strain defined based on 54 mm
rd = ar*r + br*desired; //d ofreference signal
r = r + ((rd+rd0)*period)/2.0; //integral
rd0 = rd; //stores value
er = r - strain; //error
theta_xd = gx*er*strain; //derivative of gains
theta_rd = -gr*er*r; //integral
tr = tr + ((theta_rd+theta_rd0)*period)/2.0; //integrate gains
theta_xd0 = theta_xd; //stores values
theta_rd0 = theta_rd;
PWM = tx*strain + tr*desired; //control law
PWM = -1.0*PWM; //flips sign
if(PWM>0) {//assign directions
  dr = 1;
} else {
  dr = 0;
}
st = abs(PWM)+.53; //bosts for fritction
207 \ t = t + period; //time
208 loopcount = loopcount + 1; //for data export
209 if (loopcount == 10) {
210 if (PWM > 0) {
211 signal = (PWM + .53) * 1000;
212 } else {
213 signal = (PWM - .53) * 1000;
214 }
215 e = strain * 1000.0;
216 dout = r * 1000.0;
217 txout = tx * 1000.0;
218 trout = tr * 1000.0;
219 fprintf(fp, "%d,%d,%d,%d,%d\n", e, signal, dout, txout, trout);
220 loopcount = 0;
221 }
222 }
223
224 void MSin()
225 {
226 desired = -.1 * cos(.314159265 * t) + .1;
227 strain = (distance * 3.3 - dist0) / 2.12598425; //strain defined based on 54 mm
228 rd = ar * r + br * desired; //d of reference signal
229 r = r + ((rd + rd0) * period) / 2.0; //integral
230 rd0 = rd; //stores value
231 er = r - strain; //error
232 theta_xd = -gx * er * strain; //derivative of gains
233 theta_rd = -gr * er * r; //
234 tr = tr + ((theta_rd + theta_rd0) * period) / 2.0; //integrate gains
235 tx = tx + ((theta_xd + theta_xd0) * period) / 2.0;
236 theta_xd0 = theta_xd; //stores values
237 theta_rd0 = theta_rd;
238 PWM = tx * strain + tr * desired; //control law
239 PWM = -1.0 * PWM; //flips sign
240 if (PWM > 0) { //assign directions
241 dr = 1;
242 } else {
243 dr = 0;
244 }
245 st = abs(PWM) + .53; //hosts for frictction
246 t = t + period; //time
247 loopcount = loopcount + 1; //for data export
248 if (loopcount == 10) {
249 if (PWM > 0) {
250 signal = (PWM + .53) * 1000;
251 } else {
252 signal = (PWM - .53) * 1000;
253 }
254 e = strain * 1000.0;
255 dout = r * 1000.0;
256 txout = tx * 1000.0;
257 trout = tr * 1000.0;
258 fprintf(fp, "%d,%d,%d,%d,%d\n", e, signal, dout, txout, trout);
259 loopcount = 0;
260 }
261 }
262 }
263
264 int main()
265 {
266 NVIC_SetPriority(TIMER3_IRQn, 1); //bumps up ticker priority
267 st.period(.00005); //PWM period
268 wait(.1); //give time
269 en = 1; // turn motor "on"
270 getchar(); //get key
271 dr = 1; //forward
272 st = 1; //100%
273 wait(.25); //for a burst
274 dr = 0; //go in reverse
275 while (touch < .22) { //until the system is unpressured
276 wait(.01);
st = .75;
}
dr = 1; //go forward
wait(.01); //stops backward motion
st = 0; //stop
desired = .2; //0.2 strain
wait(.3); //let settle
printf("Ready");
getchar(); //get key
for(int i = 1; i < 1; i++) { //do PI control for a step x number of times
    sprintf(filename, "/sd/PIStep%d.txt", i);
    fp = fopen(filename, "w"); // open for writing
dist0 = distance * 3.3; //starting point
    eri = 0; //resets integral values
er = 0;
t = 0;
loopcount = 0; //resets data export counter
c.attach(&PIStep,period); //do control
wait(20.5); //for 20.5s
c.detach(); //stop
st = 0;
dr = 0;
while(touch < .22) { //take pressure off
    wait(.01);
st = .75;
}
dr = 1;
wait(.01);
st = 0;
wait(2);
fclose(fp); //close file
}
for(int i = 1; i < 1; i++) { // do PI control for ramp
    sprintf(filename, "/sd/PIRamp%d.txt", i);
    fp = fopen(filename, "w"); // open for writing
dist0 = distance * 3.3; //start point
    eri = 0; //reset running values
er = 0;
t = 0;
loopcount = 0;
c.attach(&PIRamp,period); //do control
wait(20.5); //for 20.5 s
c.detach(); //stop
st = 0;
dr = 0;
while(touch < .22) {//take pressure off
    wait(.01);
st = .75;
}
dr = 1;
wait(.01);
st = 0;
wait(2);
fclose(fp);//close file
}
for(int i = 1; i < 1; i++) { //do PI control for sin
    sprintf(filename, "/sd/PISin%d.txt", i);
    fp = fopen(filename, "w"); // open for writing
dist0 = distance * 3.3; //starting point
    eri = 0; //reset running values
er = 0;
t = 0;
loopcount = 0;
c.attach(&PISin,period); //do control
wait(40.5); // for 40.5s
c.detach(); //stop
st = 0;
dr = 0;
while(touch < .22) { // take pressure off

wait(.01);
st = .75;
}
dr = 1;
wait(.01);
st = 0;
wait(2);
fclose(fp); //close file

for(int i = 1; i < 1; i++) { //do MRAC for step
    sprintf(filename, "/sd/MStep%d.txt", i);
    fp = fopen(filename, "w"); // open for writing
    dist0 = distance * 3.3; // starting point
    r = 0; //reset all values
    t = 0;
    rd = 0;
    rd0 = 0;
    theta_xd0 = 0;
    theta_rd0 = 0;
    theta_xd = 0;
    theta_rd = 0;
    tx = 7; // initial conditions
    tr = -7;
    loopcount = 0;
    desired = .2; // .2 strain
    c.attach(&MStep,period); // do control
    wait(20.5); //for 20.5 s
    c.detach(); // stop
    st = 0;
    dr = 0;
    while(touch < .22) {//take pressure off
        wait(.01);
st = .75;
    }
    dr = 1;
    wait(.01);
st = 0;
    wait(2);
    fclose(fp); //close file
}

for(int i = 1; i < 1; i++) { // MRAC ramp
    sprintf(filename, "/sd/MRamp%d.txt", i);
    fp = fopen(filename, "w"); // open for writing
    dist0 = distance * 3.3; // starting point
    r = 0; //reset all values
    t = 0;
    rd = 0;
    rd0 = 0;
    theta_xd0 = 0;
    theta_rd0 = 0;
    theta_xd = 0;
    theta_rd = 0;
    tx = 7; // initial conditions
    tr = -7;
    loopcount = 0;
    c.attach(&MRamp,period); // do control
    wait(20.5); //for 20.5 s
    c.detach(); // stop
    st = 0;
    dr = 0;
    while(touch < .22) {//take pressure off
        wait(.01);
st = .75;
    }
    dr = 1;
    wait(.01);
st = 0;
    wait(2);
    fclose(fp); //close file
for(int i = 1; i < 5; i++) {//do MRAC for sin
    sprintf(filename, "/sd/MSinLong%d.txt", i);
    fp = fopen(filename, "w");  // open for writing
    dist0 = distance * 3.3; //starting point
    r = 0;      // reset all values
    t = 0;
    rd = 0;
    rd0 = 0;
    theta_xd0 = 0;
    theta_rd0 = 0;
    theta_xd = 0;
    theta_rd = 0;
    tx = 7;//initial conditions
    tr = -7;
    loopcount = 0;
    c.attach(&MSin,period);//do control
    wait(80.5);//for 80.5 seconds
    c.detach();//stop
    st = 0;
    dr = 0;
    while(touch < .22) {//take pressure off
        wait(.01);
        st = .75;
    }
    dr = 1;
    wait(.01);
    st = 0;
    wait(2);
    fclose(fp);//close file
}
A.2 Code used for PI control of the test finger

```c
#include "mbed.h"
#include "SDFileSystem.h"

SDFileSystem sd(p5, p6, p7, p8, "sd"); // the pinout on the mbed Cool Components workshop board
c Ticker c; // for control
Serial pc(USBTX, USBRX);
DigitalOut en(p18); // for motor control
digitalOut dr(p19); // for motor control
PwmOut st(p21); // Motor PWM
AnalogIn pos1(p15); // finger position
AnalogIn touch(p17); // load on plunger

float t = 0; // time
float desired; // desired position
float er = 0; // error
float er0; // for integral
float PWM; // motor signal
float period = .01; // control period
float eri = 0; // for integral error
float kp = 2.35; // Porp gain
float ki = .28; // Int gain
int signal; // for data export
FILE *fp; // SPI file
char filename[30]; // SPI file name space
int loopcount = 0; // for data export
int dout; // for data export
float current; // current position
float max; // max position for finger

void PISinC1()
{
    desired = -.5*max*cos(.314159265*t)+.5*max; // sin input
    er0 = desired - pos1; // error
    eri = eri + ((er0+er)*period)/2.0; // integral
    er = er0; // stores value
    PWM = (eri * ki) + (er0 * kp); // control law
    if(PWM>0) { // assigns direction
        dr = 1;
    } else {
        dr = 0;
    }
    st = abs(PWM)+.53; // assigns speed
    t = t + period; // time
    loopcount = loopcount + 1; // for data export
    if( loopcount == 10 ) {
        if(PWM>0) {
            signal = (PWM + .53)*1000;
        } else {
            signal = (PWM - .53)*1000;
        }
        e = pos1*1000.0;
        dout = desired*1000.0;
        fprintf(fp, "%d,%d,%d\n",e,signal,dout);
        loopcount = 0;
    }
}

int main()
{
    getchar(); // get character to start
    NVIC_SetPriority(TIMER3_IRQn, 1); // assigns ticker priority
    st.period(.00005); // PWM period
    for(int i = 1; i < 13; i++) { // do PI control 13 times
        wait(.1);
        en = 1;
        st = 0;
    }
}
```

A.2 Code used for PI control of the test finger
dr = 0;
while(touch < .22) { //takes pressure off
    wait(.01);
    st = .75;
    dr = 1;
    wait(.01);
    st = 0;
    wait(.3);
    printf("Ready");
    en = 1;
max = .59; // max of sin
     sprintf(filename, "/sd/Fig%d.txt", i); //open file
fp = fopen(filename, "w");  // open for writing
eri = 0; //reset values
er = 0;
t = 0;
loopcount = 0;
c.attach(&PISinCl,period); //do control
wait(40); //for 40 s
 c.detach(); //stop
fclose(fp); //close file
st = 0; //stop motor
en = 0; }
}
A.3 PI simulation script

```matlab
clear
Ra = 4.5;    %Measured
La = 0.2835; %Inductance measured
kt = .24;   %motor torque constant
kb = kt;    %motor back emf constant
B = .0001;   %crtical damping
J = 1.96e-6; %Cylinder on axis 1.96e-6
TorqueApprox = 2300;  %conversion of strain to pressure tuned experimentally
SyringeArea = (.013^2)*pi;  %Area of the syringe plunger
Lt = .254;   %pitch of screw
e = .25;     %Efficiency based on EXP tuning.
LeadScrewFactor = Lt/(2*pi*e); %approximate equation relating force on a lead screw to
                              %torque on that lead screw

DS = .2;
Kp = 2.35;
Ki = .28;
sim('VolumeBasedControl'); %run simulation
save('PISin_Sim','PWM','Strain','tout');
figure(1) %below is plotting
subplot(2,1,1)
plot(tout,Strain,'k')
hold on
for i = 1:length(tout)
    Dsarray(i) = DS;
end
plot(tout,Dsarray,'r--')

h=fill([0,20,20,0],[DS*.98,DS*.98,DS*1.02,DS*1.02],['red', 'LineStyle', 'none']);
h.FaceAlpha=0.3;
axis([0 40 0 .25])
title('Model Response to PI Control')
xlabel('Time (s)')
set(findall(gca, '-property', 'FontSize'), 'FontSize', 14);
set(findall(gca, '-property', 'LineWidth'), 'LineWidth', 3);
legend({'Model Response', 'Desired Output'}, 'Location', 'SouthEast')
grid on
subplot(2,1,2)
plot(tout,PWM)
axis([-1 1])
grid on
title('PI Control Output')
xlabel('Time (s)')
set(findall(gca, '-property', 'FontSize'), 'FontSize', 14);
ylabel('PWM')
set(findall(gca, '-property', 'LineWidth'), 'LineWidth', 3);
legend({'Model Response', 'Desired Output'}, 'Location', 'SouthEast')
grid on
```

A.3.1 PI simulation script

```matlab
clear
Ra = 4.5;    %Measured
La = 0.2835; %Inductance measured
kt = .24;   %motor torque constant
kb = kt;    %motor back emf constant
B = .0001;   %crtical damping
J = 1.96e-6; %Cylinder on axis 1.96e-6
TorqueApprox = 2300;  %conversion of strain to pressure tuned experimentally
SyringeArea = (.013^2)*pi;  %Area of the syringe plunger
Lt = .254;   %pitch of screw
e = .25;     %Efficiency based on EXP tuning.
LeadScrewFactor = Lt/(2*pi*e); %approximate equation relating force on a lead screw to
                              %torque on that lead screw

DS = .2;
Kp = 2.35;
Ki = .28;
sim('VolumeBasedControl'); %run simulation
save('PISin_Sim','PWM','Strain','tout');

figure(1) %below is plotting
subplot(2,1,1)
plot(tout,Strain,'k')
hold on
for i = 1:length(tout)
    Dsarray(i) = DS;
end
plot(tout,Dsarray,'r--')

h=fill([0,20,20,0],[DS*.98,DS*.98,DS*1.02,DS*1.02],['red', 'LineStyle', 'none']);
h.FaceAlpha=0.3;
axis([0 40 0 .25])
title('Model Response to PI Control')
xlabel('Time (s)')
set(findall(gca, '-property', 'FontSize'), 'FontSize', 14);
set(findall(gca, '-property', 'LineWidth'), 'LineWidth', 3);
legend({'Model Response', 'Desired Output'}, 'Location', 'SouthEast')
grid on
subplot(2,1,2)
plot(tout,PWM)
axis([-1 1])
grid on
```
A.4 PI Simulink
function mPa = SyringeRateofChange(rads)
mPa = rads * 52354/(2*pi);
A.5 MRAC simulation script

```matlab
1 clear 2 br = .5; % for reference system 3 k0 = 2; % initial gain conditions 4 k0 = -2; 5 gx = 900; % gain rate of change 6 gr = 1000; 7 R = 4.5; % Measured 8 La = 0.2835; % Inductance measured 9 kt = .24; % motor torque constant 10 kbt = kt; % motor back emf constant 11 B = .0001; 12 J = 1.96e-6; % Cylinder on axis 13 S = 2300; % conversion of strain to pressure tuned experimentally 14 SyringeArea = (.013^2)*pi; % Area of the syringe plunger 15 Lt = .254; % pitch of screw 16 e = .25; % Efficiency based on EXP tuning. 17 LeadScrewFactor = Lt/(2*pi*e); % approximate equation relating force on a lead screw to torque on that lead screw 18 Fs = 0; 19 20 DS = .2; % Desired strain 21 sim('VolumeMRAC') % run simulation 22 clf 23 subplot(2,1,1) 24 plot(tout,Strains(:,2),'k') 25 hold on 26 plot(tout,Strains(:,1),'r--') 27 title('Model Response to MRAC') 28 xlabel('Time (s)') 29 ylabel('Strain (cm/cm)') 30 grid on 31 subplot(2,1,2) 32 plot(tout,PWM) 33 axis([0 120 -1 1]) 34 grid on 35 title('PI Control Output') 36 xlabel('Time (s)') 37 ylabel('PWM') 38 grid on 39 save('MRACRamp_Sim','PWM','Strains','tout','R','X');```

A.6 MRAC Simulink
function nPa = SyringeRateofChange(rads)
nPa = rads*0.0254/(2*pi);
function [theta_rd, theta_xd] = fcn(ead, r, x, br, gx, gr)

theta_xd = -gx*ead*r*sign(br);
theta_rd = -gr*ead*r*sign(br);
function u_mrac = fon(x, r, theta_x, theta_r)
##codegen

    u_mrac = theta_x*theta_r*r;

function xrd = fcn(xr,input,br)
\#codegen

\texttt{at} = -\texttt{br};
xrd = \texttt{ar}^*\texttt{at} + \texttt{br}^*\texttt{input};
### A.7 Mathematical implementation of fixed-end model

```matlab
function [S] = Disp2Strain(d)
S = real ((-0.090075 - 0.15602i)/(5.6883*d + ((5.6883*d - 0.010049)^2 - 0.0058467)^(1/2) - 0.010049)^(1/3)...
- (5.6883*d + ((5.6883*d - 0.010049)^2 - 0.0058467)^(1/2) - 0.010049)^(1/3)*(0.5 - 0.86603i) + 0.75309);
end
```