Validating a Finger Rehabilitation Device for Hemiparetic Stroke Patients to Quantify Patient Strength

University of Waterloo

Department of Systems Design Engineering

Group #10

Connor Irvine (20947051)

Dylan Cho (20948275)

Ivan Yu (20942405)

Sydney Thompson (20914113)

Vanessa Mac (20941183)

BME 361: Biomedical Engineering Design

April 12, 2024

Table of Contents

Situation of Concern

Stroke remains a major cause of adult disability despite various advancements in medical technology [1, 2]. Consequently, a large percentage of survivors experience limited independence. A common complication contributing to this loss of function is hemiparesis: characterized weakness on one side of the body caused by reduced motor control and impaired voluntary movement [2, 3]. As a result, these individuals experience the loss of upper body control, limited grip strength and an inability to grasp objects [4]. The grasping motion is a precursor to numerous hand gestures, thus regaining strength of this movement is essential for standard hand functionality [5]. Additionally, grip training in the early post-stroke period can improve cognitive function by promoting white matter remodeling and strengthening network connections [6].

The severity of this paralysis is related to the intensity of the stroke, which subsequently influences the ensuing physical therapy [7]. These rehabilitation programs commonly incorporate repeated, targeted exercises to retrain muscles for daily tasks [8]. Through the use of external elastic resistance, stroke patients can regain lost function and then strengthen the impacted neural and muscular pathways [7]. Rubber cords are a method for providing the required resistance for these exercises [9]. In addition to their elastic properties, when carbon black is incorporated into the material structure, electrical current can be transmitted. As seen in Fig. 1 below, when the cord is relaxed, the internal structure is highly coiled, allowing carbon black particles to aggregate and conduct an electric current. When the cord is stretched, the cross-linked chains become uncoiled, spreading apart the carbon particles, and increasing the cord's resistance. By measuring a change in resistance and current, applied forces can be quantified.

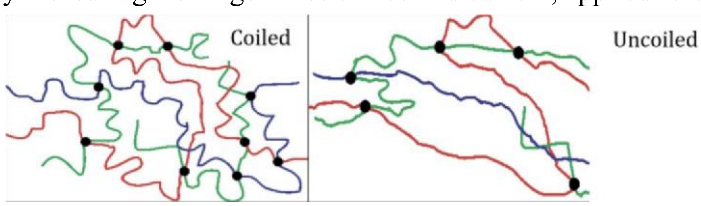


Fig. 1 Cross-linked chains of rubber under relaxed (left) and taught (right) conditions [10].

Quantifying the force applied by the patient's fingers can be an effective method for tracking progress, allowing the therapist to continuously and accurately adjust the users' rehabilitation protocols. At the beginning of a therapy program, hemiparetic patients will have limited ability to curl their fingers or may not be able to follow the proper path of motion [11]. Therefore, a motor can be employed to aid the user in completing the desired curling motion. As the exercise is repeated over time, individuals can gradually regain strength and increase independent motion. Applying the rubber cord as a connection between the motor and finger allows for assisted motion. Furthermore, differentiating between forces applied by the user versus the motor can help the therapist understand how much progress the patient is making. Ideally, the work applied by the patient should increase overtime as they become able to complete the hand gripping motion with less motor assistance. It is of interest to design a device that can compute the percentage of work the motor versus the patient does to complete a full finger curl.

Consequently, there is a need to design a useful device that will assist hemiparetic stroke patients with practicing the motion of curling their fingers into their palm, while providing feedback to the therapist about finger strength. This will aid the user with relearning how to grasp objects.

Needs Assessment

A needs assessment was completed to empathize with the user's needs and systematically identify functions. A needs hierarchy, as seen in Fig. 2, was created based on the "Soft Exo-gloves (SEG) Design Criteria" section in [12]. The main need is a userful finger exercise rehabilitation device. It was determined that the high level functions to integrate include being safe, providing adjustable assistance, being effective, and being marketable. The needs hierarchy was based on this article as this most closely matched the users, stroke patients, and the flexible design of cord. Based on the needs hierarchy, functions were generated to be used in the concept generation stage: assist with finger curling, customized to the patient's anatomy, customized to the patient's abilities, send detailed feedback to the therapist, and measure progress metrics (force, work, etc). Certain needs which are boxed in Fig. 2 using red and blue were previously targeted in Case Study 2/3 and are described in their respective report sections.

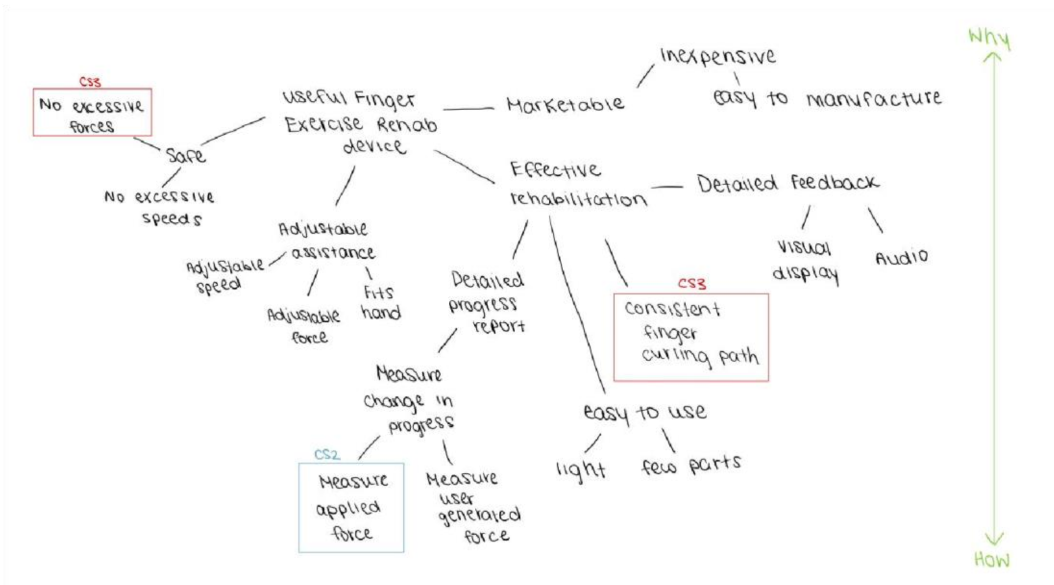


Fig. 2 Needs hierarchy based on *Design Criteria of Soft Exo Gloves for Hand Rehabilitation-Assistance Tasks* [10].

Based on the needs hierarchy, the team determined that finding a useful metric for patient progress was key to the design. Measuring both force and distance travelled by the user's finger can provide more information to the therapist. Therefore, one practical metric is work, as it combines magnitude of force and distance. This can also provide comprehensive information for the therapist to analyze forces applied over different distances or perform efficiency/energy measurements. Additionally, a functional decomposition in Fig. 3 was generated. The main function identified was to quantify work completed by the user versus the device. Then, the main function was then broken down into 4 subfunctions. Although the functions used for concept generation are based on the needs hierarchy, the functional decomposition was an important tool for the team to understand the user's general workflow.

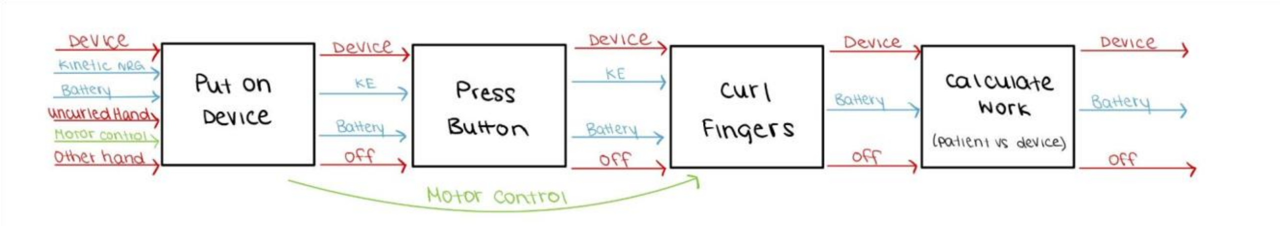


Fig. 3 Functional decomposition for main need: quantify work completed by the user versus the device. Blue = energy, Green = information, Red = material.

Definition of Criteria and Constraints

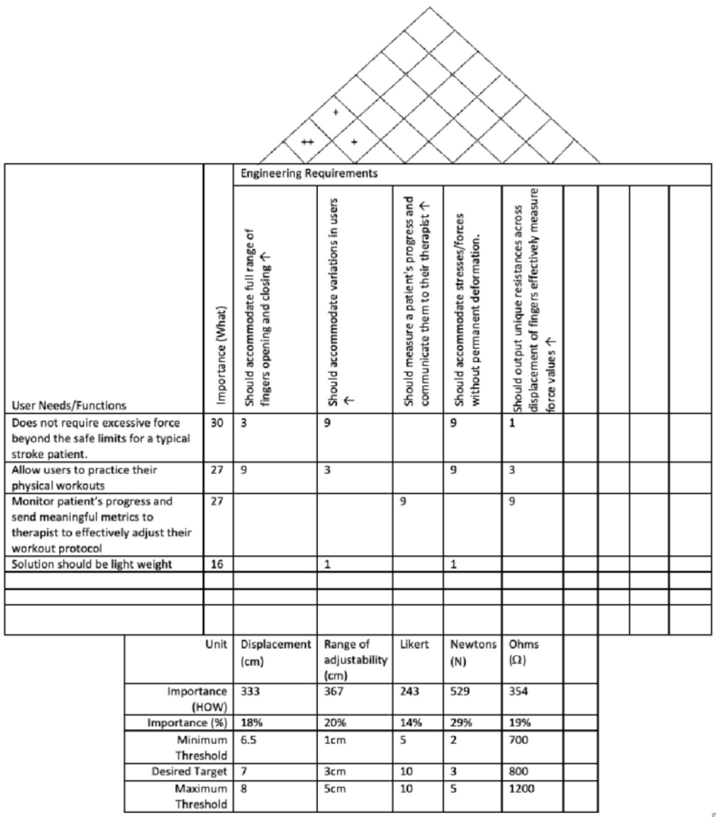
Table 1. Engineering Requirements

Requirement	Sub Requirements
R1. Should accommodate full range of fingers opening and closing	● R1.1 Should accommodate displacement of 6cm to enabled effective rehabilitation (e.g. From fingertips to bottom of palm) ● R1.2 Should accommodate stresses/forces without permanent deformation ● R1.3 Cables minimum length should accommodate various finger length ● R1.4 Should provide adequate support using a motor to support workout. ● R1.5 Should produce enough resistive forces to improve rehabilitation/achieve rehabilitation goals.
R2. Should accommodate variations in users	● R2.1 Should accommodate various speeds ● R2.2 Should accommodate variation in hand size ● R2.3 Should minimize stress on hand (i.e. decrease pain, discomfort)
R3. Consistently quantify work done by the patient during finger curl.	● R3.1 Should output unique resistances across displacement of fingers in order to effectively determine force values ● R3.2 For a given closing speed, when the user is applying no force, the device should output a unique force-distance graph. ● R3.3 Consistently quantifies the amount of work performed by the user versus the device.

The systematic tools created in the needs assessment were used to develop engineering requirements and constraints. Reviewing the requirements laid out in Table 1, many of the requirements have been investigated in Case Study 2 (R1.1, R2.3) and Case Study 3 (R1.2. R3.1).

Using a QFD in Fig. 4, it was found that metrics and patient monitoring were high priority requirements and as such, the focus of this report will be on R3.2 and R3.3. Ultimately, these requirements contribute to the main need of measuring patient strength progression. In order to collect useful data for the physician, the device must consistently quantify work done by the patient versus the motor during a finger curl (R3). R3.2 says "for a given closing speed" since the user may adjust the speed of the device depending on their comfort or if they want to practice faster/slower movements. Therefore, the device and viscoelastic cord should perform in predictable ways at different stretching speeds.

From the QFD, metrics were identified for the associated requirement. In line with R3.2: for a given motor speed when the user is applying no force, can the device output a consistent and unique force-distance graph? The force-distance calibration curve for a given speed should provide a good fit ($R^2 > 0.7$) for force-distance data collected when the user applies no force for the duration of a finger curl. This force-distance graph has an area under the curve (AUC) which is equivalent to the total work done by the motor when the patient has no strength, illustrated in Fig. 5. The trials from which the force-distance calibration curve is produced should have a consistent AUC representing total work such that the standard deviation of total work is ±10% from the mean. In other words, the main goal of an experiment to meet this requirement is to determine if we can consistently quantify the amount of work the patient can accomplish. In order to do this, we must first determine if we can consistently quantify the amount of work the motor accomplishes when the patient does no work.



User Needs/Functions	Importance (What)	Engineering Requirements: Should accommodate full range of fingers opening and closing ↑	Should accommodate variations in users ←	Should measure a patient's progress and communicate them to their therapist ↑	Should accommodate stresses/forces without permanent deformation.	Should output unique resistances across displacement of fingers effectively measure force values ↑
Does not require excessive force beyond the safe limits for a typical stroke patient.	30	3	9		9	1
Allow users to practice their physical workouts	27	9	3		9	3
Monitor patient's progress and send meaningful metrics to therapist to effectively adjust their workout protocol	27			9		9
Solution should be light weight	16		1		1	
Unit		Displacement (cm)	Range of adjustability (cm)	Likert	Newtons (N)	Ohms (Ω)
Importance (HOW)		333	367	243	529	354
Importance (%)		18%	20%	14%	29%	19%
Minimum Threshold		6.5	1cm	5	2	700
Desired Target		7	3cm	10	3	800
Maximum Threshold		8	5cm	10	5	1200

Fig. 4 Quality Function Deployment of finger rehabilitation device.

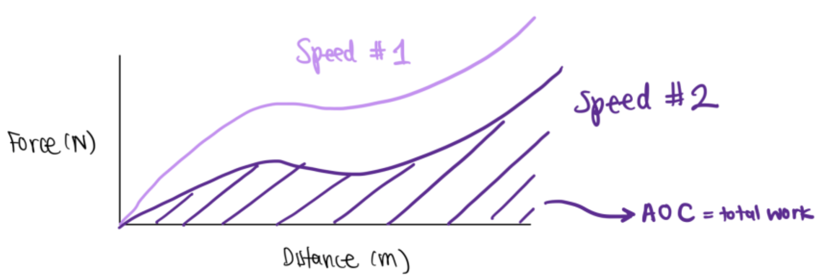


Fig. 5 Example force-distance calibration curve and area under curve (AUC) representing total work done by the motor, which should be generated by the prototype.

After addressing R3.2, in line with R3.3: can the device consistently quantify the amount of work performed by the user versus the device? The main idea behind this requirement is illustrated in Fig. 6. After achieving R3.2, the work done by the patient can be measured as the difference in total area under the force-distance calibration curve. More specifically, total work done by the patient is the area under the force-distance curve when the patient does no work subtract the area under the force-distance curve when the patient does work. In order to consistently quantify the amount of work done by the patient, a regression between difference in total work and the total work/force applied by the user should have $R^2>0.7$ and $\rho>0.7$ (where ρ is the Spearman's rank correlation coefficient). The main goal of an experiment to meet this requirement is to determine if we can consistently quantify the amount of work the patient did compared to the amount of work the device did.

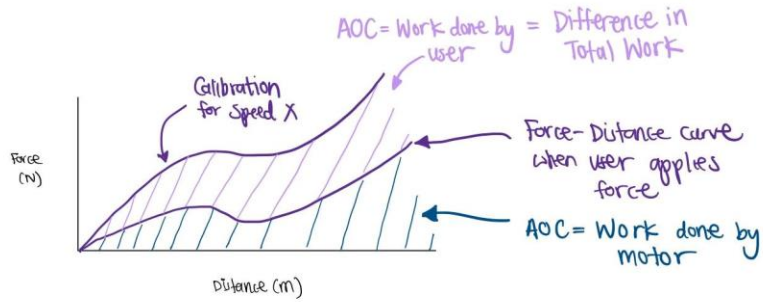


Fig. 6 The prototype should calculate work done by the user by taking the difference in AUC between the force-distance curve when the user does no work and the force-distance curve when they do work.

Various constraints were predefined by the course instructor such as using a parallax continuous rotation servo motor [13] and a maximum of 2 rubber cords. Various constraints on the design discovered through Case Study 2/3 are described in the following sections.

Case Study 2

The experiments from Case Study 2 yielded useful information to guide the design of prototypes and experiments for this project. Investigating R1.2 (should accommodate stresses/forces without permanent deformation) it was found that the rubber cord experiences permanent deformation at 70% elongation and that the highest levels of stress concentration were around where the cord is attached to the hooks. As such, experiments in Case Study 2 were only carried out to a maximum of 60% elongation. The key mechanical properties of the rubber cord are illustrated in the stress-strain curve in Fig. 7 [14].

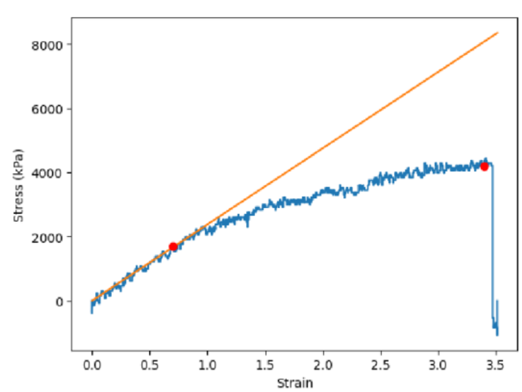


Fig. 7 Stress-strain curve from CS2 Experiment 1 failure test. Yield strength (strain=0.7, stress=1700kPa) and ultimate/fracture stress (strain=3.4, stress=4200kPa) are marked in purple dots [14].

In Fig. 8, it was found that there is a quadratic resistance-force relationship for a 4 cm cord. Trials were completed up to a maximum of 60% elongation such that the forces applied were always less than the force at permanent deformation (~5.3N). The range of cord resistances that can be measured in this range is ~400-800Ω. ANOVA and LSD analysis with percent elongation as treatments (4 treatment groups: 0%, 20%, 40%, and 60%) and cord force/resistance/voltage as the dependent variable showed that the data was most sensitive to 0-40% elongation. This indicates that larger elongations from 40-60% had insignificant effects on cord resistance even if permanent deformation did not occur [14].

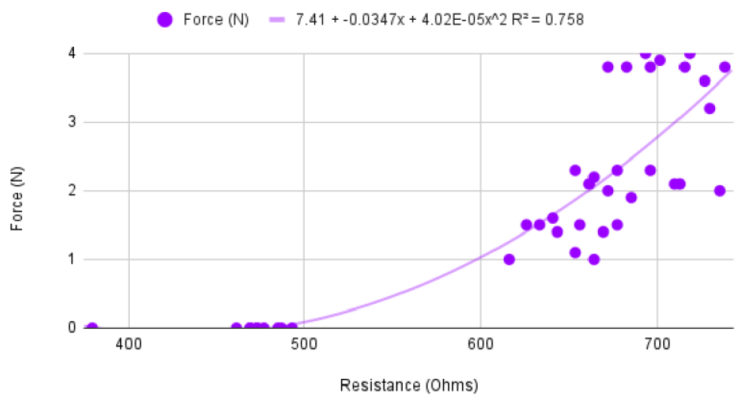


Fig. 8 Force versus the resistance of the rubber cord [14].

Case Study 3

Case Study 3 aimed to look at requirement R1.1 (qualitatively achieve finger curling motion). As shown in Fig. 9, the string was attached to one of three points on the metal finger model marked with dashed red lines (0 cm, 1 cm, 2 cm from the end of the finger). The string was then pulled at various angles (25°, 45°, and 65° relative to the palm) to simulate the rubber cord curling the finger via a motor [15].

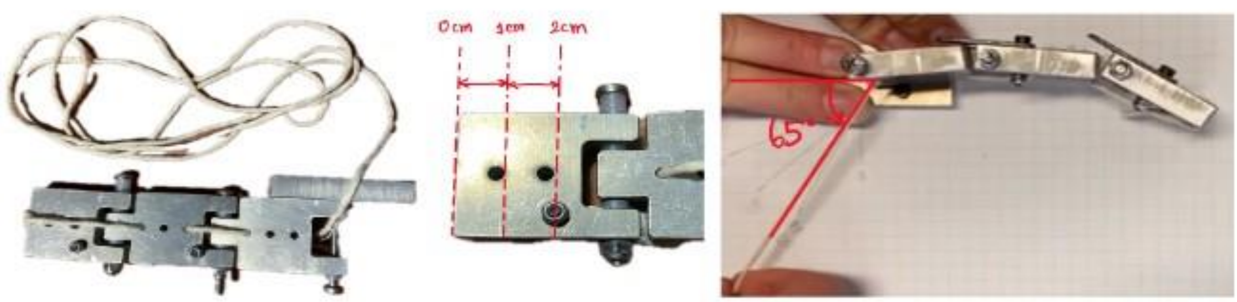


Fig. 9 Finger prototype mechanically actuated by string. The string was attached to one of three points marked in red using the holes in the distal finger segment (0 cm, 1 cm, 2 cm from the end of the finger). The right-most image shows the string being pulled at 65° relative to the palm with the string attached at 0 cm from the end of the finger [15].

Case Study 3 also aimed to investigate R2.3 (minimize stress on the finger for maximum comfort). Further investigation in SolidWorks of stress concentrations in Fig. 10 showed that the maximum stress is minimized when the application point is 1 cm. The string's angle did not affect the maximum stress. However, based on the kinematics experiments in Fig. 9, the optimal angle was to pull from the middle of the finger at an angle closer to the arm (between 25 and 45 degrees). This reduced friction on the string and allowed for smooth controlled finger curling. As such the results from Case Study 3 helped guide the prototype design in deciding what angle and distance the device should pull the patient's finger at [15].

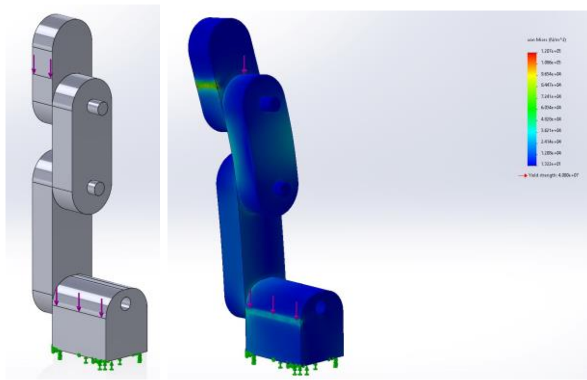


Fig. 10 Assembly set up and maximum stress (120668.67 Von Mises) for application point 1 cm from the distal end and angle 25°. The maximum stress is minimized when the application point is 1 cm [15].

Concept Generation

Table 2. Morphological chart for concept generation. Purple boxes indicate selected means

Functions	Means			
Assist finger curling	Motorized mechanism	Elastic bands	Springs	Actuators
Customizes to patient anatomy	Adjustable straps	Modular components	Moldable materials (e.g. thermoplastic)	3D scanned and printed
Customizes to patient's ability	Adjustable resistance levels	Adaptive algorithm	User controlled settings	Range of motion adjustments
Send feedback to therapist	Bluetooth connectivity	Mobile app interface	Data logging	Remote monitoring
Measure strength progression	Range of motion	Work performed	Duration of use	Force applied

Case Study 2/3 informed the engineering constraints and aided the team's understanding of how to use the rubber cord in the prototype. To link the engineering requirements/constraints with the concept, a morphological chart in Table 2 was created based on the needs hierarchy. The five selected functions include: assisting in finger curling, customizable to the patient's anatomy, customizable to the patient's ability, able to send feedback to the therapist, and be able to measure the patient's strength progression. Each function was given four potential means. The means for the first two functions (assisting in finger curl and customizable to patient anatomy) were selected due to having 3D printing materials and a parallax continuous servo motor readily available. A 3D printer also enables rapidly printing multiple iterations. One mean that was also heavily considered in customizability to patient anatomy was adjustable straps, which was implemented as well. To achieve customizability for patient ability, we decided to go with the option for the user to be able to control settings themselves. This would be implemented via Arduino code, in which the user could adjust the speed of the motor, as well as the total distance the motor actuates the elastic cord. This would allow users who have less strength to be able to receive more motor assistance depending on their needs. Customizing distance the motor travels is also essential to helping the user complete the full finger curl. Finally, the main function of the prototype was determined to be able to log and track the work done by both the patient and the motor.

The concept for the prototype involved a design with three attachment points, consisting of a 3D printed "half ring" shape on the proximal/intermediate phalanges and a finger cap on the distal phalange, as seen in Fig. 11. To attach and ensure that the parts stayed in their intended positions, holes were incorporated into each piece for the rubber cord to feed through. A slipknot was threaded through additional holes on the side to be easily secured to the finger. A circular hole was also incorporated into each finger attachment for the elastic cord to feed through to be actuated by the motor.

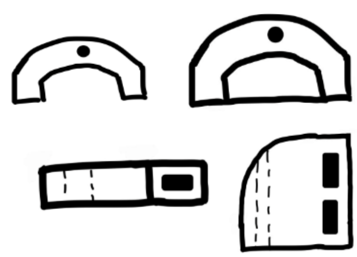


Fig. 11 Top (upper left) and side (lower left) view of the middle and lower finger attachment pieces and back (upper right) and side (lower right) view of the upper finger attachment piece.

Another important design feature was where to attach the motor to actuate the elastic cord. A 3D printed wrist attachment can be used to hold the motor in line with the finger to be actuated. A wristband design was created with holes in it to allow for a slipknot to be thread through to hold onto the patient's wrist securely. The wristband, as seen in Fig. 12, was made with a trapezoidal shape to maintain comfort and incorporates a long block for the motor to be able to screw into. The block was made to be long on purpose to allow for user customizability and optimization based on the test subject's hand anatomy.

Fig. 12 Front (left), top (middle), and side (right) views of the wristband attachment piece.

Finally, housing for the battery pack, arduino, and circuit is required. A simple box-like structure was created to be placed on the arm under the wristband attachment as shown in Fig. 13.

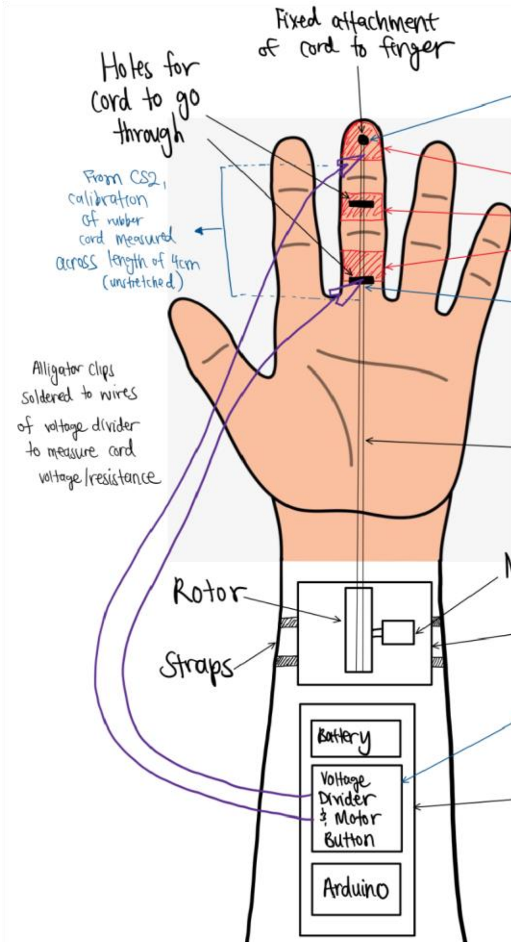


Fig. 13 Final concept sketch including housing for arduino, battery pack, and circuit

Prototyping Approach

The wristband and finger attachment pieces developed in the concept generation phase were 3D printed and assembled. The cord was threaded through the holes with slipknots at the ends to secure them. The motor was also screwed into the wristband block. The elastic cord was wrapped around the provided motor rotor, as shown in Fig. 14.

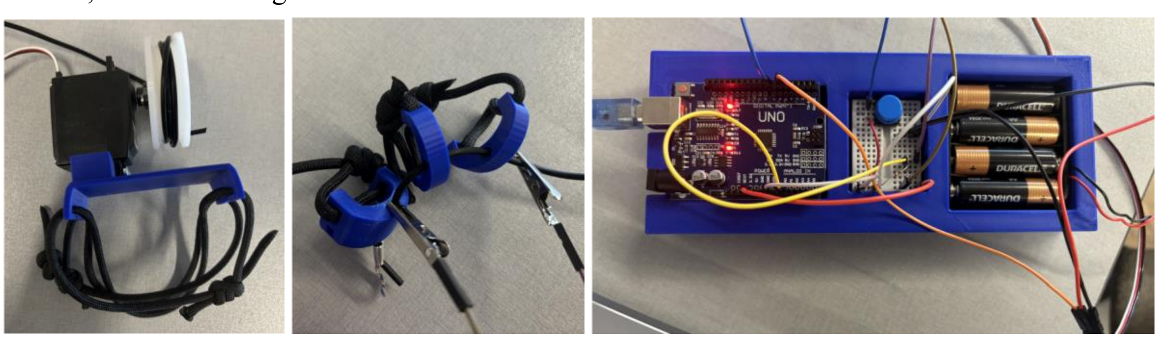


Fig. 14 Wristband attachment piece (left), finger attachment pieces with alligator clips (middle), and arduino, battery pack, and circuit housing device (right).

After creating the first prototype, we encountered a few problems. Firstly, a slip knot was insufficient to prevent the elastic cord from slipping out from the upper finger attachment piece. Therefore, the cord would be pulled by the motor without actuating the finger curl. This problem was easily fixed by attaching a metal hook to the end of the cord. Additionally, the wires connecting the elastic cord to the voltage divider circuit were difficult to secure in a consistent location on the cord since the cord is moving. This would impact the force/resistance measurements. Originally the wires were wrapped around the elastic cord and held in place by alligator clips, however the wires would sometimes break. This problem was fixed by soldering the alligator clips onto the wire then clipping the alligator clips to the cord. This solution allowed for a consistent attachment process. Note, the alligator clips were spaced apart by 4cm since the force-resistance calibration curve in Case Study 2 was generated using a 4cm rubber cord.

Another challenge was not being able to put the housing device for the arduino, circuit, and battery pack on the arm. After trying to place it on the arm, we concluded that the wires of the voltage divider circuit might get in the way of the elastic cord. It was decided for the focused prototype that the housing device would remain unattached to the user. The configuration of the final prototype is in Fig. 15.

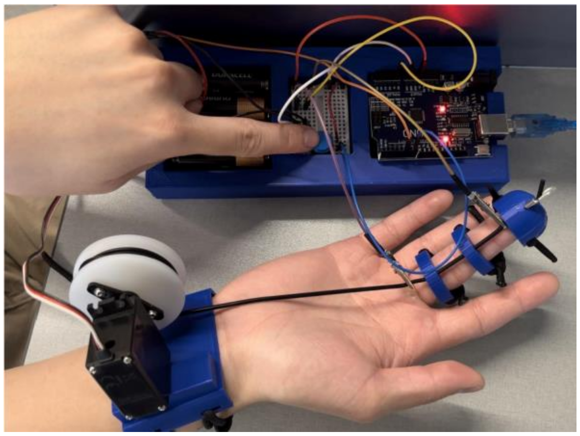


Fig. 15 Final prototype design being tested on the test subject's hand.

In Fig. 16, the design of the electrical subsystem follows a simple voltage divider powered by the arduino UNO. The voltage divider is based on Case Study 2 and utilizes the code provided in [16] to measure the variable resistance of the rubber cord. To start the motor/finger curl, a button was connected to the arduino with pull-up input to pin 7. To ensure that the value does not fluctuate, the Arduino UNO has a built in resistor with a value of about 20kΩ [17]. The value for the resistor used in the voltage divider circuit was calculated to be 580.2Ω based on 10 trials of the maximum and minimum resistance of the rubber cord. See Appendix Table A1 for the trial results and resistor calculations. The minimum resistance is measured when the cord is relaxed (4cm between the alligator clips when the hand is uncurled). The maximum resistance is measured when the cord is maximally stretched at the end of the finger curl (5.6cm between alligator clips). However the closest resistor available was a 510Ω resistor. Finally, the motor was connected to pin 5, which is activated when the button is pushed.

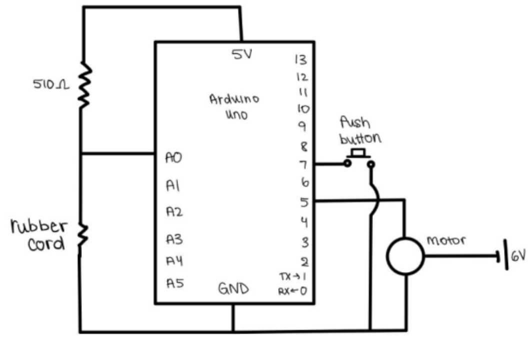


Fig. 16 Voltage divider circuit connected to and powered by the arduino UNO

In summary, the force-resistance calibration from Case Study 2 was incorporated. Also, the cord is never stretched past 60% elongation to prevent plastic deformation and the cord operates under 40% elongation for optimal sensitivity. Although Case Study 2 found that hook attachments caused stress concentrations, failure due to stress concentration is not a problem since the device operates at <60% elongation. From Case Study 3, the cord was attached in the prototype at the middle of the distal segment and pulled at an angle close to the arm (25° relative to arm).

Methodology

A p-diagram in Fig. 17 was created to organize experimental parameters, noise variables, and metrics associated with the experiments performed on the prototype. Experiments 1 and 2 are associated with R3.2 and R3.3, respectively. Prior to all experiments, the motor was calibrated using the Arduino code provided [18].

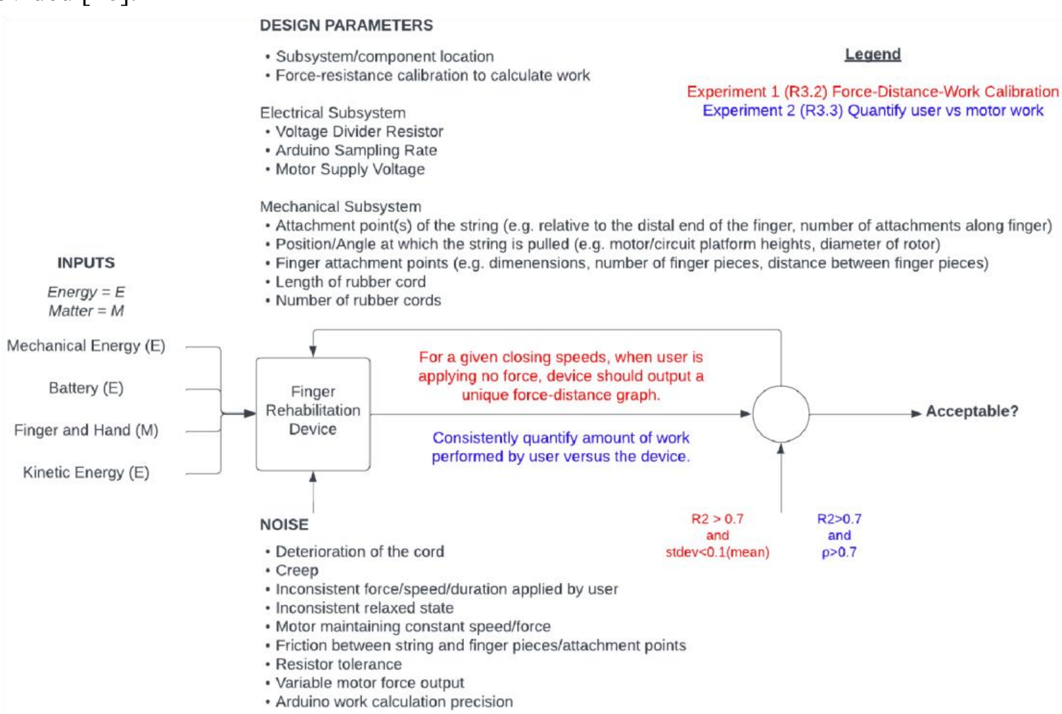


Fig. 17 P-diagram to systematically identify experimental parameters.

A. Experiment 1: Force-Distance Calibrations for Motor Speeds

Experiment 1 was a calibration experiment with 5 speed treatments set using the servo's PWM control: 1575ms, 1588ms, 1600ms, 1613ms, and 1625ms. Speed, the motor's rotation amount, and sampling interval was controlled using the Arduino code in Appendix Table B1. These speeds were chosen following exploratory testing which found that speeds greater than 1630ms could not supply enough power to consistently travel the distance of a full finger curl. Speeds lower than 1510 ms could not provide enough torque to move the finger. 10 runs at each speed was completed for a total of 50 trials (n=50). Each run started with the user's hand in an uncurled relaxed position. The device would be activated using the pull-up resistor button and the motor would curl the finger while the user applied no force, mimicking a user with no finger strength. Force-distance graphs were measured for the duration of the finger curl (i.e. finger tip touches the palm). Total work is measured by integrating a force-distance graph. The device estimates total work by sampling the cord resistance (converted to force using the CS2 force-resistance calibration) at constant distance intervals (~1.96mm) and multiplying by a constant distance multiplier (1.96mm). Note that the sampling rate (Hz) is not constant. For example, as shown in the raw data collected in Appendix C, for a speed of 1625ms, every time the cord is wound up 1.96mm, force is samplied and multiplied by distance to calculate work done by the motor over 1.96mm. This would require a sampling period of 30ms (33.3Hz). The work calculated over each interval is then summed to get total work required for the motor to curl the finger. This calculation is further illustrated in the Arduino code in Table B1 and Fig. 18 [18]. Since the cord moves the same distance each trial (i.e. the distance to completely curl the finger), each trial samples force/resistance 25 times. Replicate reset is performed between randomized speed treatments by resetting the arduino and immediately returning the cord to relaxed state (0% elongation) for at least 1.5 minutes. This also ensured creep would not confound the results. The random order was generated in Excel. If consecutive trials had the same speed by chance, the team ran an additional trial at a different speed which was not recorded to ensure the same treatment was not performed consecutively. At the start of each trial, to ensure the hand was in the same uncurled position, the force reading at 0 seconds must be <0.5N. Although the speed and rotor travelling distance were controlled in the Arduino code, small variations were observed due to the unreliable servo motor, which were noise parameters the team ignored. Uncontrolled noise parameters include the force applied by the user and the starting relaxed state of their hand. These variations were deemed negligible since the user and the team ensured they were kept as consistent as possible. The team performed ANOVA, LSD, and regression analysis on the raw data in Appendix C to assess the metrics and generate force-distance calibration curves. The data was collected by copying the values printed to the Arduino serial monitor via USB connection. A python script in Appendix Table B2 was used to convert the values to a csv format.

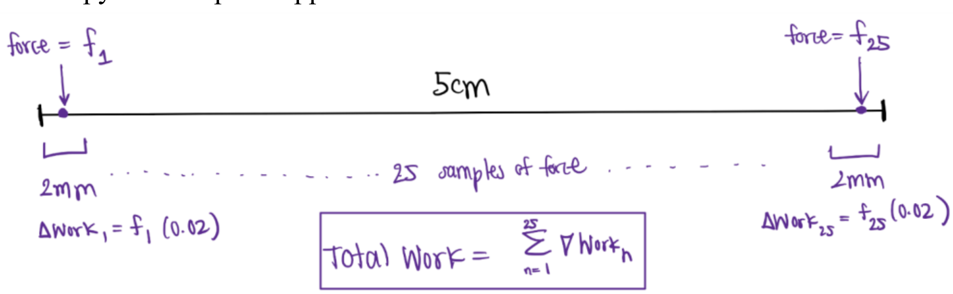


Fig. 18 Device calculates total work using force measured at constant distance intervals.

B. Experiment 2: Patient work to motor work relationship

Experiment 2 was conducted to examine the relationship between manual force exerted by the user and the change in area under the force-distance curves. Experiment 1 produced force-distance calibration curves for each speed. In real life, it is expected that the patient will try to move their finger manually. The main goal is to quantify the amount of work the user performs and how much work the motor does to assist. In theory, this manual force is going to quantifiably reduce the forces applied by the motor, and should produce a force-distance curve with area smaller than the calibration from Experiment. A lower AUC means lower total work done by the motor. Fig. 19 shows this anticipated effect, where manual force lowers the motor force-distance curve, creating a difference in work that can be measured.

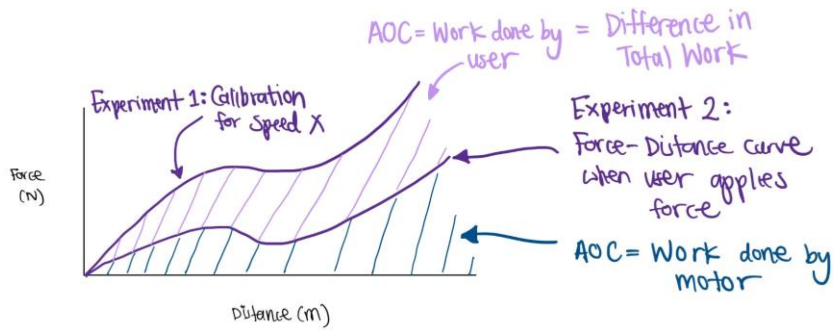


Fig. 19 Similar to Fig. 6, with additional labels showing how Experiment 1 will produce a force-distance calibration curve and Experiment 2 will produce a force-distance curve with smaller area due to the user doing work to curl their finger.

From Experiment 1's results, which is described in detail in the following section, a PWM speed of 1575 ms produced the most accurate calibration (largest R^2). The slower speed is also useful for rehabilitation. For this reason, all of Experiment 2's trials were run using a constant motor speed of 1575ms. The experimental set up is similar to Experiment 1 (i.e. replicate reset, activating motor while the user applies no force). To simulate the user applying a force by themselves, a non-conductive string was attached to the finger along the same line as the rubber cord in Fig. 20. The string was connected to a Mark-10 force gauge which the test subject pulled while the motor was activated [19]. A total of 30 runs were conducted. The same Arduino code as Experiment 1 (Appendix Table B1) is used to measure force at constant distance intervals (1.96mm = 20Hz) and calculate total work done by the motor. In addition, the force gauge measures force applied by the user.

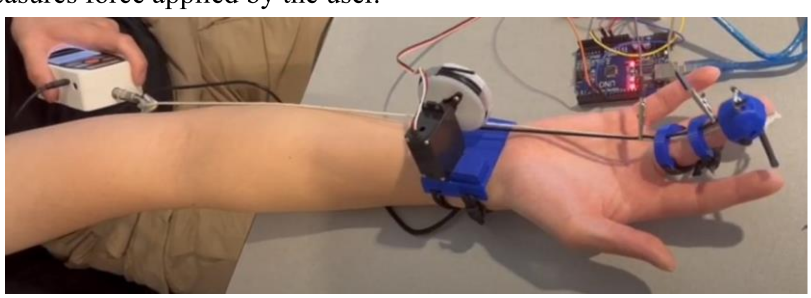


Fig. 20 Force is applied using a non-conductive string and force gauge that pulls the finger in parallel with the rubber cord/motor.

Experiment 2 consisted of multiple noise variables. Due to limitations of the Arduino, as well as the chance of the conductive cord getting caught throughout the trial, the motor may not move at a constant speed. This prevents force from being sampled at constant distance intervals to accurately calculate total work. Since we do not have access to a motor that can precisely measure distance, this noise variable was ignored. As well, the force gauge that we had access to had a low resolution of 0.5N and a sampling rate of only 10Hz. This decreases the resolution of the manual force measured, which can affect the ability to quantify a relationship between manual force and difference in the area under the force-distance curves. Lastly, due to the low sampling rate of the force gauge and how it does not measure force at constant distances, it was only feasible to take an average of the force applied by the user. In general, a larger average force applied using the force gauge should cause a larger decrease in the area under the force-distance curve. However, an average force does not account for the position in the finger curl where it is applied, which may change the effect on work. Using the force gauge to precisely measure forces applied over the finger curling distance may help reduce noise.

After the 30 trials were completed, a regression would be performed to establish a relationship between manual force and the change in motor work/area under the force-distance curve. The force gauge data was collected using the MESUR® Lite software [20]. As with Experiment 1, rubber cord force data was copied from the Arduino serial plotter and processed using python code in Appendix Table B2. A limit of detection will also be conducted to determine what force applied by the patient can be differentiated from when the patient does no work.

Results

A. *Experiment 1: Force-Distance Calibrations for Motor Speeds*

To ensure creep did not occur, in Fig. 21, the relative run order versus the total work measured was plotted for each speed to ensure permanent deformation did not occur. No trends occurred which means that the results are not confounded with the order.

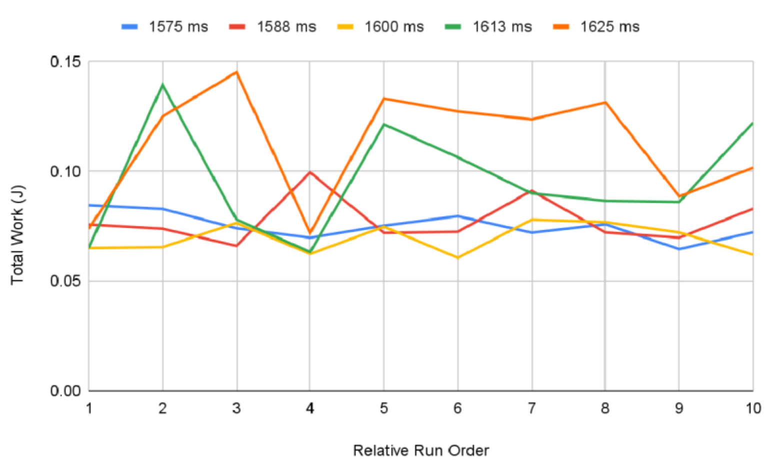


Fig. 21 Relative run order for each speed against total work done by the motor calculated by the device.
Table 3. Summary of total work measured by the device across speed treatments.

Speed (ms)	Mean (μ) Total Work (J)	Standard Deviation (σ) Total Work (J)	Achieved Metric 1 (σ < 0.1μ)?
1575	0.0751	0.00604	Yes
1588	0.0776	0.0105	Yes
1600	0.0694	0.00693	Yes
1613	0.0959	0.0256	Yes
1625	0.112	0.0262	No

Total work calculated by the device is described in Fig. 22 and Table 3. As speed increases, forces and total work tend to increase with speed.

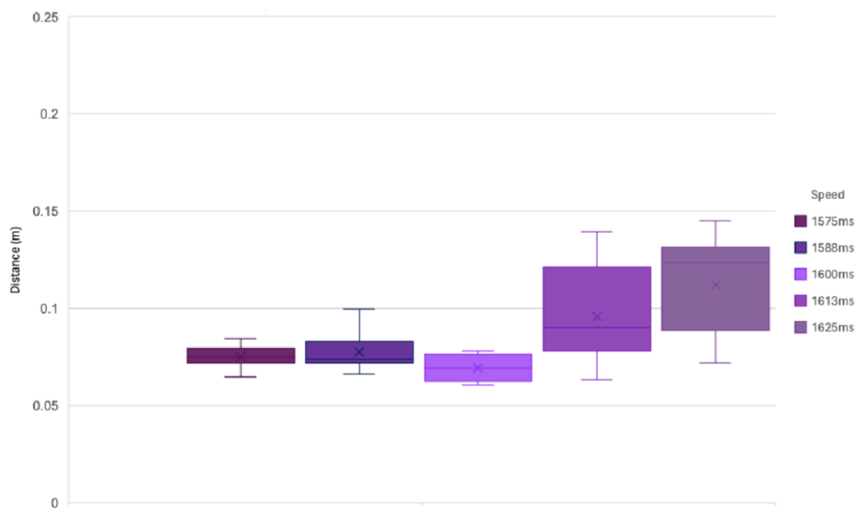


Fig. 22 Total work done by the motor when the user applies no force, calculated using the device.

Table 4. Summary of Single Factor ANOVA on total work for 5 speed treatments (α=0.05).

Source of Variation	SS	df	MS	F	P-value	F crit
Between Groups	0.0125	4	0.00312	10.2	6.04E-06	2.58
Within Groups	0.0138	45	0.000307			
Total	0.0263	49				

A single factor ANOVA in Table 4 with speed as treatment was conducted. There are k=5 treatments (1575, 1588, 1600, 1613, 1625 ms) and n=10 observations per treatment. The analysis found that the total work across different speeds are significantly different, with a small p value of (p=6.04E-06). This implies that speed does affect the total work which means that separate force distance calibration curves are required for different speeds.

Table 5. Summary of LSD on total work for 5 speed treatments (α=0.05).

Speed (ms)	Mean Diff Total Work (J)	LSD t-statistic	Are they different?
1575 vs 1588	0.0248	0.0250	No
1575 vs 1600	0.0572		Yes
1575 vs 1613	0.2074		Yes
1575 vs 1625	0.3706		Yes
1588 vs 1600	0.0820		Yes
1588 vs 1613	0.1826		Yes
1588 vs 1625	0.4278		Yes
1600 vs 1613	0.2646		Yes
1600 vs 1625	0.4278		Yes
1613 vs 1625	0.1632		Yes

A least significant difference (LSD) test was performed to determine which speeds produced different AOC/total work, as summarized in Table 5. The LSD found that only 1575 and 1588ms speeds have similar total work/AOC of the force distance graphs. The force-distance calibration curves were generated using a cubic polynomial in Fig. 23. All curves had an $R^2 > 0.7$, with the slowest speed (1575ms) having the largest R^2=0.95. As speed increases, the AOC visibly increases, indicating larger work performed by the motor.

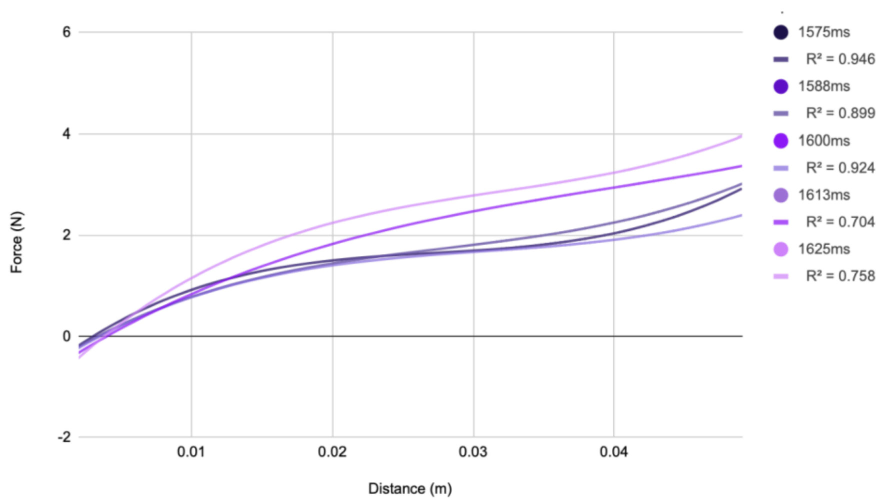


Fig. 23 Force vs Distance when the user applies no force, assuming a constant distance.

B. Experiment 2: Patient work to motor work relationship

In Fig. 24, a regression analysis for the 30 runs was conducted to determine if a positive linear correlation existed between the manual work done by the test subject and the difference in motor work from the calibration. The metrics are as follows: R^2=0.048 and ρ=0.30.

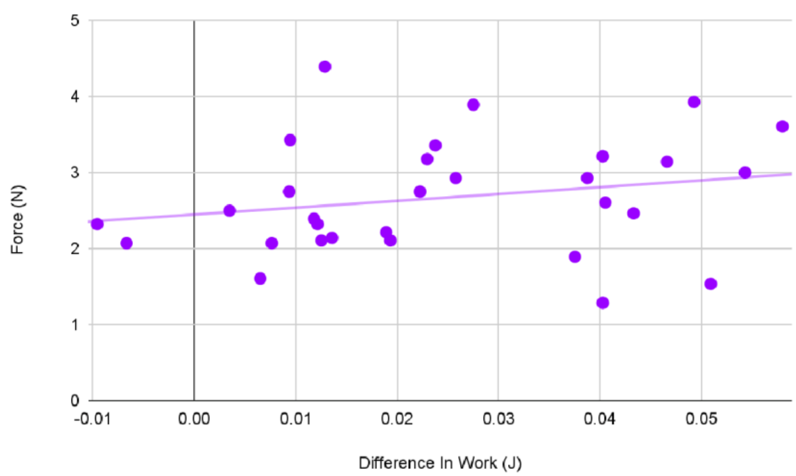


Fig. 24 Regression analysis of Experiment 2 to calibrate change in area (J) under the force distance curve (compared to Experiment 1's calibration of 1575 ms) against the average force (N) applied by the test subject using the force gauge.

The regression determined there was no correlation between the difference in work and the manual force exerted by the patient. A positive linear correlation was expected. The force-distance curves for each of the 30 trials are plotted in Fig. 25.

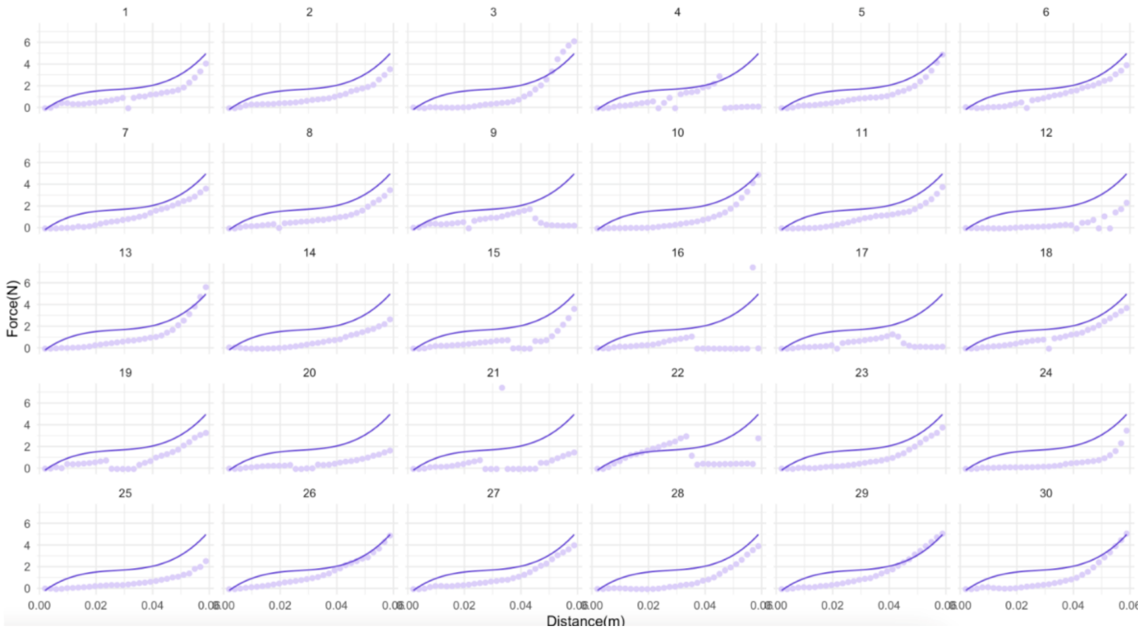


Fig. 25 Force Distance curves of 30 trials from Experiment 2.

A limit of detection was performed to test the model's accuracy in detecting the presence of patient work. The limit of detection for different standard deviations is shown in Fig. 26.

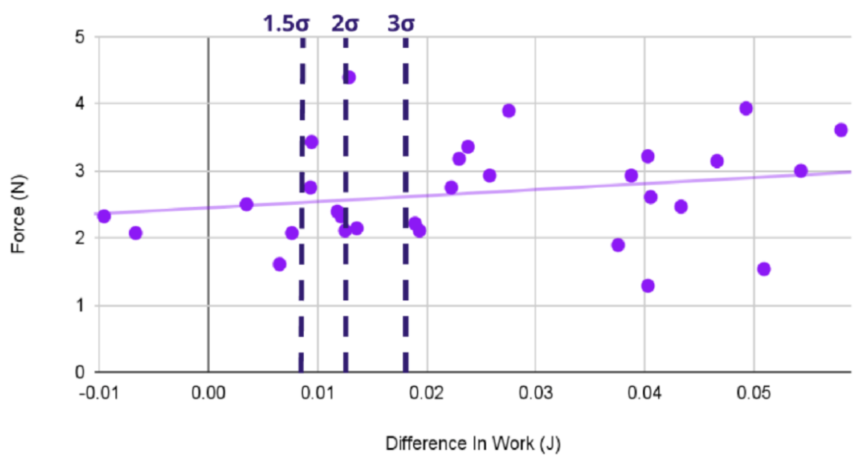


Fig. 26 Limit of detection at different standard deviations.

A receiver operator characteristic (ROC) curve was plotted below. It should be noted that any work difference reading that is negative will always be rejected, therefore, the maximum false positive rate for the ROC curve would be 50%. This was accounted for in Fig. 27, where the ROC curve only reaches a maximum of 0.5 false positive rate.

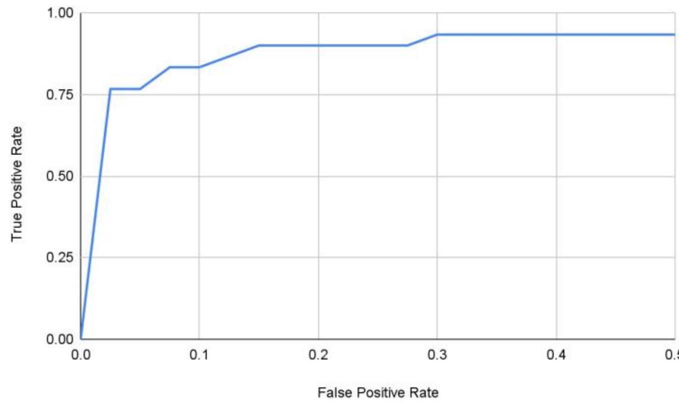


Fig. 27 ROC curve for Experiment 2 data.

ANALYSIS

A. *Experiment 1: Force-Distance Calibrations for Motor Speeds*

As speed increases, total work increases, which can be attributed to larger torques produced by the motor to achieve faster speeds. The motor may also accelerate faster to achieve the desired speed which results in larger forces. The first metric for Experiment 1 was that the device should consistently output force-distance curves with a consistent AUC (representing total work). All speeds achieved this,. except 1625ms which means that for most speeds, our force-distance calibration was consistent. The highest speed (1625ms) is the least consistent and had the largest standard deviation in total work/AUC potentially due to limitations of the servo motor's power supply. The LSD analysis found that 1575 and 1588 ms had similar total works. However, since most speeds had different total works, we still made separate force-distance calibrations for each speed. This will provide more accurate curves for the therapist to use a benchmark for the work done by the motor when the patient has no strength.

B. *Experiment 2: Patient work to motor work relationship*

Since no correlation was observed between the difference in motor work and the manual force applied by the force gauge, analysis of our trials was conducted to determine the cause. It was observed using videos of the trials that towards the end of the finger curl, the motor was pulling the finger at the third attachment point without contributing to the curling motion. This means that the motor is applying force that is being counted towards the total motor work being measured; however, it is not indicative of the work the motor is doing to assist the patient's motion. Looking at the trials using Fig. 25 shows evidence of this observation. For all but trial 3 of the 30 trials, there is a significant reduction in motor force at the beginning of the finger curl movement. However, the force increases significantly towards the end. Fig. 28 shows an example of this effect in trial 3. There is a reduction in work at the start, indicating that the user is doing work and the motor is providing less assistance. At the end, the string gets caught on the third attachment point, and the motor begins pulling the finger in without curling it. However, without proper sensors or a motor that can more accurately give metrics regarding distance, this data cannot be truncated. Therefore, the prototype in its current state cannot consistently quantify the patient's work using differences in the area under the force-distance curve.

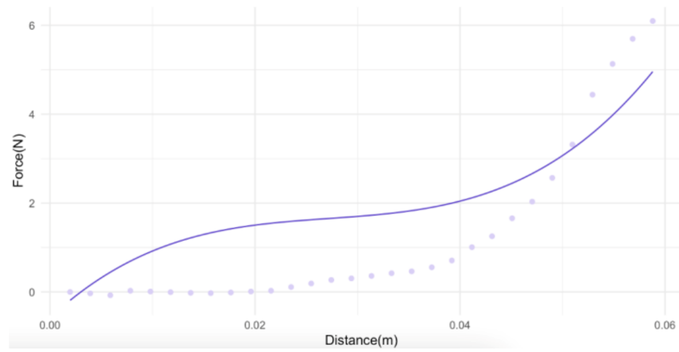


Fig. 28 Measured force-distance for trial 3 of Experiment 2.

A limit of detection was then performed for three standard deviation values in Fig. 26. For 1.5 standard deviations, it was found that a mere 6.5% false positive chance yielded a true positive rate of 83.3%. This indicates that the prototype is suitable to detect the presence of manual force. The ROC curve in Fig. 27 compares the false positive rate to the true positive rate, where a false positive would be a detection of manual force when none was applied. To create the ROC curve, a function was developed that could output a z-score threshold corresponding to a given false positive rate. As mentioned previously, only work readings lower than the calibration work result in a false positive. This means that the z-score threshold must satisfy the condition that the amount of data to its left is equal to the false positive rate. Once the z-score was calculated for a false positive rate, it was multiplied by the standard deviation for the calibration work to establish a threshold of detection associated with that chance of false positives. The relative frequency of data points that exceeded this threshold was calculated, and used as the true positive rate. The curve achieves close to a perfect classifier (area under the ROC curve equal to 1), indicating that the prototype is sensitive enough to determine the presence of patient work. It should be noted that the true positive rate could never reach 100 since multiple data points yielded a negative difference from the calibration work.

Recommendations and Conclusion

Experiment 1 achieved the metrics defined in requirements and produced consistent force-distance calibration curves at different motor speeds. Therefore, this calibration curve can be used as a benchmark for the therapist to compare work done by the motor when the patient applies force. It is important to note that the calibrations do not represent 'true' work since the experiment assumes that the motor is moving at a constant speed in order to sample at constant distance intervals. Small inconsistencies in the servo's travel distance were observed. In the future, a stepper motor could be used to improve the consistency of the finger curl distance and precisely measure in degrees distance travelled [21]. Additionally, since the viscoelastic rubber cord's properties vary with speed, differences in manufacturing should be considered to determine if the force-distance curves must be recalibrated. Blocking experiments can be used to compare different rubber cord batches. The maximum speed of the servo tested (1638 ms) resulted in inconsistent distances due to insufficient power. However lower speeds, around 1510 ms, were not adequate to move the finger. This draws the conclusion that only the application of speeds within this range will provide reliable data.

As seen in Experiment 2, the prototype cannot accurately measure the work done by the patient based on the difference in area under the force-distance curve. However, the limit of detection is relatively small. The area under the ROC curve was close to 1, which indicates that the device can be used to accurately determine the presence of work done by the patient; however, the regression shows that it is not sensitive enough to quantify the amount of work. In future experiments, a force sensor with higher resolution and sampling rate than the Mark-10 could be used. Also, using sensors to detect when the curling motion is complete will ensure only useful work done by the motor to assist the user is recorded. For example, a pressure sensor between the finger and palm could be used to determine when the curling motion is complete. The experiment had the test subject manually apply the force with the force gauge. In the future, an automated force can be applied to produce consistent force applications. Furthermore, hemiparetic patients experience severe weakness and cannot apply large amounts of force. For more realistic experimental conditions, the force gauge should apply smaller forces between 0 N and 1 N.

In conclusion, the prototype can successfully assist hemiparetic stroke patients with practicing the motion of curling their fingers into their palm, while providing feedback to the therapist about finger strength. In Experiment 1, the behaviour of the cord was measured for different speeds so that the therapist can quantify work done by the motor when the patient has no strength. Analyzing different speeds also ensures the device can operate at a speed that is comfortable for the user. However, based on Experiment 2, the sensitivity of the device to changes in patient work should be improved to provide detailed feedback to the therapist about work done by the motor versus the patient. This will increase the effectiveness and useability of the device.

References

[1] J. Dawson, A. H. Abdul-Rahim, and T. J. Kimberley, “Neurostimulation for treatment of post-stroke impairments,” Nature News, https://www.nature.com/articles/s41582-024-00953-z (accessed Apr. 9, 2024).

[2] J. S. Lora-Millan, F. J. Sanchez-Cuesta, J. P. Romero, J. C. Moreno, and E. Rocon, “Robotic exoskeleton embodiment in post-stroke hemiparetic patients: An experimental study about the integration of the assistance provided by the reflex knee exoskeleton,” Nature News, https://www.nature.com/articles/s41598-023-50387-8#:~:text=Hemiparetic%20gait%20is%20charact erized%20by,%2C%20and%2For%20ataxia5. (accessed Apr. 9, 2024).

[3] S. Frenkel-Toledo et al., “Shared and distinct voxel-based lesion-symptom mappings for spasticity and impaired movement in the Hemiparetic Upper Limb,” Nature News, https://www.nature.com/articles/s41598-022-14359-8#:~:text=By%20%27hemiparesis%27%20we%20refer%20in,the%20side%20of%20the%20stroke(accessed Mar. 10, 2024).

[4] P. Raghavan, “Upper Limb Motor Impairment after stroke,” Physical medicine and rehabilitation clinics of North America, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4844548/ (accessed Feb. 10, 2024).

[5] C. Liu et al., “Predicting hand function in older adults: Evaluations of grip strength, arm curl strength, and manual dexterity - aging clinical and experimental research,” SpringerLink, https://link.springer.com/article/10.1007/s40520-016-0628-0#citeas (accessed Apr. 12, 2024).

[6] X. Shang, X. Meng, X. Xiao, Z. Xie, and X. Yuan, “Grip training improves handgrip strength, cognition, and brain white matter in minor acute ischemic stroke patients,” Clinical Neurology and

Neurosurgery, https://www.sciencedirect.com/science/article/abs/pii/S0303846721004157#:~:text=Grip%20trainin g %20in%20the%20early,grip%20training%20effect%20on%20cognition. (accessed Apr. 12, 2024).

[7] C. E. Lang, S. L. DeJong, and J. A. Beebe, “Recovery of thumb and finger extension and its relation to grasp performance after stroke,” Journal of neurophysiology, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2712280/ (accessed Feb. 10, 2024).

[8] M. Kurosaki et al., “Functional Recovery after Rehabilitation in Patients with Post-stroke Severe Hemiplegia,” Progress in Rehabilitation Medicine, vol. 7, no. 0, p. n/a, Jan. 2022, doi: 10.2490/prm.20220039. (Accessed Apr. 9, 2024)

[9] J. Veldema and P. Jansen, “Resistance training in stroke rehabilitation: systematic review and meta-analysis,” Clinical Rehabilitation, vol. 34, no. 9, pp. 1173–1197, Jun. 2020, doi: 10.1177/0269215520932964. (Accessed Apr. 9, 2024)

[10] R. Hunter (2024). BME 361 Calibration of Conductive Rubber (Accessed Apr. 9, 2024)

[11] C. E. Lang, S. L. DeJong, and J. A. Beebe, “Recovery of thumb and finger extension and its relation to grasp performance after stroke,” Journal of neurophysiology, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2712280/ (accessed Apr. 9, 2024).

[12] J.-M. Dávila-Vilchis, J. C. Ávila-Vilchis, A. H. Vilchis-González, LAZ-Avil&#x00E9, and U. S, “Design Criteria of Soft Exogloves for Hand Rehabilitation-Assistance Tasks,” Applied Bionics and Biomechanics, vol. 2020, p. e2724783, Aug. 2020, doi: 10.1155/2020/2724783. (Accessed Apr. 9, 2024)

[13] R. Hunter (2024). BME 361 Motor Specs. (Accessed Apr. 9, 2024)

[14] C. Irvine, D. Cho, I. Yu, S. Thompson, V. Mac, “Case Study 2: Calibration of Conductive Rubber.” (accessed Apr. 12, 2024).

[15] C. Irvine, D. Cho, I. Yu, S. Thompson, V. Mac, “Case Study 3: Kinematic and Finite Element Analysis of Finger Model.” (accessed Apr. 12, 2024).

[16] R. Hunter (2024). Motor_and_Rubber_Code_v2.2.ino. (Accessed Apr. 9, 2024)

[17] “Digital Pins,” docs.arduino.cc, https://docs.arduino.cc/learn/microcontrollers/digital-pins/ (accessed Apr. 12, 2024).

[18] R. Hunter (2024).motorCalibration.ino. (Accessed Apr. 9, 2024)

[19] “Force Gauge - Series 3 - Mark-10,” Mark-10 Force and Torque Measurement. Accessed: Mar. 26, 2024. [Online]. Available: https://mark-10.com/products/force-gauges/series-3/ (Accessed: 12 April 2024).

[20] Basic Data Acquisition Software - MESURLite - mark-10 (2023) Mark. Available at: https://mark-10.com/products/software/mesur-lite/ (Accessed: 12 April 2024).

[21] "Basics of Stepper Motor," Oriental Motor U.S.A. Corp., https://www.orientalmotor.com/stepper-motors/technology/stepper-motor-basics.html#:~:text=Steppe r%20motors%20can%20be%20precisely,time%20they%20run%20open%20loop. (Accessed Apr. 9, 2024)

.

APPENDIX

Table B1. Arduino code to control the servo motor and measure force-distance curves. Built upon [16].

```
#include <Servo.h>
Servo servo1;

int RubberPin = A0; // input pin that reads the voltage drop across the rubber
wire, can be any of the analog pins A0-A5 int servoPin = 5; int buttonPin = 7;

// Note: 1.3ms is full speed clockwise. 1.7ms is full speed counter clockwise

float R_voltage;    // measured voltage drop value of the rubber wire float
R_resistance; // calculated resistance value of the rubber wire float
force; // calculated force value at the current sampling time float
motorInput;  // calculated input to the motor from a given speed float
distanceTravelled = 0; bool run = false;

int i=0;              // counter for force and data arrays

// Important values float motorRadius = 0.02; // motor
radius in m float angularVelocityMax = 5.236; // rad/s
float maximumMotorSpeed = angularVelocityMax *
motorRadius; float minimumMotorSpeed = -maximumMotorSpeed;
float totalWork = 0;


// SET INPUTS float speed = 0.03927;    // set motor speed in m/s float distance =
0.05;  //set distance in m (note: servo is kind of broken so need to adjust this
based on speed) const int Resistor = 1000;    // this is the fixed value of the
other resistor in the voltage divider circuit const float sample_interval = 50; //
interval of time between resistance (and force) measurements (in milliseconds)
const float sample_interval_seconds = sample_interval / 1000; // sample interval
in seconds

// SET MULTIPLIERS float distance_multiplier; // distance
travelled per time interval

void setup() {
 // startup the serial monitor for data to be displayed
```

```
 Serial.begin(9600); // initialize serial communication, 9600 is the baud rate
 Serial.print('\n');
 Serial.println("Force, Distance, Resistance, Work"); // starts the program on a
new line in the serial monitor

 // Handle the speed related calculations for the motor right away, as they are
constants
 motorInput = handleMotorInput(speed);

 // Second Check for ensuring motor input is not too high or low as radius was not
measured exactly
 if (motorInput > 1700) motorInput = 1700;
 if (motorInput < 1300) motorInput = 1300;

 // Attach servo and button
 servo1.attach(servoPin);
 pinMode(7, INPUT_PULLUP);

 // Calculate the distance travelled per sample time
 distance_multiplier = speed * sample_interval_seconds;
}

void calculateForce(float resistance)
{
 // Use the regression model from CS2 to determine the force force = (4.02 *
 0.00001 * resistance * resistance) - (0.0347 * resistance) + 7.41;
}

float handleMotorInput(float inputSpeed)
{
 // Ensure speed is not too great or low
 if(inputSpeed > maximumMotorSpeed)
 { speed =
   maximumMotorSpeed;
 }

 if(inputSpeed < minimumMotorSpeed)
 { speed = minimumMotorSpeed; // Note: minimum speed would be most negative,
   not
zero
 }

 // Calculate the pulse length in ms for the desired speed
 float angularVelocity = speed / motorRadius;
```

```
 // Note: motor maximum angular velocity is 5.236 rad / s. Min speed is 1300, max
speed is 1700. Separated on each sideof 1500 (0 rad/s) by 200

 float angularVelocityPercent = angularVelocity / 5.236;
 float motorInput = 1500 + angularVelocityPercent * 200; // pulse width travelling
backwards spinning wise

 return motorInput;
}

void loop()
{
 // button logic int input =
 digitalRead(buttonPin); if(input
 == LOW)
 { run =
   true;
 }

 // ISSUE: MOTOR MOVES A LOT MORE FOR LOWER SPEEDS. Distance it thinks it travels
 while(run)
 {
   // Ensure we can continue to move the motor without crossing the
maximum distance we set as an input if(distanceTravelled > distance -
distance_multiplier)
   {
     // Detatch the motor
     servo1.detach();
     Serial.println();
     Serial.println("distance, work, microsecond, speed");
     Serial.print(distanceTravelled, 4);
     Serial.print(",");
     Serial.print(totalWork, 4);
     Serial.print(",");
     Serial.print(motorInput);
     Serial.print(",");
     Serial.println(speed, 4);

     run = false;
     break;
   }

   // Actuate the motor servo1.writeMicroseconds(motorInput); // Note: get this
   to move for exactly the
sampling time and code should work
```

```
   // Calculate the force value at each sampling time
R_voltage = analogRead(RubberPin) * (5.0 / 1023.0);  // analogRead() reads the
voltage and returns a number betweenn 0 and 1023 that is proportional to the
voltage. Therefore, to reverse this and get voltage readings, we must divide by
1023 and multiply by the input voltage of the circuit which is 5V
   R_resistance = (5/R_voltage - 1)*Resistor; // voltage divider equation re-
arranged to find R2.

   calculateForce(R_resistance); float currentWork
   = force * distance_multiplier; float time =
   sample_interval * i;

   // Print information
   Serial.print(force, 5);
   Serial.print(",");
   Serial.print(distance_multiplier, 5);
   Serial.print(",");
   Serial.print(R_resistance, 5);
   Serial.print(",");
   Serial.println(currentWork, 5);

   totalWork += currentWork;

   i++; distanceTravelled += distance_multiplier; // iterate by distance
   travelled per
sample interval to get the distance travelled so far delay(sample_interval);
   // the loop is paused for the sample interval time
 }
 }
```

Table B2. Python script to process Arduino serial monitor raw output into csv files to perform analysis in experiment 1 and 2.

```
# Arduino data over distance -- Force, Distance, Resistance,
Work import pandas as pd import csv

df = pd.read_csv('experiment_2_data.txt',
delimiter=",") trial = 1 value = [1] skip_next = False

for ind in df.index:
   if skip_next:
   skip_next = False
   continue

   if (df[' Work'][ind] == ' Work'):
     with open("work.csv", "a") as csv_file:
       writer = csv.writer(csv_file,delimiter=',')

       print(value)
       writer.writerow(value)

       trial = trial + 1
       value = [trial]
     continue
   if (df[' Work'][ind] == ' speed' or df[' Work'][ind] == ''):
     skip_next = True continue

   value.append(df['

   Work'][ind])

with open("work.csv", "a") as csv_file:
     writer =
     csv.writer(csv_file,delimiter=',')
     print(value) writer.writerow(value)

# Arduino total/average data - distance, work, microsecond,
speed import pandas as pd import csv

df = pd.read_csv('experiment_2_data.txt',
delimiter=",") trial = 1 value = [1] skip_next = True

with open("arduino_avg_work.csv", "a") as csv_file:
       row = ['distance', 'work', 'microsecond', 'speed']
       writer = csv.writer(csv_file,delimiter=',')
       writer.writerow(row)

for ind in df.index:
```

```
   if not skip_next:
     skip_next = True

     with open("arduino_avg_work.csv", "a") as csv_file:
       value.append(df['Force'][ind])
       value.append(df[' Distance'][ind])
       value.append(df[' Resistance'][ind])
       value.append(df[' Work'][ind]) writer =
       csv.writer(csv_file, delimiter=',')
       print(value) writer.writerow(value)

       trial = trial + 1 value = [trial]
  continue

if (df[' Distance'][ind] == ' work'):
  skip_next = False continue

if (df[' Distance'][ind] == ' distance' or df[' Distance'][ind] == ''):
       skip_next = False continue
```