Hand Shaking Soft Robot

Shadmehr Gitiforooz

March 1st, 2021

ME 125EH - Winter 2021

University of California, Santa Barbara

Abstract:

This paper discussed the design, fabrication, and modelling of a hand shaking orbot that was built off concepts introduced in Professor Elliot Hawkes's Winter 2021 ME 125EH class. The robot is meant to sense a human hand using a capacitance sensor and actuate a pneu-net soft muscle which engulfs the human hand that was sensed, thereby "shaking" the hand. The robot was unable to be built to support five fingers, but a one finger system serves as a proof of concept. This robot takes heavy inspiration from the past works in the soft robotics field, and is meant to serve as an introduction which excites people about the fast emerging field of soft robotics.

Intro:

The goal of this robot is to tie together multiple focuses from the soft robotics field. Soft muscles combined with soft sensors can serve to be the future of prosthetics and robots that interact with humans. To build a hand shaking robot, the radius of curvature of the soft muscles needed to be much smaller than those explored in Lab 3 - Soft Gripper of ME125EH [1]. In order to do so, pneu-net pneumatic muscles were selected as they only required a 3D printed mold and offered tremendous flexibility. Inspiration was taken from Yazhou Wang's 2019 research [2]and Yilin Sun's 2019 research work [3].

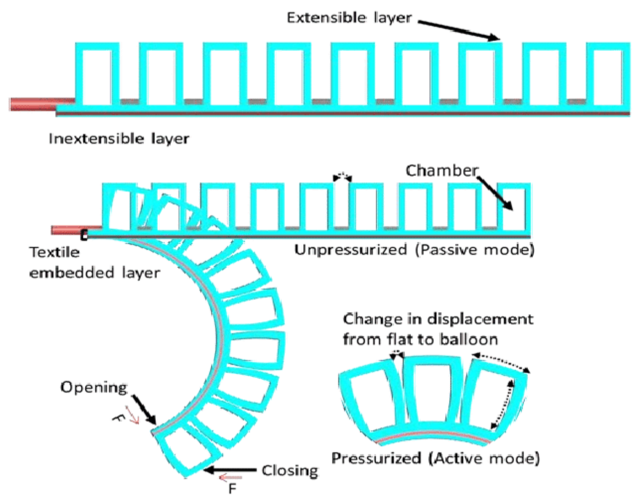

Figure 1: Pneu-net muscle [4]

Additionally, the integration of soft sensors was inspired by Thomas George Thuruthel's work [5] using soft muscles and soft sensors to mimic human perception and create a neural net of sensors. Seeing the importance of soft sensors, Lab 6 - Soft Capacitance of ME125EH [6] served as a key building block for this system's sensing capabilities.

Design:

In order to create the desired radius of curvature of about 5cm, the optimal dimensions for a pneu-net soft muscle were required, thus. Wang's work dictated the dimensions of mold and muscle. This design containing multiple cells of air is preferable to the actuator design explored in Lab 3 as the force from the pressurized air in this more intricate design is further from the base of the actuator. This will be shown in the Modeling section. In order to pressurize the muscles or "fingers," a pneumatic pump must be used. Air flows into the finger when a pneumatic solenoid is triggered.

In order to trigger the solenoid valve, the sensor must detect a hand. When the sensor detects a change in capacitance, a Teensyduino sends 12V to the solenoid.

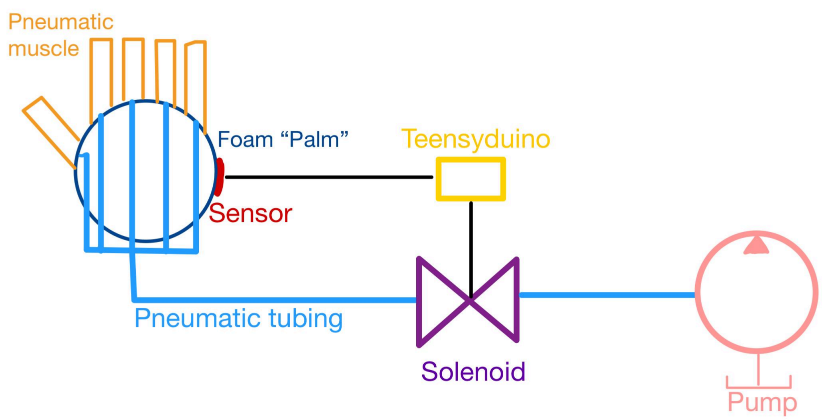

Figure 2: Schematic of hand's systems.

The code, circuit, and Teensyduino serve as the brains of the operation. The Teensy reads the capacitance readings from the sensor and when it reads a value deemed to be a light human touch, 5V is output. Utilizing a transistor, when 5V is output from the Teensy, a different circuit closes. This closed circuit contains the 12V battery pack and the 12V solenoid, thus opening the solenoid. Due to difficulties getting consistent capacitance readings during a sensor touch, the 5V output is maintained for three seconds, keeping the fingers actuated for three seconds as well. The code is outlined in Appendix A.

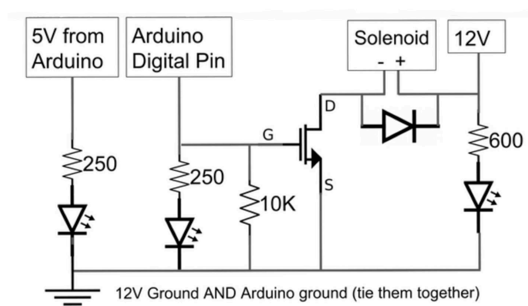

Figure 3: Circuit for all systems integrated into build. LEDs are included in this circuit as reference for user. [7]

Fabrication:

A three piece mold (Figure 4) of the pneu-net actuator was 3D printed. The mold is filled with Dragon Skin10. The Dragon Skin is heated with a blow dryer to cure. A flat, rectangular mold (A) was printed as well to serve as a mold for the base of the actuator. The base mold is filled and a piece of fabric is placed over the curing Dragon Skin to act as a strain limiter for the bottom of the actuator. The lower (B) and upper (C) halves of the top mold are placed on top of one another and filled with Dragonskin. After being removed from the molds, the base and top of the actuator are connected with more Dragon Skin.

A) base mold B) lower half of top mold C) upper half of top mold

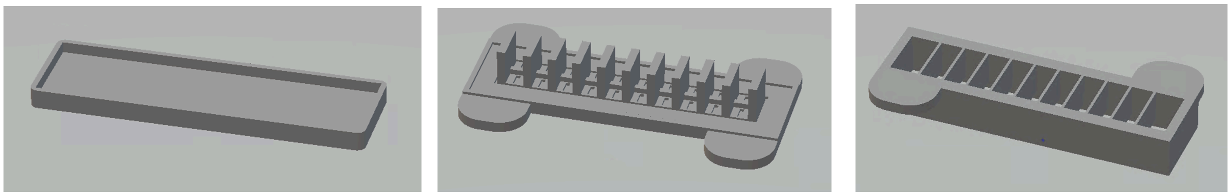

Figure 4: Three piece actuator molds

A hole about 3 mm in size was poked into the end of the finger. 6mm pneumatic tubing was pushed through the hole. The elasticity of the Dragon Skin resulted in it creating an airtight seal around the tubing. The finger was placed in a foam "palm" and pneumatic tubing connects it to the pump and solenoid. The capacitance sensor uses conductive fabric uncased by rubber strips. The fabrication methods outlined in Lab 6 from ME 125 were used.

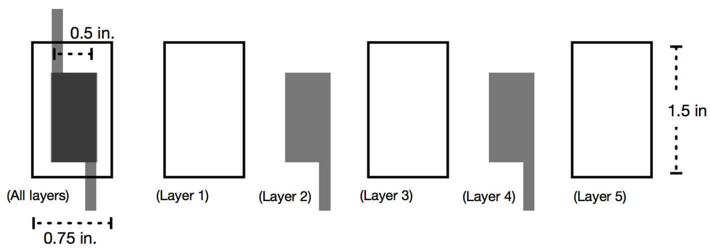

Figure 5: Soft sensor design [6]

In making the Dragon Skin actuators, fabrication errors were encountered. Four Dragon Skin muscles were poured. Two actuators had leaks which kept the muscle from bending to its potential. This could be a result of an inconsistent pour or from forceful mold release attempts. One actuator had a clogged air channel, not allowing it to bend. This was likely caused when connecting the top and base parts of the actuator with Dragon Skin Only one actuator was built as it was designed. Thus, this iteration of the robot only sports one finger. (Actuator 3 had its leaks successfully plugged after the demonstration and is used in the radius of curvature measurement.)

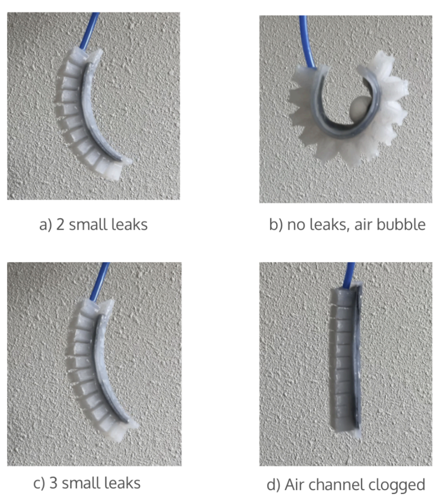

Figure 6: All actuators built

Modelling:

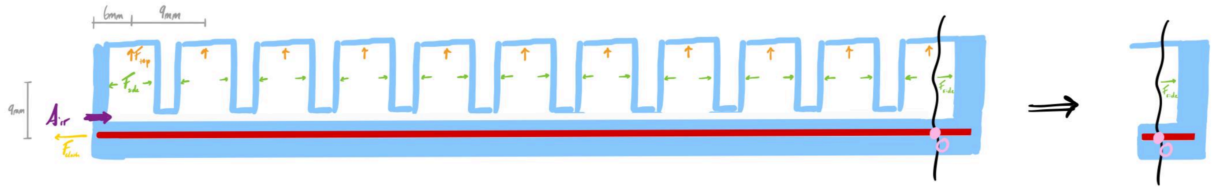

Figure 7: Free body diagram of actuator

The radius of curvature model is governed by beam bending. The radius of curvature will be the same over one cell as it will be over all eleven cells. Thus, this derivation will consider just one of the eleven cells of air. The base of the actuator can be treated as a solid beam. The bending moment will therefore be calculated from force acting on the beam causing the bend.

The bending moment, M, relates to the radius of curvature, R: [8]

$$M = \frac{EI}{R} \Rightarrow R = \frac{EI}{M} \tag{1}$$

Where $E$is the Young's modulus of Dragon Skin 10 and I is the moment of inertia.
To find M, the force on the side wall,F_{side}, must be calculated:

Pressure supplied: $P \approx 16\,psi = 110.3\,kPa$
Area of side wall:
$A_{side} = (10mm)(14mm) = 1.4\,x\,10^{-4}\,m^2$

$$F_{side} = P\,A_{side} \tag{2}$$
$$F_{side} = 15.44\,N$$

Then the moment at the new origin, M, can be calculated: (The force from the cloth can be ignored in this calculation as that force is applied at the new origin.)

$$M_0 = F_{side} \bullet y \tag{3}$$
$$M_0 = 15.44\,N \bullet 9mm = 0.139\,Nm$$

Equation (1) is now revisited:

Young's Modulus: [9] $E = 22\,psi = 151\,kPA$
Moment of Inertia:
$I = \frac{1}{12}bh^3 = \frac{1}{12}(20mm)(15mm)^3$

$$I = 5.63\,x\,10^{-9}m^4$$

$$R = \frac{EI}{M}$$
$$R = 6.11\,mm$$

Combining equations (1), (2), (3), we arrive at an equation that will let us plot radius of curvature as a function of pressure supplied:

$$R = \frac{EI}{M} \Rightarrow R = \frac{EI}{(P\,A_{side})\,y} \tag{4}$$

This model confirms our assumptions that as higher pressure is supplied, the radius of curvature decreases and the finger bends more.

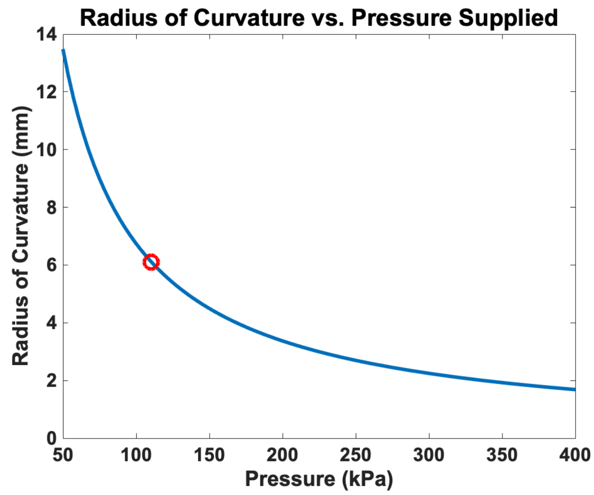

Figure 8: Radius of curvature of muscle as a function of pressure supplied

Results:

The actual radius of curvature was determined by measuring the diameter of the circle conveniently formed by the pneumatic muscle. Using this method, the radius of curvature was measured to be 1.5 cm. However, it was noticed the actuator's bending is being stopped by itself when it bends into a circle. Thus, I kept most of the straight and the radius of curvature was unsurprisingly much lower. The new Radius of curvature was measured to be 9 mm. This is very close to our predicted radius of curvature (6.11 mm). The difference in predicted and actual radius of curvature is likely due to air bubbles in the actuator as a result of an inconsistent pour. (Actuator (c) had its leaks successfully plugged and was used in this measurement as actuator (b) has an air bubble on its bottom side.)

Figure 9: Actuator when radius of curvature is restricted by itself

Figure 10: Actuator when allowed to bend to its potential

When assembled, the system worked smoothly. A demonstration video can be viewed here. Due to fabrication issues previously discussed, only one finger is implemented into the demonstration. Leaks in flawed actuators prevented the reliable actuator from bending to its potential when alone. Another issue was encountered during circuit assembly, when human error resulted in a Teensyduino receiving 12V and no longer becoming functional.

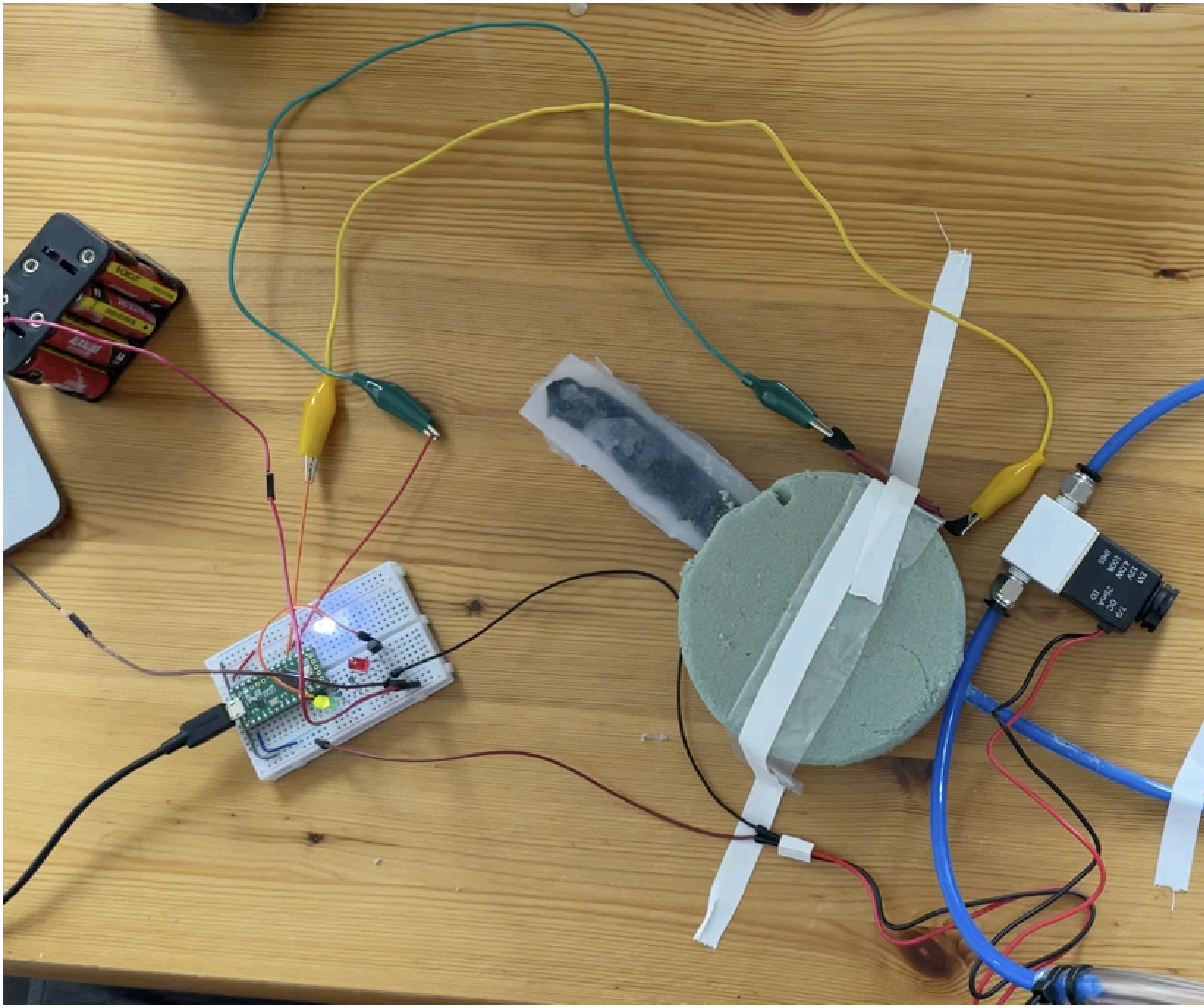

Figure 11: Hand Shaking Robot from above when all systems connected.

Conclusion:

Although I was not able to realize my vision of a five fingered hand shaking robot, this project was successful in that it showed the functionality of the system. Additionally, I was able to diagnose the issues that prevented the intended scenario from playing out. I look forward to acquiring more resources and fabricating five functioning muscles and building a full hand. Moving forward, I would also like to acquire and use a functioning pressure gauge that will allow me to plot a 'Radius of curvature of muscle as a function of pressure supplied' graph to better compare the muscle to the model. Overall, this project is seen as a success in my eyes as it intertwines circuits, soft muscles, pneumatics, and soft sensors, but I look forward to even bettering the robot.

Acknowledgements

I thank Professor Hawkes and Brian Dincau for their guidance throughout the quarter. Additionally, I would like to thank the Hawkes Lab for generously supplying resources for the robot, Matthew Devlin, and Trevor Marks as well.

Resources

[1] Hawkes, Elliot. Gauchospace, 2021, Lab 3 - Soft Gripper.

[2] Wang, Yazhou. (2018). New structure of pneumatic networks actuators for soft robotics. The Journal of Engineering. 2019. 10.1049/joe.2018.9023.

[3] Yilin Sun, Qiuju Zhang, Xiaoyan Chen, Haiwei Chen, "An Optimum Design Method of Pneu-Net Actuators for Trajectory Matching Utilizing a Bending Model and GA", Mathematical Problems in Engineering, vol. 2019, Article ID 6721897, 12 pages, 2019. https://doi.org/10.1155/2019/6721897

[4] Murali Babu, Saravana Prashanth & Sadeghi, Ali & Mondini, Alessio & Mazzolai, Barbara. (2019). Antagonistic Pneumatic Actuators with Variable Stiffness for Soft Robotic Applications. 10.1109/ROBOSOFT.2019.8722803.

[5] T. George Thuruthel, B. Shih, C. Laschi, M. T. Tolley, Soft robot perception using embedded soft sensors and recurrent neural networks. Sci. Robot. 4, eaav1488 (2019).

[6] Hawkes, Elliot. Gauchospace, 2021, Lab 6 - Soft Capacitive Sensor.

[7] "Control Pneumatic Cylinder with Arduino." *YouTube*, YouTube, 29 Dec. 2019, www.youtube.com/watch?v=iXNNjtK5hp8&list=WL&index=11&t=158s.

[8] *Beam Bending*, Continuum Mechanics, www.continuummechanics.org/beambending.html.

[9] "Dragon Skin™ 10 VERY FAST Product Information." *Smooth*, www.smooth-on.com/products/dragon-skin-10-very-fast/.

A Arduino Implementation: `Hand_Shake.ino`

Code used to take in sensor readings and actuate solenoid valves with Teensyduino:

```
int solenoid = 11;
int led = 13;

void setup()   {
   Serial.begin(38400);
   pinMode(solenoid, OUTPUT);
 }

 void loop()
 {
   // Read and store capacitance values
   int a = touchRead(A3);
   int b = touchRead(A2);
   // a and b are capacitance values in physical units, with
   //    1000 = 20 pF (pico Farads)

   // displaying two values on same plot
   Serial.print(a); Serial.print(","); Serial.println(b);

    if (b > 1600){
    digitalWrite(solenoid, HIGH);   // open solenoid
    digitalWrite(led, HIGH);        // turn on reference LED
    delay(3000);
    }
    else {
    digitalWrite(solenoid, LOW);    // close solenoid or keep solenoid closed
    digitalWrite(led, LOW);         // turn off or keep off reference LED
    }

   delay(100); // sampling at about 1/(100 ms) = 10 Hz

 }
```

B MATLAB Implementation: HandShake_model.m

Code used to plot 'radius of curvature vs. pressure supplied' model:

```
clc; clear variables; close all;

E = 151;             % kPa
y = 0.009;           % m
I = 5.625*10^(-9);   % m^4
A = 1.4*10^-4;       % m^2

P = linspace(50, 400);  % initialize pressure range from 50 kPa to 400 kPa

R = E*I./P/y/A*1000;       % Eqn discussed in paper (1000mm/1m)

plot(P, R); hold on;
xlabel("Pressure (kPa)")
ylabel("Radius of Curvature (mm)");
title("Radius of Curvature vs. Pressure Supplied");
plot(110.3,6.11,'ro');

fixfig;
```