DESIGN AND DEVELOPMENT OF A HAND EXOSKELETON

A Final Year Project Report

Presented to

SCHOOL OF MECHANICAL & MANUFACTURING ENGINEERING

Department of Mechanical Engineering

NUST

ISLAMABAD, PAKISTAN

In Partial Fulfillment
of the Requirements for the Degree of
Bachelors of Mechanical Engineering

by

Mohammad Talha Bin Noman
Bilal Ahmed Kamran
Musawwir Ahmad

June 2023

EXAMINATION COMMITTEE

We hereby recommend that the final year project report prepared under our supervision by:

Mohammad Talha Bin Noman	297145.
Bilal Ahmed Kamran	289726.
Musawwir Ahmad	302604.

Titled: "DESIGN AND DEVELOPMENT OF A HAND EXOSKELETON" be accepted in partial fulfilment of the requirements for the award of BACHEOR OF MECHANICAL ENGINEERING degree with grade ___

Supervisor: Dr. Yasar Ayaz, Professor School of Mechanical & Manufacturing Engineering (SMME) and Chairman NCAI	_____ Dated:
Committee Member: Dr. Khwaja Fahad Iqbal, Assistant Professor School of Mechanical & Manufacturing Engineering (SMME)	_____ Dated:
Committee Member: Dr. Sara Babar, Assistant Professor School of Mechanical & Manufacturing Engineering (SMME)	_____ Dated:

_____ _____

(Head of Department) (Date)

COUNTERSIGNED

Dated: _____ _____

(Dean / Principal)

ABSTRACT

Paralysis is a common affliction in the world today and has been for ages. The historical bias against it can be judged from the famous case of the German Emperor Wilhelm III. New terminologies such as differently-abled have begun to be used to reduce the stigma around to encourage acceptance of it. As much as it is difficult for the person to live through the ordeal, it is harder still for society to handle paralyzed patients with care; hence it is always a point of interest for all those involved with someone with paralysis, for the patient to regain some form of autonomy to assist them in their activities of daily living. A prolonged period of paralysis spawns other challenges, such that prolonged disuse of muscles can cause atrophy. To stop paralysis in its track or better still recover function, is therefore highly sought. Stroke patients are usually undergoing physiotherapy and medication to help through the process. Physiotherapy involves rehabilitative exercises aimed at restoring motor function. The aforementioned rehabilitative exercises often involve prescribed movements and are limited to specialized training centers where specific training equipment is made use of use of. The process of completely regaining motor function while quite desirable soon becomes a boon, and its repetitive nature and the confinement that comes with it being carried out in small spaces can render it frustrating.

It has been observed that almost sixty percent of all patients suffering from paralysis, present with upper extremity dysfunction [1]. This is acutely traumatizing to the patient because it nibbles away at the most recognizable sign of autonomy – hands, and as such renders the patient too dependent. Any sort of recovery with regards to hands is highly welcome because it acts as a step towards reclaiming autonomy for the patient and improves their mental state. Hand exoskeletons are a recent innovation and have been adopted to assist in the process of rehabilitation. Hand exoskeletons can be considered a piece of portable equipment and thus the patient can easily engage in ‘rehabilitation’

without the need of being in a specialized training center which the patient might discompose patients owing to their unpleasant associations with it.

The present thesis presents the design and development of a hand exoskeleton. While the primary aim is to assist in the process of rehabilitation for patients suffering from paralysis. It does not preclude the possibility of its being used in other areas where an application may be found to exist. Rock climbing or other sports that may require a large amount of force from the hands, cannot do away with the possibility of a device that may assist in force augmentation. This feature of force augmentation may also render it very useful in military exercises. The force of recoil from a rifle or any such personnel may be easily countered by way of force augmentation achieved through a hand exoskeleton. This military application just cited can be appreciated, when you realize that the force from recoil can in the worst-case scenario even cause a fracture in the clavicle – or 'beauty bone'.

The design presented in the present thesis proposes a unique solution that caters to the design of the linkages. The dimensions of the links in a hand exoskeleton, if not selected properly can make for unnatural trajectories of the hand, which may introduce undesirable stresses in the hand and at any point, thus rendering the process of rehabilitation more painful. The design in this paper makes use of differential evolution to overcome this problem. With the natural trajectory of the finger identified, the process of finding the right dimensions of the link to make the exoskeleton more user-friendly becomes easy. With the right trajectory identified through software like Kinovea, and feeding the data in a code, the algorithm gives the dimensions most likely to efficiently approximate the given trajectory.

With the development of such a hand exoskeleton, that is portable and user-friendly, its deployment could work wonders. Exoskeletons in existence usually make use of either soft or hard robotics. Soft robotics involves shape-memory alloys or Bowden cables that

are actuated with fluids. Hard robotics involves mechanisms that operate on linkages and transmit power mechanically rather than through fluids as is the case with soft robotics. The hand exoskeleton in the present thesis bases its design on hard robotics and optimizes the dimensions of the linkages to make it follow the natural motion of the finger. The control of hand exoskeletons is an ongoing area of research and as such there are a few options that exoskeletons operate on to achieve control. There is the approach of using electromyographic (EMG) signals which pick up signals from healthy nerves. One of the approaches involves sensors mounted on a healthy hand, to replicate its motion onto the other hand – this presupposes a healthy hand. FSR rehabilitation is another method and it involves the use of FSRs – force-sensitive resistors. FSR rehabilitation leads to better monitoring of the patient's progress which helps the concerned physiotherapists to modify the plan of exercises.

ACKNOWLEDGMENTS

This project was a year-long commitment, and it was only possible through the guidance and support we received from our mentors. We would like to extend our gratitude to our supervisor and mentor, Dr. Yasar Ayaz (HI), who gave us the mindset to question our decisions and appreciate ways in which this project could incorporate other fields and lend itself an ‘interdisciplinary’ nature and its potential for future commercialization.

We would like to thank Dr. Khwaja Fahad Iqbal for his insightful comments and constructive criticism. His guidance was instrumental in refining our research ideas and methodology. We would also like to thank Dr. Sara Babar for her insightful feedback. Her keen attention to detail allowed us to appreciate the merits of fine detail in professional research.

We would also like to thank Dr. Aamir Mubashir for his crucial insight into how to conduct a dynamic analysis of the design.

Working on the design and development of the hand exoskeleton was not an easy task; however, the constant support from our mentors and family, and friends helped us through each phase and allowed us to appreciate the impact of the work we were doing. This project left an indelible mark on us as mechanical engineers, realizing how our work had the potential to change lives. This noble pursuit in sight, made all the work we did more worthwhile.

ORIGINALITY REPORT

Thesis

ORIGINALITY REPORT

8%	5%	4%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dokumen.pub Internet Source	<1%
2	www.biodiversitylibrary.org Internet Source	<1%
3	Inseong Jo, Joonbum Bae. "Design and control of a wearable and force-controllable hand exoskeleton system", Mechatronics, 2017 Publication	<1%
4	1library.net Internet Source	<1%
5	vtechworks.lib.vt.edu Internet Source	<1%
6	Submitted to University of Paisley Student Paper	<1%
7	Submitted to Coventry University Student Paper	<1%
8	Ottó Botond Lőrinczi, Lívia Hanusovszky, Petra Aradi. "Evaluation of Three Different Phalangeal Motion Measurement Systems	<1%

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

1.1 Background

Humans are much too fascinated with mechanization. The popularity of the Ironman franchise, cyborgs, or the antagonist Dr. Octopus from the Spiderman franchise; points out the fascination humans have for mechanical suits that could augment their strengths. While the stuff of science fiction surely knows no bounds, it has been an active area of research since the latter part of the last century. The present thesis documents an attempt at designing and developing a hand exoskeleton. Hand exoskeletons can be used for rehabilitative purposes by patients suffering from paralysis that affects the upper extremities. Hand exoskeletons are capable of force augmentation and in such a capacity, may be utilized wherever the need may arise – such as military exercises involving rifles or any such weapons where the incidence of recoiling may be expected – in such a case force augmentation would counter the effect of recoil. Sports or any operations that may require a considerable amount of effort exerted by the hand, may safely be carried out by a hand exoskeleton.

1.2 Aims and Objectives

Strokes are a debilitating medical condition and are described by the World Health Organisation (WHO) as "rapidly developed clinical signs of focal (or global) disturbance of cerebral functions, lasting more than 24 hours or leading to death, with no apparent cause other than of vascular origin" [14]. Stroke patients are more likely to suffer from any sort of incapacity. Around fifty million stroke patients worldwide suffer from some sort of physical or cognitive disability that impacts their emotional well-being and due to severely reduced mobility, there is much greater dependence on other individuals for carrying out activities of daily living (ADL) [15]. Around two million people suffer a stroke every year and this figure rises every year by 6.7% [16]. It has been estimated that around sixty percent of all stroke patients suffer from upper extremity dysfunction [1].

This device is primarily focused on stroke patients, to assist them in restoring the motor function of their hands to make them self-sufficient in carrying out ADL. Such a disability may arise from a spinal cord injury [17], stroke [18], or cerebral palsy [19].

1.2.1 Hand Module Objectives

The present modules have been modeled after RobHand [1].

- An underactuated, top-mounted, lightweight, linkage-based hand exoskeleton.
- The exoskeleton should ideally provide structural rigidity and be pliant to the natural trajectory of a human finger.
- The exoskeleton should be able to exert 50 N [1] with each of the fingers supporting 10 N.
- Finite Element Analysis of proposed design
- Development of prototype of proposed design
- Implementation of a control system
- Demonstration of application

1.2.2 Control Module Objective

- A control module independent of EMG sensors
- The system should be able to reckon the flexion of the finger, independent of each joint.
- Capable of transmitting signals to the hand module.

1.3 Project Deliverables

- A hand exoskeleton that can assist in ADL

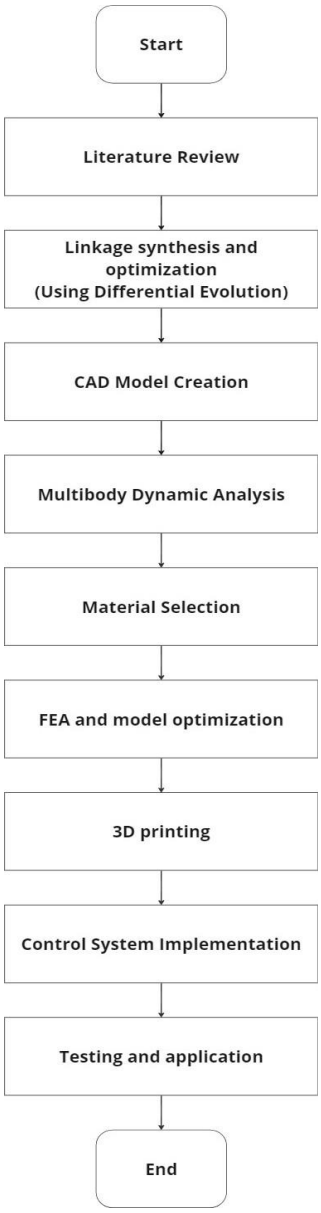
- A hand exoskeleton that incorporates ergonomic comfort, i.e., follows the natural path of the finger.

- A control module with sensors to detect movements from the healthy hand and map onto the hand, with the hand exoskeleton mounted on.

1.4 Research Methodology

Several designs of hand exoskeletons were reviewed from the available literature, and a few design parameters were identified. The literature review gave us a good idea of what project deliverables to set. The design deliverables were decided for the hand exoskeleton to be lightweight, top-mounted, and underactuated. The challenge of designing an exoskeleton that conforms to the natural path of the finger, without using a fully actuated system, was overcome by using an optimization algorithm. The optimisation algorithm used was differential evolution. The dimensions so obtained gave the first iteration of the design. Three iterations had to be run to reach the present state of the design. With the design phase complete, a dynamic analysis was run on the design through MSC Adams. The results from the multibody analysis yielded results that proved vital in material selection. With the material selected, Finite Element Analysis was run on the model and it was further optimised. This was followed by fabrication. The prototype was fabricated through 3D printing. A control system was implemented on the prototype to achieve ‘mapping movements of the healthy hand onto the paralysed hand’. The project was closed with the testing of the final prototype.

The overall methodology of the project is summarised in the flow chart on the next



1.5 Thesis Structure

1.5.1 Chapter 2: Literature Review

This section gives a basic summary of the findings. There is a classification involved of hand exoskeletons, and ways to achieve control of the mechanism. There is also a brief section that summarises the different diseases that cause paralysis.

1.5.2 Chapter 3: Methodology

This chapter shows in detail the work undertaken to synthesize a preliminary design for the hand exoskeleton.

1.5.3 Chapter 4: Results and Discussions

This chapter deals with the analysis of the hand exoskeleton and discusses the results of the analysis.

1.5.4 Chapter 5: Conclusions and Recommendations

This chapter summarises the whole process from the concept behind the idea to how it was eventually realized and where the team is headed in the future.

CHAPTER 2: LITERATURE REVIEW

2.1 Loss of Motor Function

Paralysis refers to the loss of muscle function and results from an impairment in the nervous connection that exists between the brain and the muscle. Paralysis can present itself either in the form of weakness or complete paralysis that renders a part of the body unresponsive and stiff to any internal or external stimuli. Hemiplegia refers to the complete paralysis of one side of the body and hemiparesis refers to weakness in one side of the body. The two halves of the brain control the opposite sides of the body so that the part of the brain that may be affected in the event of a stroke, renders the opposite side of the body weak or paralyzed. The three types of strokes are ischaemic, hemorrhagic, or transient ischaemic attacks. Ischaemia refers to severely reduced blood flow to the brain. Ischaemic strokes occur when a blood clot narrows or blocks an artery in the brain, reducing blood flow[42]. If a blood vessel leaks or ruptures, blood spills over into the brain tissue – this results in a hemorrhagic stroke and warrants immediate surgery [42].

Atherothromboembolism in the carotid artery is the chief cause of an ischaemic transient attack [43]. Time to intervention is crucial and the risk of stroke within ninety days of a transient ischaemic stroke is 2% among those treated within 72 hours of the stroke so time is key in its treatment [43].

Strokes affect about 800, 000 people each year in the United States alone. [42] While debilitating it may be for the patients and toll-taking for the individuals involved with the affected, the cost incurred on the economy is substantial too, with 68.9 billion dollars claimed by the direct or indirect consequences of the disease in 2009 alone [21]. It has been estimated that around four million people suffer from hemiparesis or a similar disability in the United States and six million in developing nations [37] [38]. The annual incidence of stroke in Pakistan is 250/100,000 [22]. Loss of motor function is most commonly attributable to neurological disorders such as stroke, myasthenia graves, or

motor neuron disease. In a study conducted by Parker et al, it was observed that 24% of patients were affected by moderate or severe paralysis after three months of suffering a stroke. Of the 24% severely affected, 17% experienced impairment in their dominant limb. Spontaneous recovery while not impossible, cannot be prevailed upon and rehabilitation efforts are increasingly strenuous and challenging. Rehabilitation does yield promising results. Statistics say that 40% of patients show improvement and a further 13% show positive signs of motor recovery [39]. Motor recovery is more involved in the upper extremity compared to the lower extremities and gradually decreases with time [18] so early intervention is more fruitful than late.

2.2 Rehabilitation for Motor Function Restoration

Three factors play a crucial role in the effectiveness of physiotherapy aimed at restoring motor function [39]. The earlier the intervention the better chances there are of motor recovery. The second factor that proves effective is Task-oriented learning. Task-oriented learning leans towards positive reinforcements, rendering each task fulfilling. While it has been observed that repetitive monotonous tasks can prove to be frustrating, undesirable, and test the patience of patients; they are nonetheless effective. In fact [39] states repetition intensity improves the chances. The converse may be argued by citing J H Carr and Shepherd [40], who argue for motor relearning techniques where the focus shifts from repetitive tasks to ones that the patients are more likely to encounter in their lives and directly involve ADL. They also focus on exercises that primarily target a certain area of motor skills – a given task requires the application of those target motor skills to accomplish. They also argue for the tasks to be carried out in the most appropriate environment so the sensory inputs can modulate to the environment.

2.2.1 Rehabilitation techniques

Rehabilitation techniques can be as varied as the motor skill they cater to – the one skill that is meant to be recovered. Just as workout routines focus on stimulating specific

muscles, similarly rehabilitative techniques involve a specific approach whereby each exercise aims at specific joints. Traditionally, most exercises rely upon external intervention; though with the introduction of FSR rehabilitation, this could change. Force-sensitive resistors (FSR) resist any motion taken by the patient and in doing so allow the patient to independently exercise. There are at least eight techniques in use. Motor rehabilitation using Virtual Reality (VR) [28], Brunnstrom movement therapy [29], Motor relearning program [30], constraint-induced movement therapy [31], robot-aided sensorimotor stimulation [32], augmented reality approach [33], and repetitive hand movement [34].

2.2.2 Rehabilitation stages and criteria

Identifying the stage of recovery in the process of rehabilitation is crucial in ensuring the effectiveness of the process. Fugl-Meyer Assessment is a quantitative assessment for stroke recovery based on Twitchell and Brunnstorm's concept of stages in the process of motor recovery [35] [36]. Fugl-Meyer assessments have been tried and tested. The most conclusive measure of the effectiveness of an assessment is the reproducibility of results – precision, and accuracy. The Fugl-Meyer assessment fulfills the criterion to an acceptable degree [37]

2.3 Robotic Rehabilitation

Robotic rehabilitation focuses on rehabilitation involving robotic techniques. One such technique employs hand exoskeletons for motor recovery of the upper extremities. Research in the field of robotics has been going on for the last two decades. A significant portion of research in the field of robotics for rehabilitation has been directed toward devices, One of the most well-known, if not the first, hand exoskeletons that had been produced was HEXORR. HEXORR stood to be a standard for subsequent hand exoskeletons to follow.

2.4 Types of Exoskeletons

2.4.1 Hard Exoskeleton Systems

2.4.1.1 RobHand

This exoskeleton [1] has the advantage of being underactuated – the number of actuators is less than the degrees of freedom for a finger. It is a hard exoskeleton. It can allow force augmentation of 10 N, even though it is only 750 g in weight. There are two vector loops involved in its mechanism.

Figure 1 RobHand

2.4.1.2 Shape Memory Alloy Exoskeleton

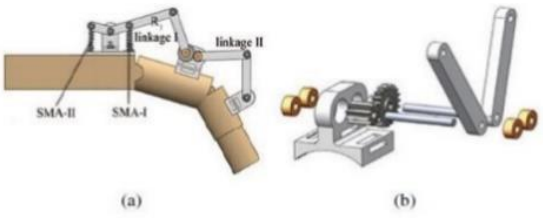


Figure 2 Shape Memory Alloy Exoskeleton

T. Tang et al [25] developed a hand exoskeleton system consisting of kinematic chains of four-bar linkages and shape memory alloys used as actuators. The device is top-mounted and uses gears as links between two four-bar linkages. It uses sensors in its feedback control system.

It has the advantage of being small, light, and has a comparatively simpler structure than RobHand. It uses geared linkages and what stands out in it, is that it has a low torque requirement, but it can cover the full range of motion (ROM).

2.4.1.3 Hand of Hope

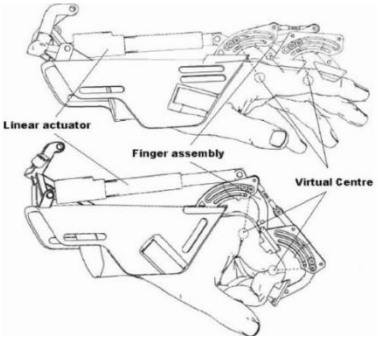


Figure 3 Hand of Hope

This is a lighter exoskeleton (500 g) compared to RobHand (750 g). It stands out as portable, self-contained, and EMG controlled.

2.4.1.4 HEXORR

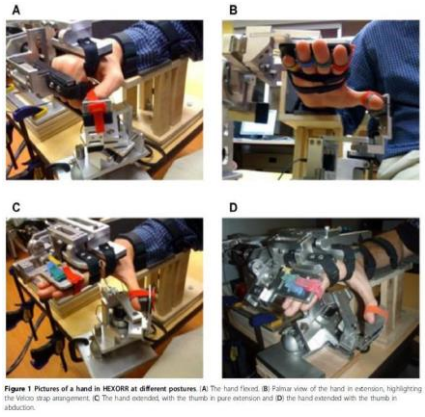


Figure 4 HEXORR

2.4.1.4.1 Main design parameters

HEXORR was one of the first exoskeletons. It has the advantage of back-drivability and gravity compensation. There is low friction between the gear trains and the electric motors and is driven by four-bar linkages.

2.4.1.5 Design presented by Iqbal et al

The design presented by Iqbal et al [23] presents a portable device. The device is underactuated, top-mounted, back-driveable, provides position feedback, consists of three links, and has a single point of attachment. Its focus is towards compatibility with forces that a human hand can exert or experience – so it hints towards a 'force compatible' hand exoskeleton.

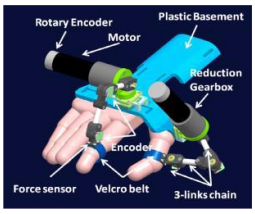


Figure 5 Iqbal et al 2010

2.4.1.6 Design presented by Jo et al

Jo et al [24] presented a force-controllable hand exoskeleton system. The device is underactuated and top-mounted and series actuators are employed in the design to provide force feedback.

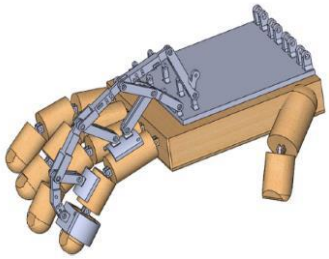


Figure 6 Jo et al

2.4.1.7 Design presented by W.A. Surendra et al

W.A. Surendra et al. [26] developed a rather unique hand exoskeleton, exhibiting a combination of tendons and linkages. Tendons actuate the mechanism and linkages are used for force transmission. It is top-mounted and underactuated.

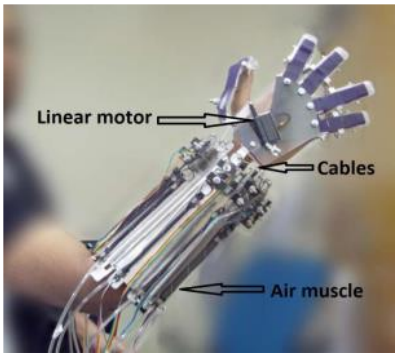


Figure 7 W. A. Surendra

2.4.1.8 Design presented by M. Transcossi et al

M. Transcossi et al [27] are unique in the way that it incorporates the wrist too. The exoskeleton extends over the wrist and the hand. The hand exoskeleton employs linkages, is top-mounted, and is underactuated up to two degrees of freedom. One actuator controls the motion of the thumb while the other controls the motion of the rest of the four fingers.

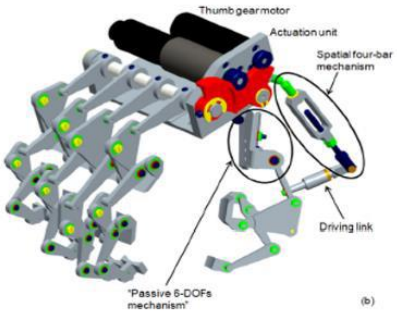


Figure 8 Image of design presented by M. Transcossi et al

2.4.1.9 Design presented by P. Ben-Tzvi et al

P. Ben-Tzvi et al [28] present an underactuated, top-mounted, sensing, and force feedback hand exoskeleton. It is an end-effector hand exoskeleton.

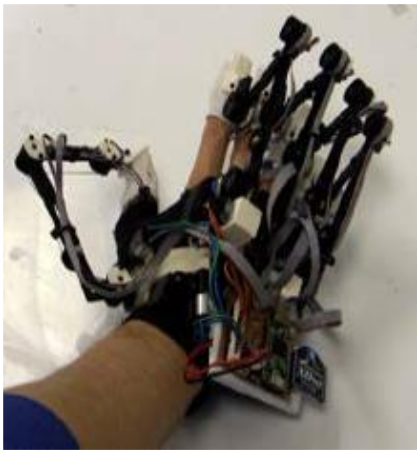


Figure 9 Prototype of the design presented by P. Ben Tzvi et al

2.4.1.10 Design presented by A. Wege et al

Figure 10 Prototype of the design presented by A. Wege et al

A.Wege et al [29] present a fully actuated and top-mounted hand exoskeleton. The system employs linkages, pulleys, and Bowden cables.

2.4.1.11 HANDEXOS

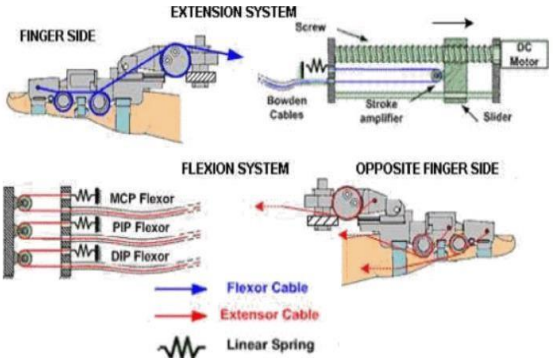


Figure 11 Design of HANDEXOS

HANDEXOS actuates using Bowden cables. It is also underactuated and has low inertia – easier to start a motion with and can be remotely actuated.

2.4.2 Soft Exoskeleton Systems

2.4.2.1 HWARD

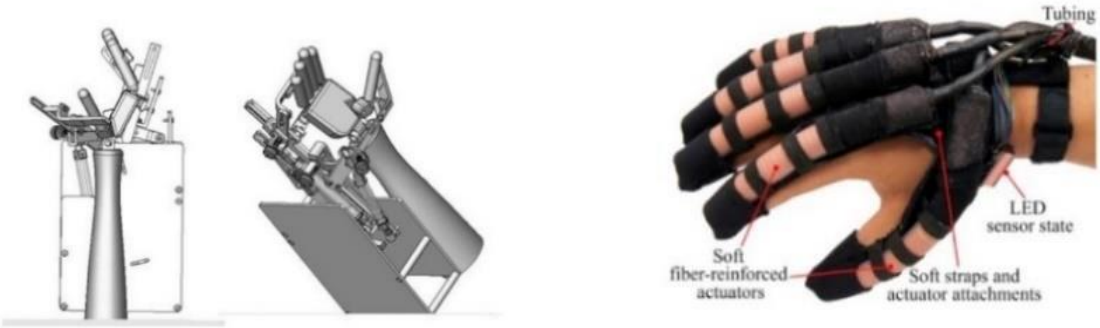


Figure 12 Design of HWARD

HWARD operates on soft robotics and demonstrates back-drivability like HEXORR and also takes advantage of a levered mechanism. The glove makes use of soft elastomeric

chambers reinforced with fiber reinforcements to produce bending and twisting under actuation.

2.4.2.2 Design presented by S.A. Fischer et al

The hand exoskeleton presented by S.A. Fischer et al, makes use of Bowden cables controlled through servomotors. This glove actuates over the full range of motion.

Figure 13 Prototype of the design by S. A. Fischer et al

2.4.2.3 Design presented by J. Arata. et al

J. Arata. et al [30] innovated a sliding spring mechanism in three layers to accommodate the bending of the finger. The system is underactuated.

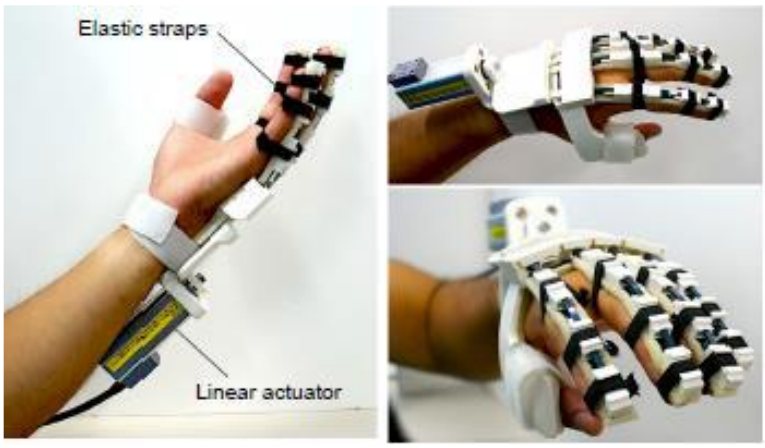


Figure 14 Prototype of the design by J. Arata et al

2.4.2.4 Design presented by Yang et al

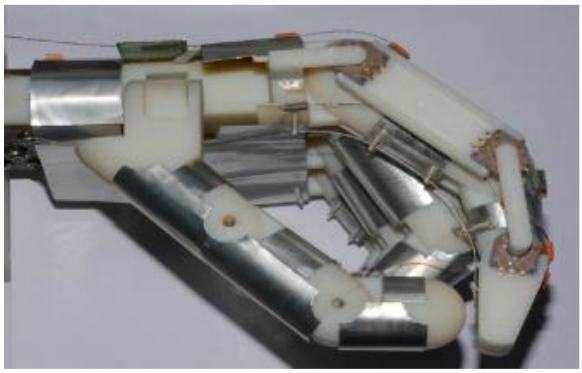


Figure 15 Prototype of the design by Yang et al

The hand exoskeleton proposed by Yang et al [32] was designed for patients in their later stages of recovery.

2.4.2.5 Design Presented by Y. Hasegawa et al

Y. Hasegawa et al [33] developed a tendon-driven top-mounted exoskeleton. It employs bioelectric signals for control – EEG or EMG.

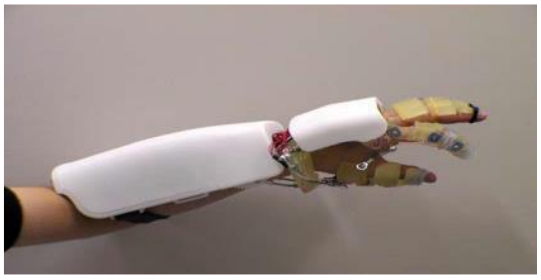


Figure 16 Prototype of the design by Y. Hasegawa et al

2.4.3 Actuation

Exoskeletons are differentiated into fully actuated or underactuated mechanisms for the control of the system. A mechanism is fully actuated when there is an actuator for each degree of freedom of the mechanism. A fully actuated mechanism would automatically

remove the issue inherent in underactuated mechanisms. A fully actuated hand exoskeleton would more or less allow the mechanism to follow the natural trajectory of the finger – a logarithmic curve, quite smoothly. However attractive such a system may sound; it fails on a few counts when its feasibility is judged under the microscope. A fully actuated mechanism with its many actuators is bound to be cumbersome and the idea of installing 24 actuators would certainly defenestrate any idea of the device being lightweight. In the case of using electroencephalographic or electromyographic signals to control the mechanism, it would be difficult achieving it because of the weak signal distributed among all the actuators.

An underactuated mechanism refers to a hand exoskeleton where the number of degrees of freedom outnumbers the actuators attached. The present thesis has focused on the development of a hand exoskeleton that uses a single actuator for each finger – thus five in total. Underactuated mechanisms thus ensure that the EEG or EMG signal is efficiently utilized. An underactuated mechanism would be lighter. The one-count where a fully actuated mechanism could be justified was the faithfulness in following the natural path of the finger; however, successful implementation of the differential evolution algorithm ensures that the underactuated mechanism solves the issue.

2.4.4 End effector and exoskeleton systems

End-effector actuation systems are specific types of underactuated mechanisms. There can be three joints at most in an exoskeleton modeled on a human finger. For end-effector exoskeleton systems, force is applied on the distal phalange instead of the proximal or the intermedia joints. A technique that ensures maximum propagation of force entails the application of force in an outward-bent fashion to force the finger to reach the furthest.

End-effectors are distinct from other exoskeletons in the way that other exoskeletons may mount their actuators on the joints that precede the distal joint.

2.5 Comparison

The rigid mechanics involved in hard exoskeletons ensures better force transmissibility and force augmentation. The rigid structure is easily controllable; thus, achieves reliable performance owing to its structural integrity, force transmissibility, and efficient control. Hard mechanisms while quite efficient can make for heavier devices. Hand exoskeletons like HEXORR solve the issue by providing gravity compensation and back-drivability. Gravity compensation is a great feature that allows the heavy hand to be lifted against the ground with little force needed by the user to lift – the rest provided by the system; thus it may aptly be said to have low inertia. Hard exoskeletons can be bulky whereas soft exoskeletons are more likely to be lightweight. One key area where soft exoskeletons are better is compliance and softness. This compliance and softness lend it an ergonomic advantage over hand exoskeletons. Hand exoskeletons can achieve ergonomic advantages but that often entails a fully actuated mechanism which makes it bulky and heavy. The rotational elements of soft exoskeletons are not rigid and as such, the finger of a soft exoskeleton pivots on the joints of the human hand.

2.6 Signal Extraction

2.6.1. Electroencephalography

Electroencephalography is a compound of three Greek works and translates to ‘electrical mapping of the brain’ – mapping of the electrical signals produced by the brain. Electroencephalography (abbreviated to EEG) measures variations in voltage across brain neurons, due to ionic current that may be attributed to action potentials. It is an electrophysical monitoring method that monitors electrical activity inside the brain. To ‘harness’ these signals for control in a hand exoskeleton, a Brain-Computer Interface (BCI) is developed. BCIs are remarkable in that they provide a direct line of communication between the wired brain and the computer. BCIs can be of a few types depending on where they pick signals from and how invasive they are.

Invasive Brain-Computer Interface implants are directly planted onto the brain's grey matter; it is for this reason that invasive BCIs are normally only used when there is a neurosurgery in progress. Since the brain's grey matter is directly taken signals from, with nothing impeding it; fully invasive BCIs produce the highest quality of EEG signals.

Partially invasive BCIs, while not as invasive as direct plantation onto the brain, are still considered invasive, with the BCI planted onto the skull. There may arise a discrepancy in the quality of the signal compared to the fully invasive BCI but it is nevertheless still considered invasive to ensure a high-quality signal.

Non-invasive BCIs are interfaces developed by plating electrodes on the outer surface of the skull in a non-invasive manner. Non-invasive BCIs claim the chief of research grants in this area. Non-invasive BCIs have a fine temporal resolution, ease of use, transferability, and affordability. One key factor In this does include their non-invasive nature though the same thing does affect the quality and non-invasive BCIs are more susceptible to noise than the invasive BCIs

Training for EEG-based BCIs:

Training the EEG control is a time-taking process. It usually takes patients several months to accustom themselves to it.

2.6.2 Electromyography

Electromyography is a compound word that could mean 'electrical mapping of muscles'. Electromyography uses a device called an electromyogram that produces an electromyograph. An electromyograph detects electric potential signals muscles are simulated and maps out the information on an electromyograph. The neuromuscular activity measured through the EMG exhibits a correlation with the exertion of force for the movement of muscles. EMG measurements need to be mapped with the muscle activity to train the model and a few methods exist such as neural networks [64], neuro-fizzy classifiers [65], and support vector machines.

2.6.3 Motion-based control

Apart from the option of electromyography and electroencephalography, there is also a method of motion-based control. Motion-based control relies on another hand, for its motion to be replicated onto the hand affected by paralysis. The hand could be an entirely external source – someone else's hand, or in the case of hemiparesis or hemiplegia – a paralysis that affects one side of the body, the healthy hand which is the 'chiral equivalent' may be used to detect motion and then the signal replicated onto the affected hand.

2.6.4 Comparison

Electroencephalography, electromyography, or motion-based control have each their respective trade-offs. When it comes to EEG, it is very much likely for the authors if they chose to attempt this approach, would work with non-invasive EEGs. EEGs claim a good amount of time in just setting up and the training period they require from the patients to get accustomed to it can last for months. EEG is besides, too complex to obtain the required information and is rarely mobile, rendering confinement. The only option for a patient willing to undergo physiotherapy. EMG on the other hand, presents another unique set of difficulties. It requires the knowledge of 'neuromuscular thresholds' of the specific patient to work ideally; thus, they may not be an effective way to achieve control where a degree of general application is sought, as in the case of the authors of this thesis. Motion-based controllers thus make for a more easily implementable solution. Motion-based sensors require a healthy hand for motion; thus they may be restricted to patients with hemiparesis or hemiplegia.

CHAPTER 3: METHODOLOGY

3.1 Biomechanics of a Human Hand

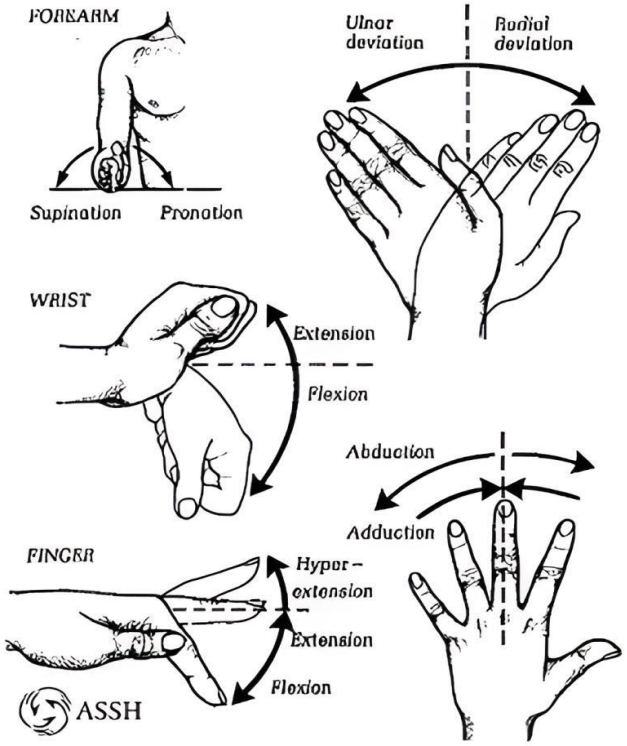


Figure 17 Movements that a human hand is capable of

According to research [20], the human hand consists of 19 bones, 19 joints, and 29 muscles. The fingers contain 14 phalanx bones and 5 metacarpal bones, which form 14 joints, allowing for various movements of the hand. The primary movement of the finger involves flexion-extension of three rotational joints (labeled in Figure 1): metacarpophalangeal (MCP), proximal phalangeal (PIP), and distal phalangeal (DIP) joints. To ensure the exoskeleton can facilitate natural flexion-extension motions with three degrees of freedom (DOFs), the design of the hand exoskeleton must be carefully considered. Additionally, the system must be capable of transmitting interactive forces to

the finger to provide resistive forces to the finger in the extension direction during object grasping.

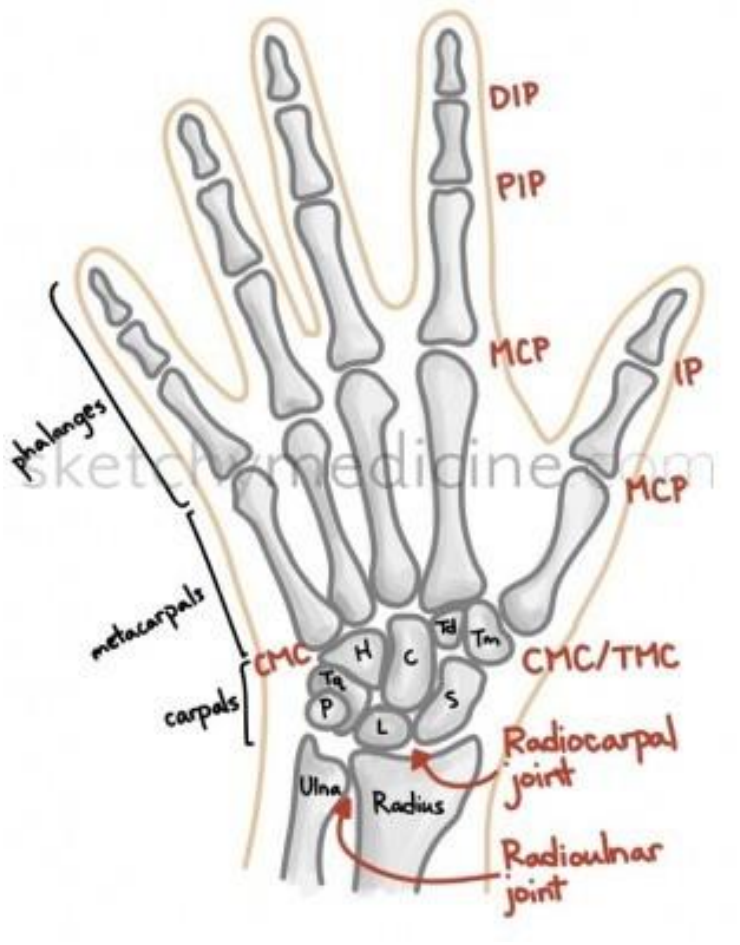


Figure 18 Labelled image showing the finger joints and bones Courtesy of Sketchy Medicine

Hand exoskeletons have come about in recent years to assist in the process of physiotherapy. Stroke patients usually undergo treatment wherein physiotherapists recommend exercise, which the patients undergo at specialized training centers or other spaces, under supervision. Hand exoskeletons have come about as aids in the process of rehabilitation. Rehabilitation of hands to whatever extent; either it is improved motor function or complete motor recovery, is always well-received by the patients, because not

only does such a line of treatment allow for better performance in ADL, but so does the mental well-being of the patient improve. To achieve this feat, hand exoskeletons are designed to exhibit a functional range of motion (FROM) – this is the minimum range of motion required to perform ADL. The functional positions of the three joints in human fingers: the Metacarpophalangeal (MCP) joint, the Proximal Interphalangeal (PIP) joint, and the Distal Interphalangeal (DIP) joint, are crucial for the positions that the hand makes such as pinch or grasp, during ADL. It is for this reason that the design of an effective exoskeleton must, of necessity, consider the functional positions of these joints for optimal functionality and utility for the user. Uncertainties in the identification and characterization of objects of ambiguous shape and size lead to inaccurate models for the object's position and orientation in space. Inaccurate models of the sort, of risk positioning errors in tasks that may involve finger placement. This can potentially lead to the object at hand escaping the range of grasp, thus rendering the completion of the task more difficult. The problem at hand can be solved, if the hand exoskeleton may have the ability to adapt to the geometry of the object that needs to be grasped, to some extent. It needs to be appreciated here that friction is the force that makes gripping or grasping possible. If the friction between the object grasped and the fingers is less than the vertical force directed towards the earth, the object will fall from its resting position, resulting in grasping failure. Grasping can be called a complex 'maneuver' in that a successful grasp rests on two key aspects. The contact force should be greater than the vertical force, to maintain the object in the position it is in, yet the force must not exceed a limit that may threaten the integrity of the object in hand. To overcome this problem and arrive at a compromise, exoskeletons have sensors and control systems to adjust the applied forces to maintain the integrity of the object as well as the grip.

Table 1 Functional Range of Motion of a Human Digit

Joint	*Functional Range of Motion*	*Average Functional Range of Motion*
MCP	62.7° [2]	61° [3]
PIP	78.3° [2]	60° [3]

This is particularly important in applications where the objects being manipulated are delicate, fragile, or require a precise touch.

According to research [20], the average and median coefficients of friction across 65 objects and 3 surfaces are 0.300 and 0.255, respectively. These coefficients of friction are essential to understanding the forces required to grasp and manipulate objects. Based on another study [41], the required grasping force for each finger is approximately 7.3N. Understanding the coefficients of friction and the required grasping forces can help inform the design of exoskeletons that can provide the necessary forces to maintain a secure grip on objects.

3.2 Design Parameters for the Exoskeleton

- **Lightweight**: Should be made from a suitable lightweight material to ensure that the exoskeleton is ergonomic.
- **Smooth**: The mechanism should function without any sudden jerks.
- **Top mounted**: The mechanism should be mounted on the dorsal side of the hand to prevent it from interfering with the grasp.
- **Natural Curve**: The mechanism should follow the natural logarithmic path of a finger.
- **Conformance**: The exoskeleton mechanism should be able to adapt to the geometry of the object being grasped.

- **Compliance**: The exoskeletons should be comfortable and ergonomic for the user.
- **Axis alignment**: The rotational axis of the mechanism should be perfectly aligned with the joints of the human finger to ensure the comfort of the patient while the exoskeleton is being used.

3.3 Types of Exoskeleton Mechanisms

3.3.1 Tendon-driven Mechanisms

Tendon-driven mechanisms are a commonly used system that uses tendons as the primary means of transmitting force and motion. However, there are some challenges associated with this method. One significant issue is the need for precise alignment with the axis of the finger to achieve optimal results. Furthermore, mounting the tendons can pose difficulties as they should not pass through the palm as it would hinder grasping objects. Tendon-based mechanisms can also cause the distal joint to move faster than other joints, leading to unnatural finger movements. Additionally, because tendon-driven mechanisms have been extensively studied, our goal of developing a novel approach would be challenging to achieve using this method.

3.3.2 Rotational Systems

In the early stages of developing hand exoskeleton systems, engineers explored a new rotational system to optimize the transmission of torque between links. The aim was to generate maximum force at the end effector. They worked on a seesaw-type mechanism (called mechanism 21), which had the advantage of allowing significant force variation by changing the pivot point of the rigid links. However, this design posed a problem when trying to follow the natural trajectory of the fingers. We have chosen to pursue this design and using Differential Evolution have been able to create a mechanism that follows the natural trajectory of a human finger with an accuracy of 5 degrees.

3.4 Final Concept

The proposed design was first validated in Linkages. The design in question is using only four-bar mechanisms, with two mechanisms being path generation mechanisms and one of the mechanisms being a function generator. This approach allowed us to keep the mechanism simple and improve its reliability by reducing the number of failure points in the system.

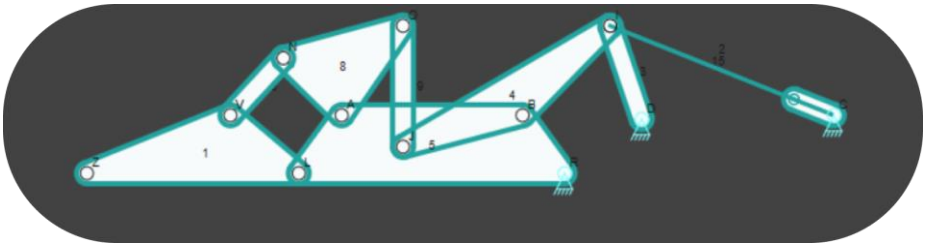


Figure 19 Initial kinematic design of the mechanism

3.5 Differential Evolution

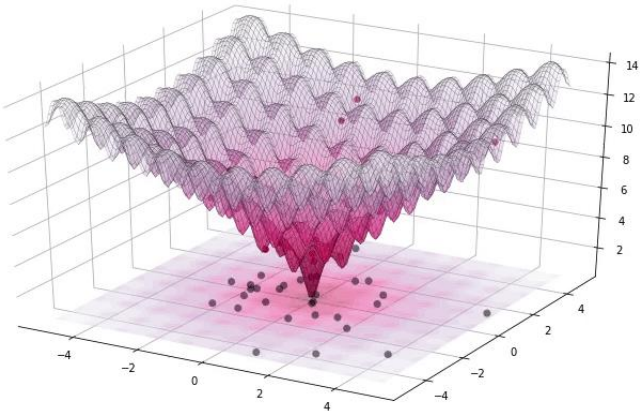


Figure 20 Differential Evolution Algorithm Visualized. Can be viewed on the link: https://pablormier.github.io/assets/img/de/ackley.gif

Differential Evolution (DE) is a highly effective optimization algorithm that was invented in 1997 by R. Storn and K. Price. It is used to solve black-box optimization problems,

which involve finding the minimum value of a function f(x) without knowledge of the analytical form of the function or its derivatives.

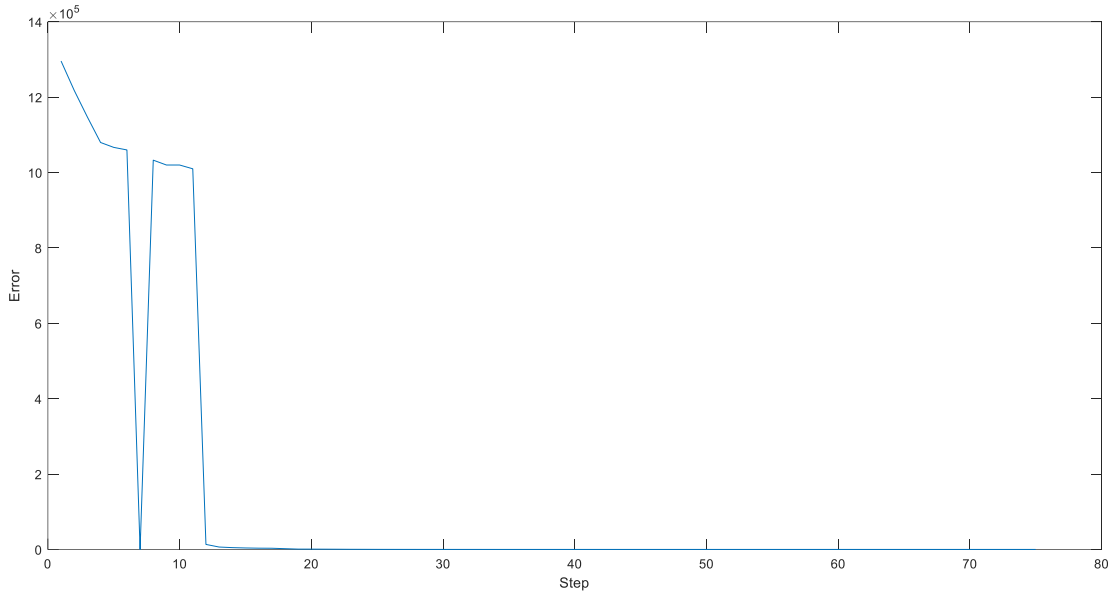
Differential evolution is an algorithm for optimizing any set of solutions. It does not require or concern the underlying problem. If the problem is simple enough to be reduced to a few values, differential evolution runs iterations until it arrives at the optimum solution. Optimization is at the heart of differential evolution, and it is for this reason that it can find application wherever the need to optimize may arise.

At the heart of the DE algorithm is the process of mutation, crossover, and selection. The mutation is a process in which a new candidate solution is generated by adding a weighted difference between two randomly chosen solutions from the current population to a third solution. Crossover, on the other hand, involves the recombination of two or more candidate solutions to create a new one. Finally, selection involves choosing the best candidate solutions from the population to form the next generation of solutions.

The DE algorithm also employs a fitness function to evaluate the quality of candidate solutions. The fitness function measures how well a candidate solution performs on the optimization problem and guides the search toward better solutions. Equations from the theory of machines were used in the code to construct the fitness function. The code can be found in APPENDIX II of this report. The mechanism had ten joints – two for each finger.

One of the strengths of DE is its ability to explore a large search space efficiently. This is achieved using a population-based approach, which enables the algorithm to search multiple regions of the search space simultaneously. Additionally, the use of mutation and crossover operators helps to maintain diversity in the population, preventing the algorithm from getting stuck in local optima.

The graph presented herein plots error against step for the optimization of our design using differential evolution.



3.6 Mathematical Model

The mathematical model will reference the four-bar function that is included in Appendix II. Additionally, for a more mathematical explanation, the corresponding mathematical equations which serve as the basis for the four-bar function are given in Appendix I.

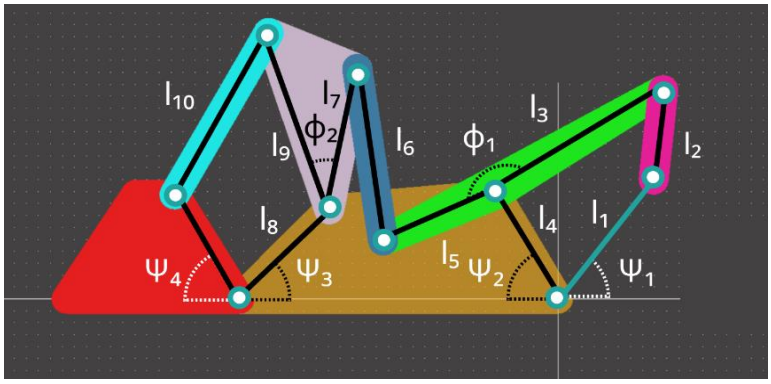


Figure 21 Variables identified on the model

$$\theta_3', \theta_4' = fourbar(l_1, l_2, l_3, l_4, \theta_2, mechnumber = \text{"1"})$$

$$\theta_4 = \theta_4' - \varphi_1 + \varphi_2$$

$$\theta_3 = \theta_3' - \varphi_1 + \emptyset_1$$

$$\alpha = \theta_4 - \pi$$

$$\theta_6', \theta_7' = fourbar(l_{8-5}, l_5, l_6, l_7, \theta_3 + \theta_{8-5} mechnumber = \text{"2"})$$

$$\theta_7 = \theta_7' - \theta_{8-5} + \emptyset_2$$

$$\theta_{10}', \theta_{8-10}' = fourbar(l_8, l_9, l_{10}, l_{8-10}, \theta_7, mechnumber = \text{"3"})$$

$$\theta_{8-10} = \theta_{8-10}' - \varphi_3 + \varphi_4$$

$$\beta = \theta_{8-10} - \pi$$

3.7 CAD Model

After the dimensions of the exoskeleton were found using the optimization algorithm and the mathematical model above, the CAD model of the design was created using the dimensions of the generated mechanism.

Iterations 1 and 2 were created using different dimension bounds in the differential evolution algorithm. These iterations were then analyzed in MSC Adams after which the error between the required angles and the generated angles was calculated. The dimension bounds were then revised to ensure that the error was minimized.

3.7.1 Second Last Iteration

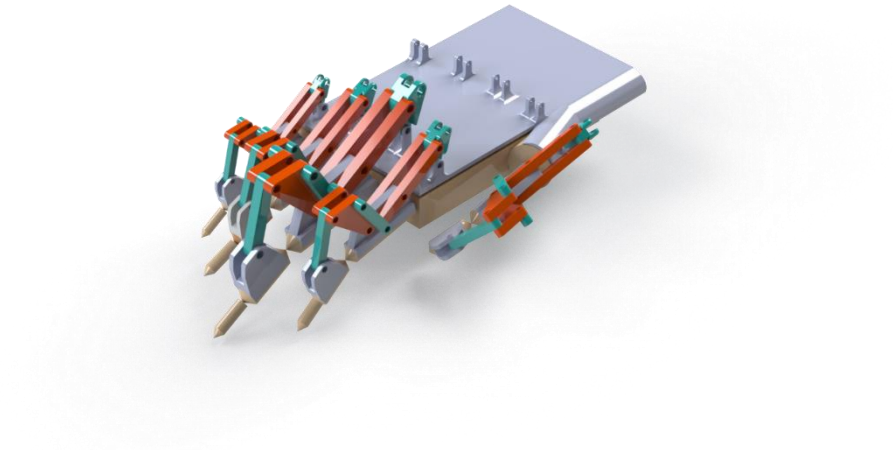


Figure 22 CAD Model Rendering of the Exoskeletons' second last iteration.

3.7.2 Last Iteration

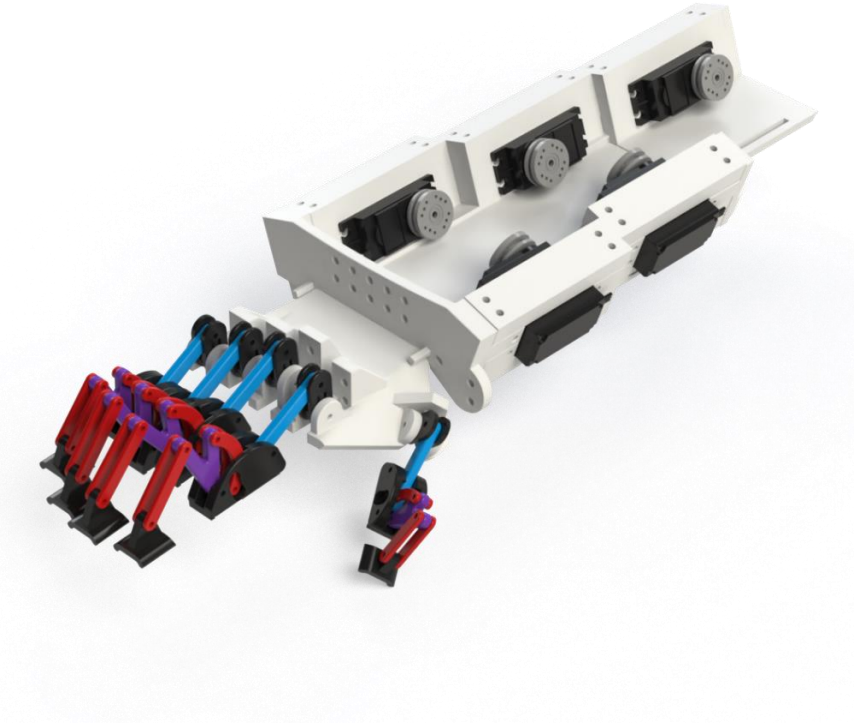


Figure 23 CAD Model Rendering of the Exoskeleton's last iteration

3.8 Control Mechanism

Three types of control mechanisms were identified and expanded upon in chapter 2. Control could be achieved by using electromyography (EMG), electroencephalography (EEG), or motion-based control achieved through a ‘cyberglove’. While EEG and EMG did sound attractive, motion-based control was the cost-effective solution, and more technically feasible in our case. Motion-based control involved the use of a cyberglove. A cyberglove had flex sensors in it. These flex sensors, allowed us to measure the digits in the hand that are bent. The authors of the present work used an already available in-house cyber-glove present in the Robotics and Intelligent Systems Engineering (RISE) lab at SMME.

Figure 24 Image showing the Cyberglove present in the RISE Lab

The raw analog data obtained from the flex sensors was subject to noise and fluctuations. ‘Noise’ here does not refer to clamor or commotion, but to disturbances in the signal that render the quality of the signal below that which is required for efficient processing. The raw data needed conditioning to form any semblance of ‘analyzable’ data to proceed. In our case, the signal obtained by the Arduino would only contain small rapid fluctuations which could easily be catered for by using an exponential smoothing algorithm. An exponential smoothing algorithm is used for discrete values of time and it filters out any noise with high frequency. It follows a ‘seemingly’ simple mathematical model which is demonstrated below.

$$s_0 = x_0$$

$$s\text{-}t = \propto x_t + (1{-}\propto)s_{t-1}$$

Where, x_t is the $raw\ data$, s_t is the $conditioned\ data$, and $\propto\ is\ smoothing\ factor$ which can take on values $0< \propto <1$. A smoothing factor value of 1 corresponds to no filter while a value of 0 would lead to constant signal output. Finally, a value of 0.6 would satisfy the time constant requirements. The time constant here refers to the period within which the smoothed response reaches $1 - \frac{1}{e} = 63.2\%$ of the value of the initial signal. The mathematical relation between the time constant and the smoothing factor is given below.

$$\alpha = 1 - e^{-\frac{\Delta T}{\tau}}$$

Where ΔT is the sampling interval and for fast sampling time, $\alpha \approx \Delta T/\tau$.

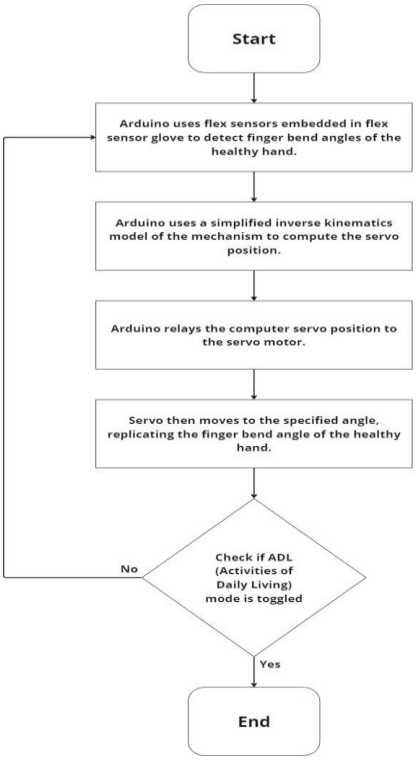


Figure 25 Flow Chart Depicting the Control Methodology of the Exoskeleton.

There are flex sensors embedded within the cyberglove. The sensors act as variable resistors. The sensors have a basic value of resistance – the resistance whereat the sensors do not experience any strain or bent. The bent introduced in the sensors causes their resistance to change. The sensors are connected in series with resistors with the same basic value of resistance of the sensors. The circuit hence formed acts as a potential divider, and in the event of application of a bent; the voltage distribution across the circuit varies. The voltage output from the circuit is then processed by a control system and can be used to control robotic movements. The flex sensors, hence detect the bend angles within individual fingers. The bend angles so obtained, are used to determine the potential position of the servomotor on the Hand Exoskeleton, that would mirror the 'pose' of the healthy hand. This 'requisite' position of the servomotor is computed through an inverse kinematics formulation by Arduino. The 'desired' pose is achieved when the computed joint angles are relayed to the servomotor, on the Hand Exoskeleton; thus achieving the same bend angle as the cyberglove.

Electronic Components:

The electronic components that are going to be used in the exoskeleton are Arduino UNO, MG 996R Servo motors, Flex sensors, and Resistors.

The illustration generated on TinkerCAD™, shows the circuit created in the prototype. The system was powered by two 18650 cells each producing 3.7 V. Arduino operates on a voltage of 5V, so a buck converter was used to step-down the voltage to 5 V. The servomotors were powered by the 5 V output of the buck converter, and connected to Arduino through the PWM pins 5, 6, 9, 10, and 11. The flex sensors were connected in series with fixed resistors, and likewise powered through the 5 V supplied by the buck converter.

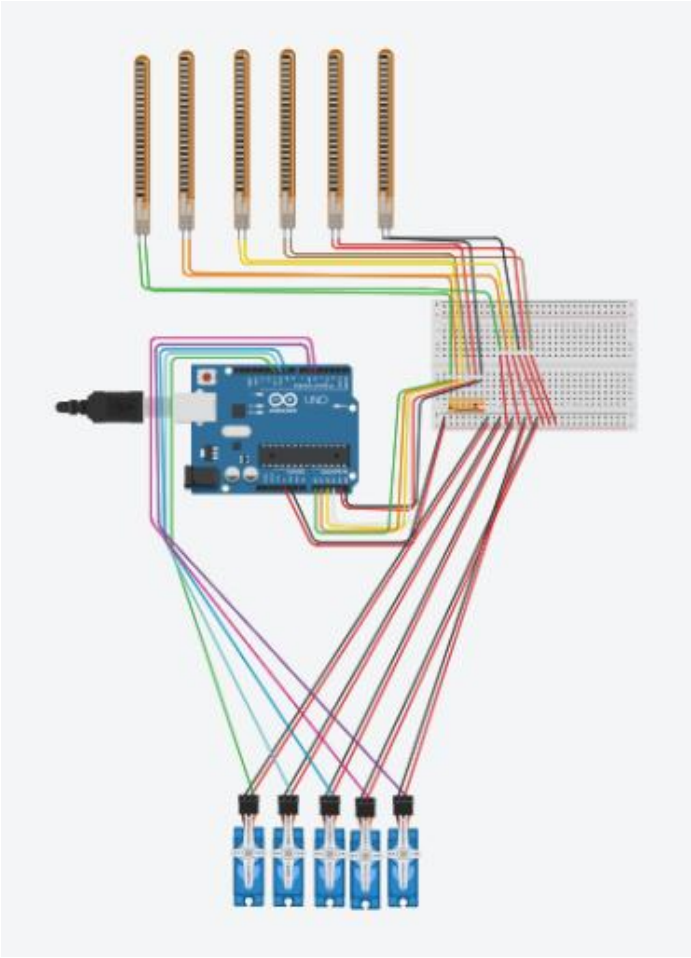


Figure 26 TinkerCAD Circuit Diagram

3.8.1 Modes incorporated into the Exoskeleton

3.8.1.1 ADL Mode

In this mode, the exoskeleton will facilitate the patient in carrying out basic activities of daily living by making use of the Cyber glove.

3.8.1.2 Rehabilitative Mode

In this mode, the exoskeleton will monitor the patient's response to various rehabilitative exercises. The physiotherapist will monitor the patient's progress and make any required changes to the rehab plan that are required so that the recovery of the patient is accelerated.

CHAPTER 4: RESULTS AND DISCUSSIONS

4.1 Dimensions of the Mechanism

The lengths of the mechanism for the index finger obtained after optimization are summarized below:

Table 2: Link lengths and Offset Angles obtained in the final iteration for the index finger.

Number	1	2	3	4	5	6	7	8	9	10
Length (cm)	2.6	1.4	3.3	2.8	2.0	2.9	2.4	2.1	3.1	3.1
φ (rad)	0.9	1.0	0.8	1.0	-	-	-	-	-	-
$\emptyset$ (rad)	3.0	0.5	-	-	-	-	-	-	-	-

For reference, see the picture below:

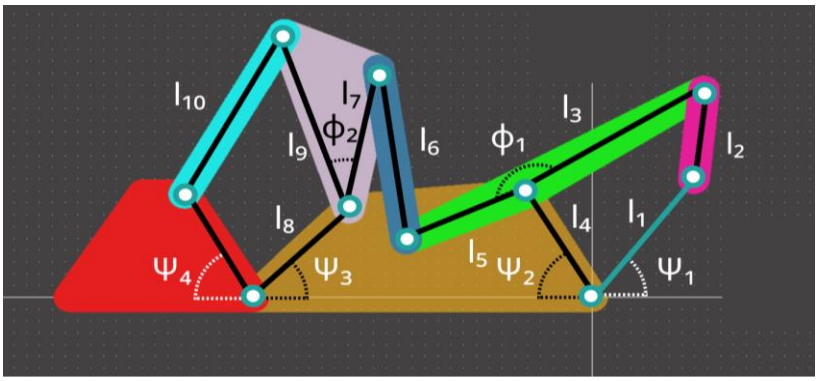


Figure 27 MSC Adams Model

4.2 FEA

4.2.1 Set-up

The CAD model, which was created in SolidWorks, was converted to the FEA model by applying necessary boundary conditions and meshing.

As the linkage mechanism is a dynamic system, with input force continuously acting at the input link and links are in motion so first of all, multi-body dynamics was required to calculate forces at each joint with continuous change in the angles of the finger due to motion at a certain applied force.

The figure below represents a snapshot of the mechanism in MSC Adams for multibody dynamic analysis.

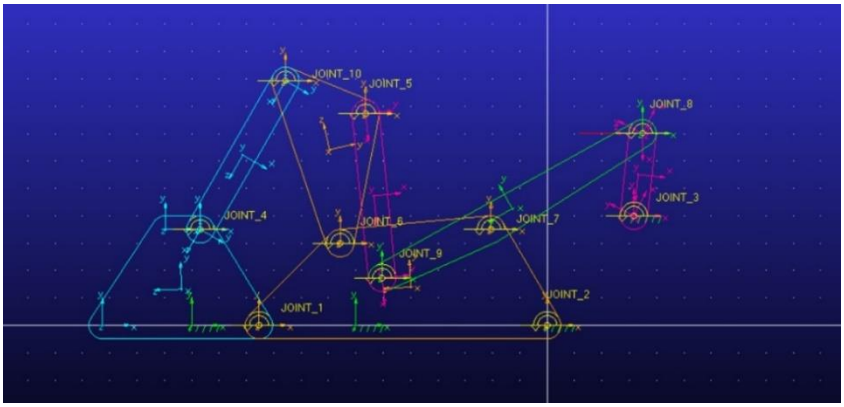


Figure 28 Analysis being run on MSC Adams

The curves were generated with force on the y-axis and time on the x-axis, meanwhile, the relationship between angle and time was calculated so at any instant, the force vs angle relationship was found by merging the result of both analyses. For joint 1, the Maximum force was found to be at an instant slightly before 0.025 seconds and at that instant, values of angles α and β is 27.58^{o} and 107.89^{o} respectively.

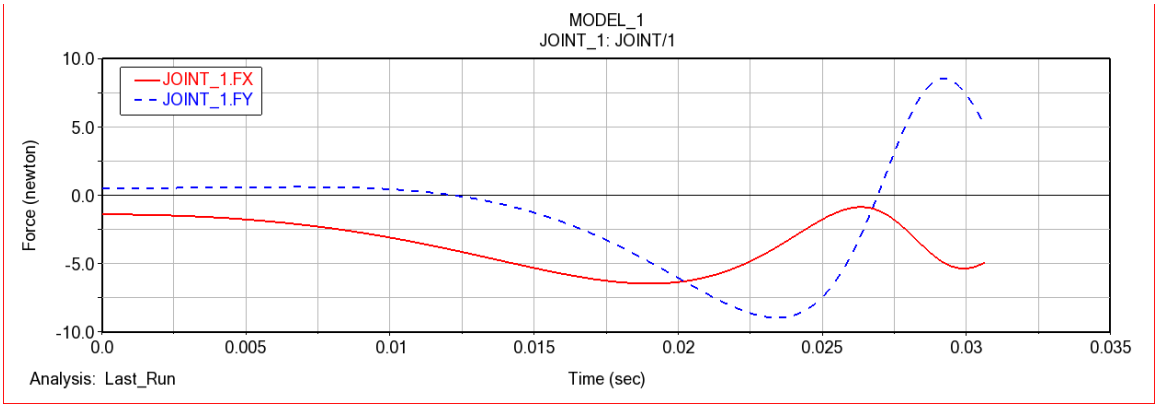


Figure 29 Joint 1 Force curve

Similarly, joints 5,6,7, and 9 have critical force values at this angle too. Force curves of these joints are as shown:

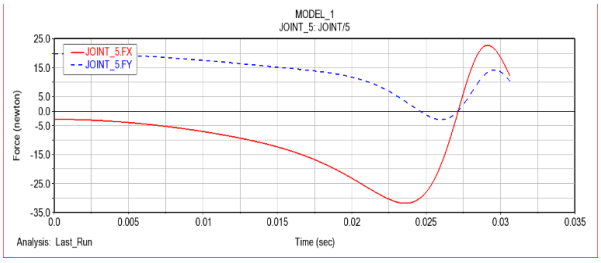


Figure 28 Joint 5 Force curve

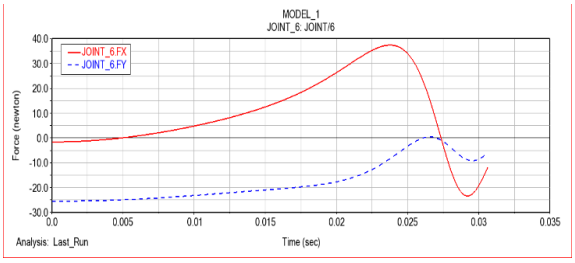


Figure 29 Joint 6 Force curve

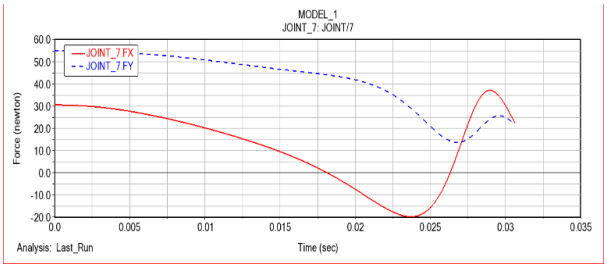


Figure 30 Joint 7 Force curve

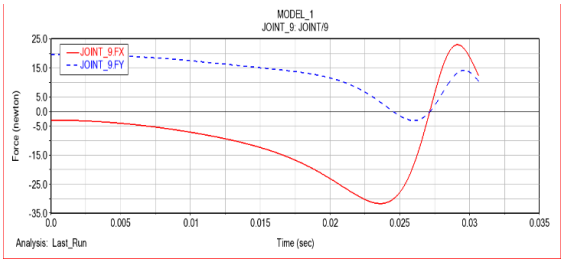


Figure 31 Joint 8 Force curve

For Joint 2, maximum forces were found to be at the instant when force is just applied, i.e., at t=0[s]. At this instant, values of angles α and β is 0^o and 0^o respectively.

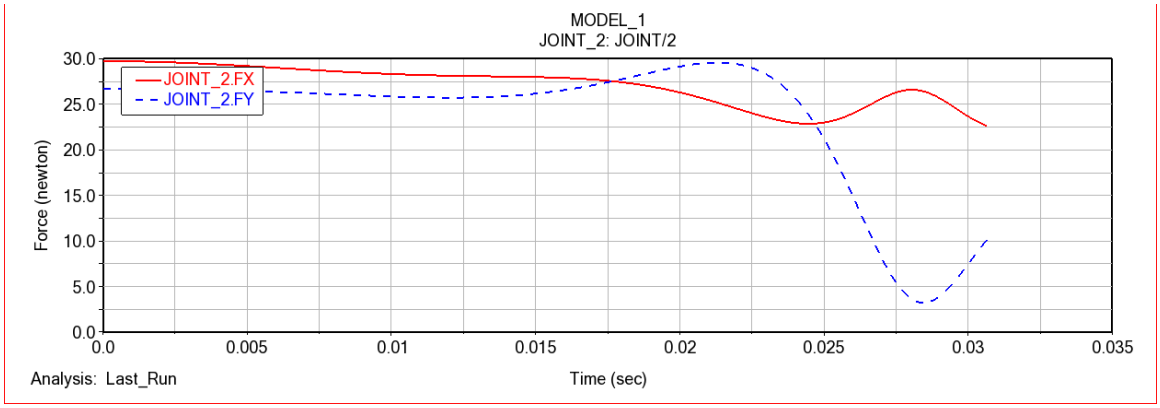


Figure 32 Joint 2 Force curve

Similarly, joints 3 and 8 have the same critical angles as can be seen in the curves below:

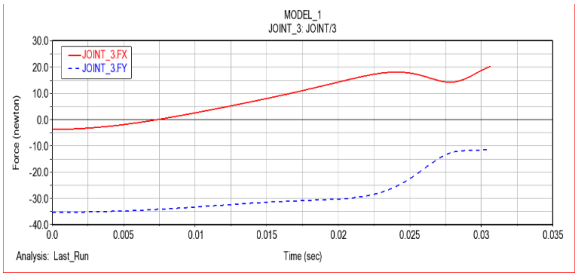


Figure 33 Joint 3 Force curve

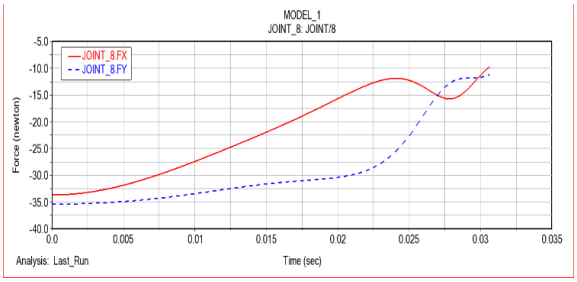


Figure 34 Joint 8 Force curve

For the rest of the joints, the maximum forces appeared at the instant just before 0.03 seconds, and at that instant, values of angles α and β are 44.94° and 154.58° respectively. These joints include 4 and 10 as shown below:

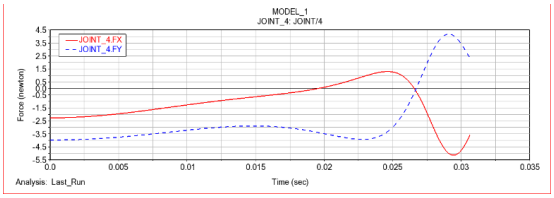


Figure 35 Joint 4 Force curve

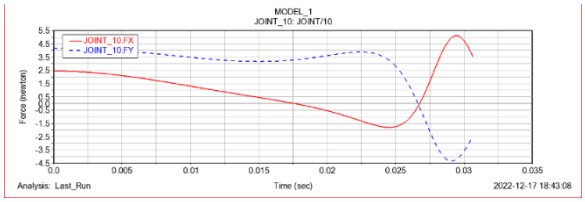


Figure 36 Joint 10 Force curve

So, the results of MBD can be summarised in the form of three load cases as shown in the table below. The angles α and β are the angle of the Proximal and Intermediate phalanges with the horizontal axis measured counter-clockwise:

Table 3 Joint Angles for the identified load cases

Angle	*Load Case 1*	*Load Case 2*	*Load Case 3*
α	44.94 °	0 °	27.58 °
β	154.58 °	0 °	107.89 °

4.2.2 Mesh formation

The figure represents the meshing done in COMSOL for subsequent analysis. Normal element size was used in meshing and loads were applied at each joint with corresponding alpha and beta angles found from multi-body dynamics.

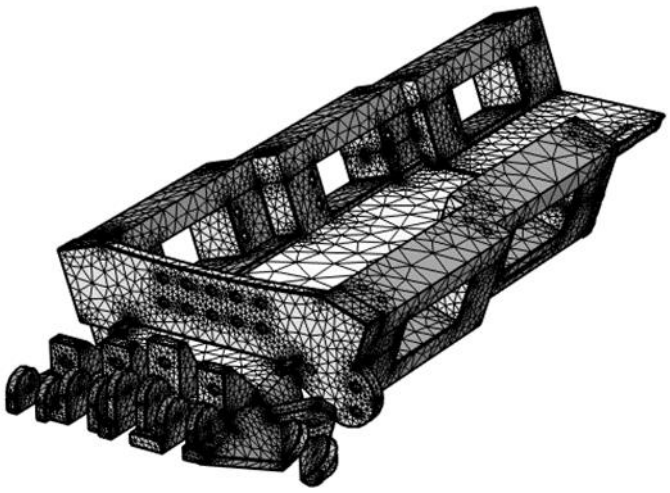


Figure 37 Meshed Base Plate

4.2.3 Boundary Conditions

The forces i.e., F_x and F_y at each joint were applied using boundary load configuration and the whole mechanism was made static through fixed constraint command.

4.2.4 Stress Contours

After the application of necessary boundary conditions, the study was run, and stress plots were generated for each load case.

These plots depict the Von Mises stress induced in the model due to the applied loadings and the unit used to represent this distribution is N/m^2.

For load case 1, both ternaries and one input binary were found to have maximum stresses as can be shown in the plots.

The stress plots for load case 1 follow:

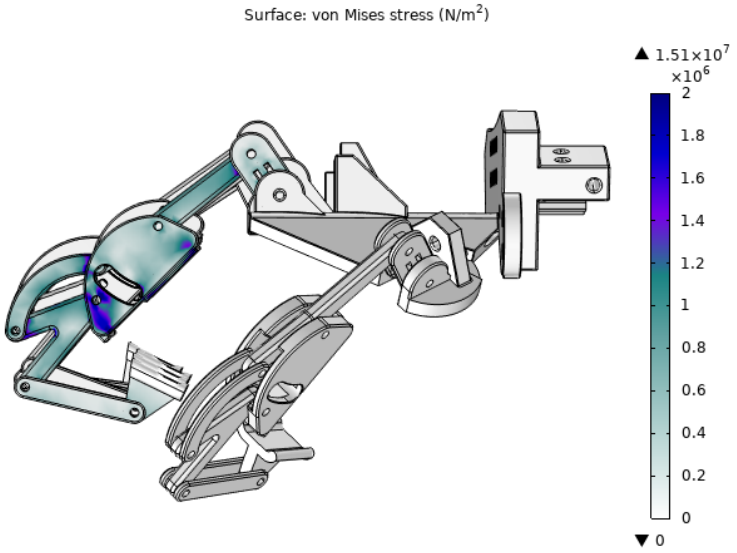


Figure 38 Stress Contour for Load case 1 (A)

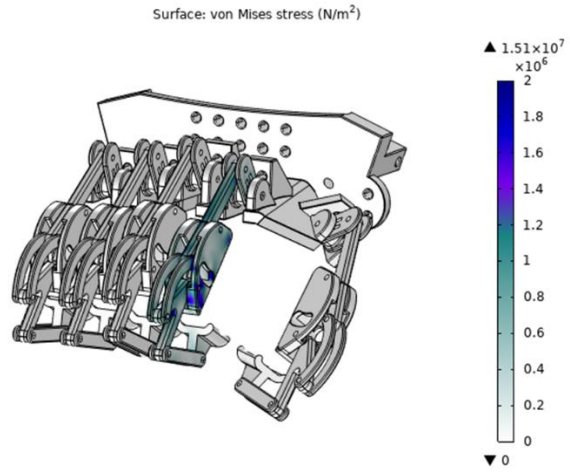


Figure 39 Stress Contour for Load case 1 (B)

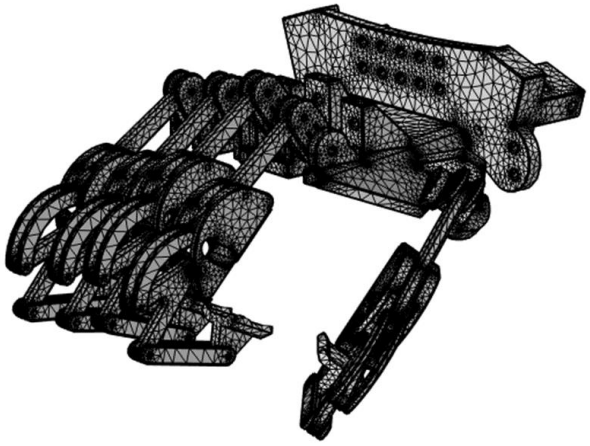


Figure 40 Meshed Exoskeleton

This load case occurs near the end of the flexion motion. The areas especially near the joints have maximum stress on both ternaries and input links.

Below are the plots showing the results from load case 2:

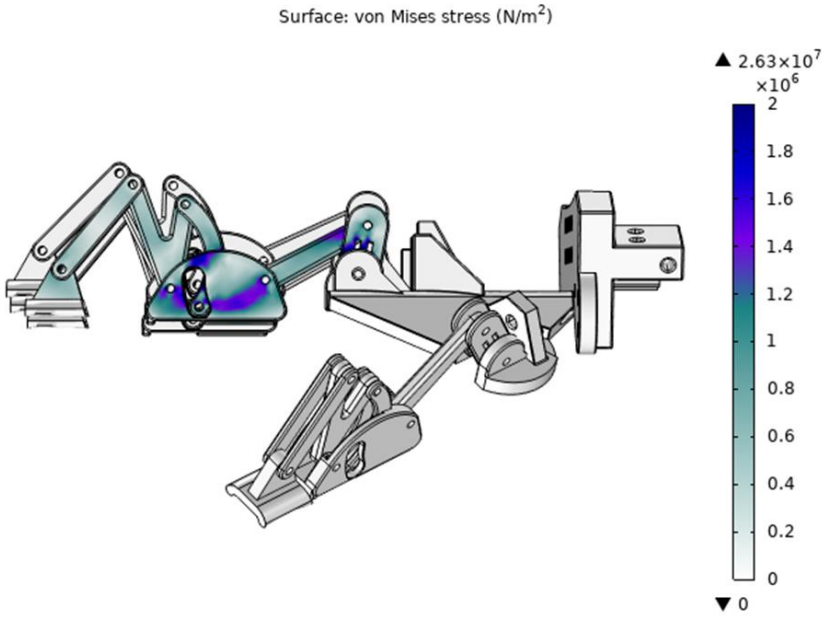


Figure 41 Stress Contour for Load case 2 (A)

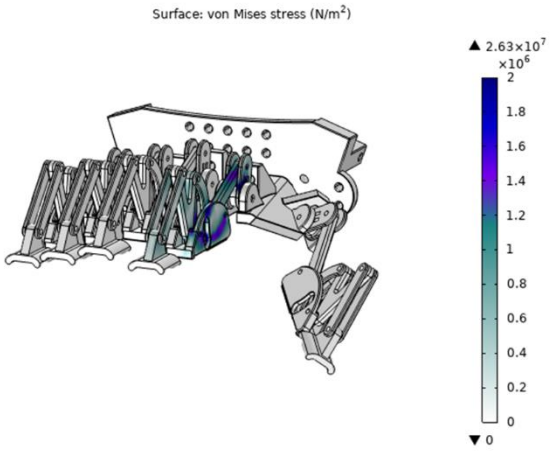


Figure 42 Stress Contour for Load case 2 (B)

This load case is when the motion of the finger is about to start. So, in this case, only the input link and first ternary bear the maximum stress as the forces are maximum in these links to initiate the motion.

For load case 3, the following plots were found:

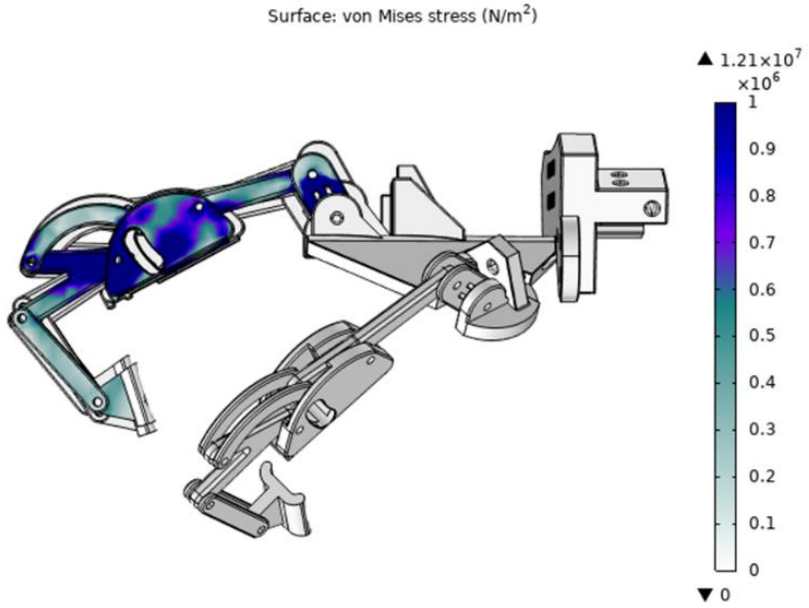


Figure 43 Stress Contour for Load case 3 (A)

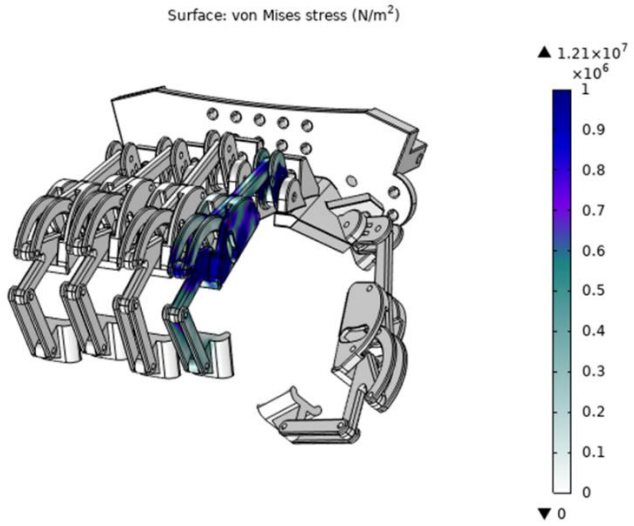


Figure 44 Stress Contour for Load case 3 (B)

This load case occurs at the midway of the whole ROM of the finger. It has the maximum stress in the MCP support that will be mounted on the finger.

This mechanism is mounted on a base plate. The base plate was also analysed in COMSOL for structural integrity.

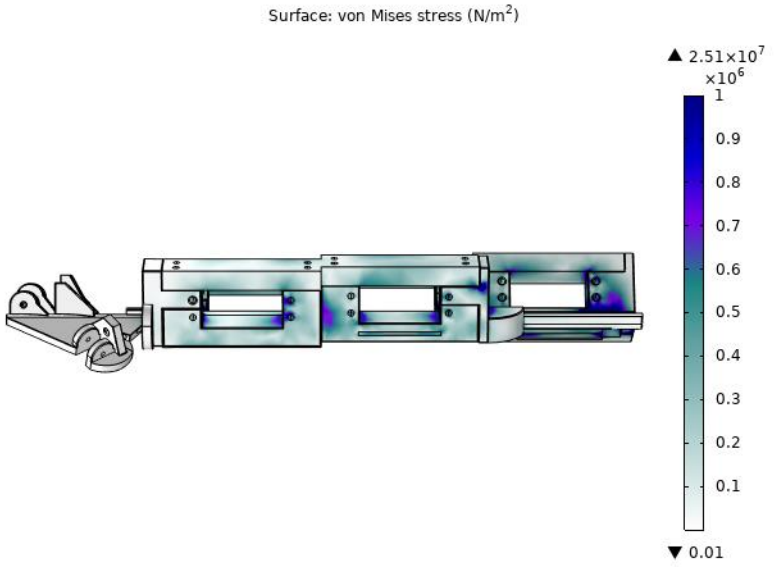


Figure 45 Base Plate FEA

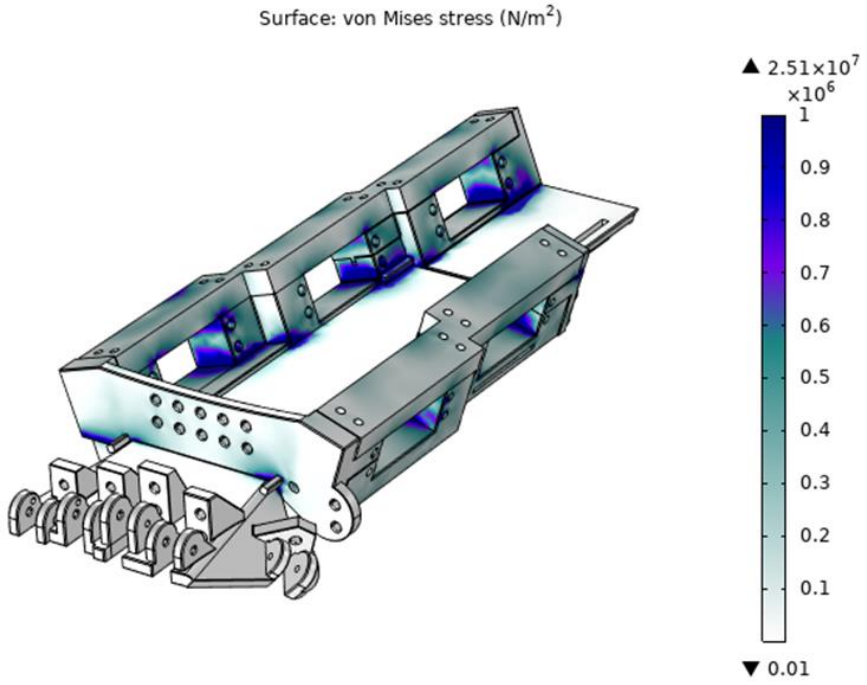


Figure 46 Orthogonal View of Base Plate

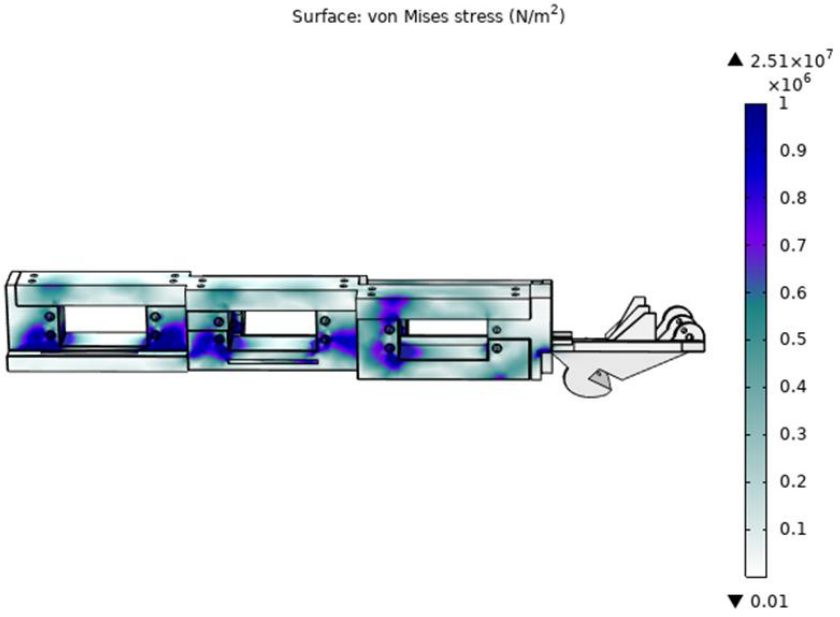


Figure 47 Left View of Base Plate

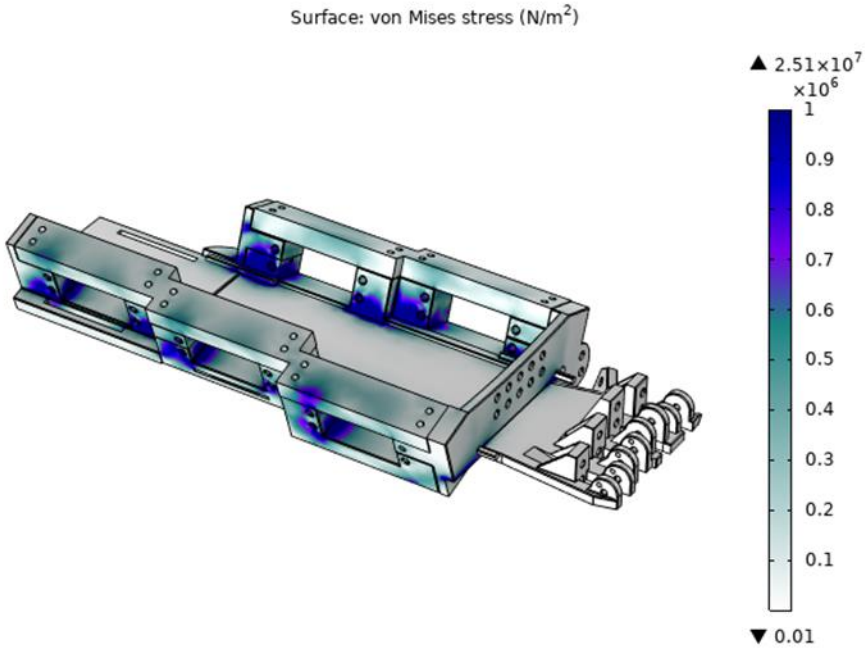


Figure 48 Orthogonal View of Base Plate

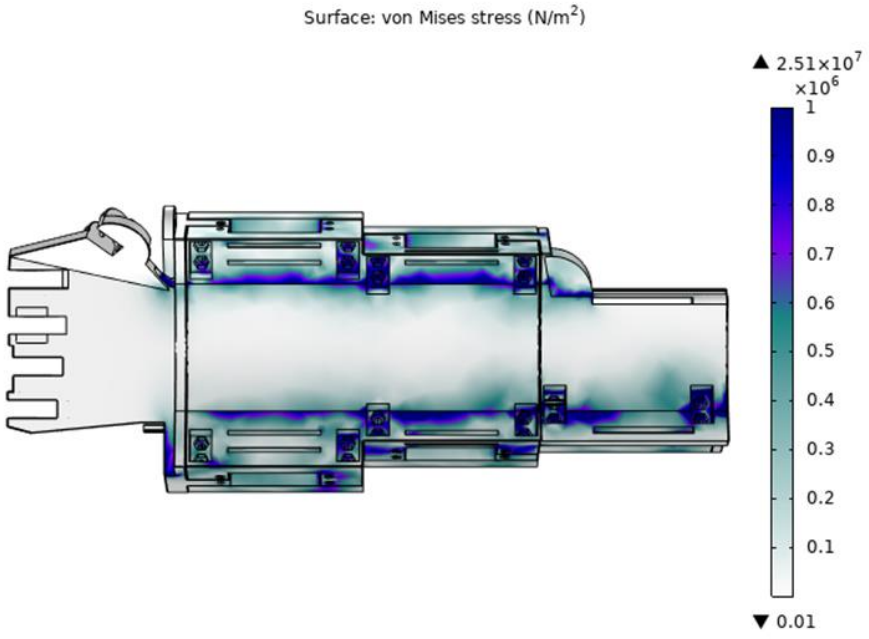


Figure 49 Top View of Base Plate

4.2.5 Von Mises Stress and Displacements

The following results were obtained through the analysis.

Results from Von Mises Stress and Displacement Plots:

- Maximum Von Mises of 15.1 MPa and Displacement of 87.752 µm in the first case.
- Maximum Von Mises of 26.3 MPa and Displacement of 137.112 µm in the second case.
- Maximum Von Mises of 12.1 MPa and Displacement of 144.969 µm in the third case.

Ploylactic acid (PLA) was used in the 3D Printing of the Base Plate.

- Young's Modulus: 4.107 GPa
- Poisson Ratio: 0.35
- Density: 1.287 g/cm^3

Polyethylene terephthalate Glycol (PET-G) was used in the 3D printing of the linkages. PLA was preferred over PET-G for the 3D Printing of the base due to financial constraints. Important properties of PET-G are:

- Young's modulus: 4.107 GPa
- Poisson's ratio: 0.33
- Density: 1.24 g/cm^3

4.3 Fabrication and Prototyping

4.3.1 Cardboard Model

The dimensions of the links that were already found were used to form a model made up of cardboard. Binaries and ternaries were cut off from the cardboard and were joined using thumb pins (to make the joints revolute).

Figure 51 Links that were made for initial design validation

It was done to check whether the dimensions found are realistic and to get a rough idea that how the linkage chain works. This crude test proved that the dimensions are realistic, and the chain is working.

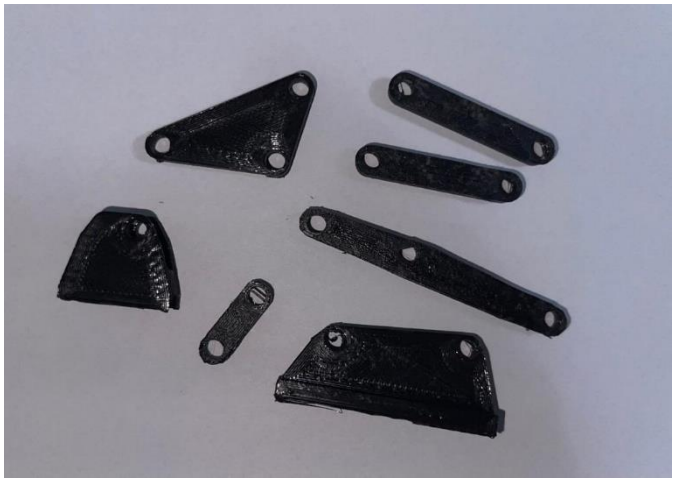


Figure 52 3D Printed Links

4.3.2 Prototyping

Links were 3D printed using PET-G (Polyethylene terephthalate glycol) material. For this initial prototype, some issues were found. There were some edges in the 3D model which were smaller than the nozzle of the 3D printer which means those edges were not printed with the required thickness. This was taken care of in our final prototyping phase

The rest of the structure of the hand exoskeleton was 3D printed using PLA – polylactic acid.

Thermoplastic Polyurethane was not used due to its elastic properties. TPU is prone to bending which makes it unsuitable for hard robotic mechanisms. 3D Printing was carried out with PLA for the base plate whereas PET-G was specifically used in printing the links. PET-G was chosen for the links due to its greater resistance to wear. Links are the components that are in motion. Links within adjacent fingers occasionally brush off against each other. These lend a greater cause for the links to wear. 3D printed parts include a base plate and the links. The base plate has been designed with a dihedral shape. The base plate has several mountings of cell-holders. The cell holders are arranged in a linear fashion across the length of the baseplate, along the two ends. The dihedral design of the baseplate conforms well to the natural curve of the wrist and the lower arm. The dihedral design of the baseplate distributes the weight of the mountings. The cell-holder mountings are fastened onto the baseplate with M3 bolts. Right below the cell holder mountings, the space is occupied by motors – six in number arranged along the ends, across the length of the baseplate. The links in the mechanism are fastened with each other using M2 nuts and bolts. There are pulleys installed along the length of the baseplate, adjacent to the motors. The pulleys support a string that passes over onto the links. The strings join the links in a single mechanism allowing them to react to external forces. If there had been no strings, the links would have hung loose – irresponsive to the pulses of the actuators. The string used is a fishing line, appropriated here due to its strength. The mechanism formed by the links is an underactuated mechanism. The fingers of the mechanism are mounted on the fingers of the user by bands.

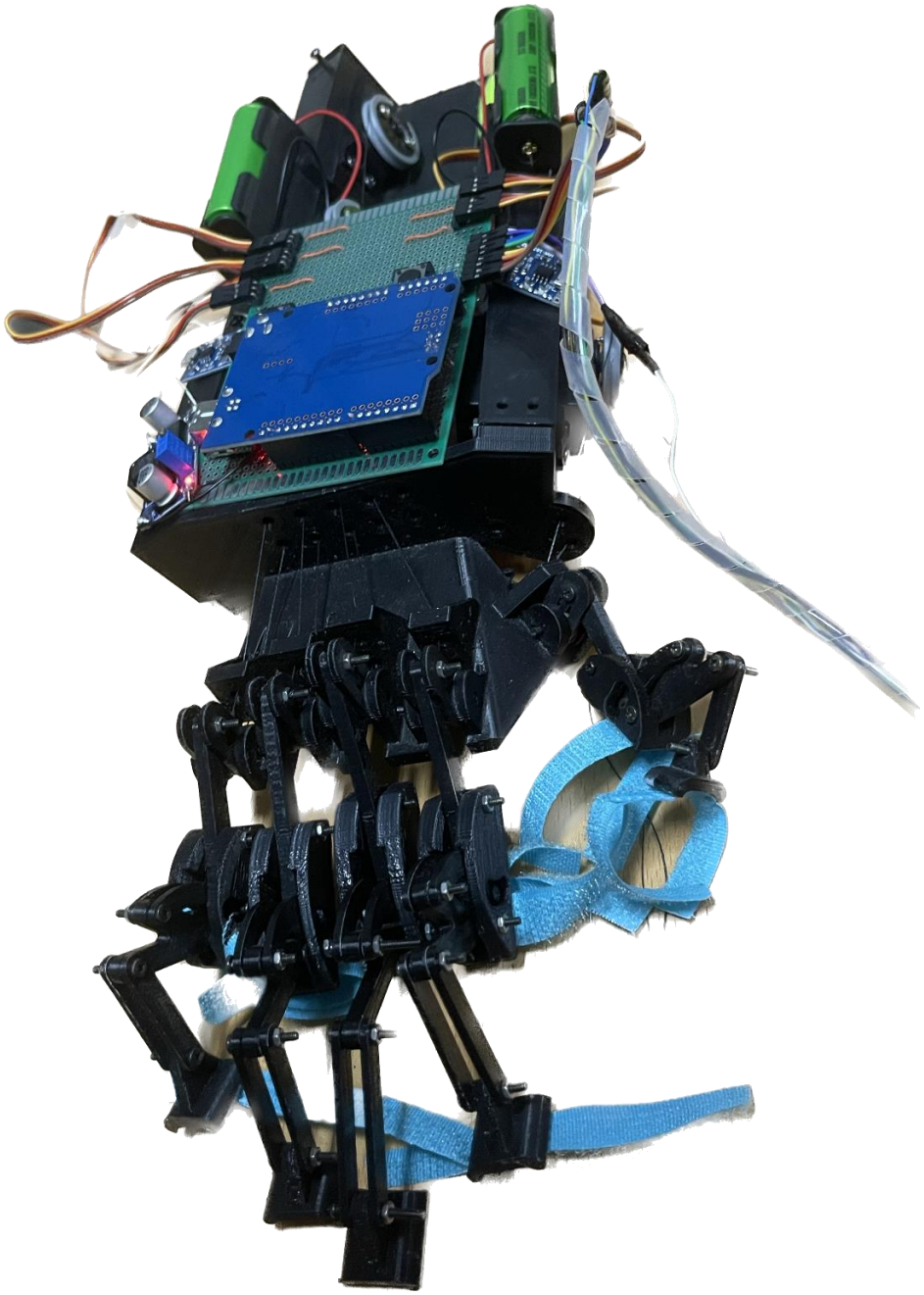


Figure 53 Final Physical Prototype

CHAPTER 5: CONCLUSION AND RECOMMENDATION

Focus of the Project

The project focused on the design and development of a hand exoskeleton for the rehabilitation of motor function in patients with hemiparesis. A literature review was conducted to gauge the present developments in the field and a few exoskeletons were reviewed. Our findings highlighted the different ways to approach this problem and the different applications of hand exoskeletons wherever the need may arise to augment effort.

Incidence of Stroke and Upper Extremity Dysfunction

The incidence of stroke does not show signs of abating, in fact, it has only been rising gradually over the years all over the world. Stroke can have multiple causes, but it does not spare anyone the misery that comes with it. Stroke is a debilitating condition and affects the physical and mental well-being of its patients. An incidence of stroke not only affects the person who suffers it but those around him who are left with the responsibility of their care. It has been reported that most patients with stroke suffer from paralyzed hands. Hands are the most recognizable form of autonomy for humans and performing activities of daily living, greatly depends on hands. Recovering or improving motor function greatly improves the mental well-being of stroke patients.

The Role of Robotics and Design Specifications of the Hand Exoskeleton

Humans have a fascination with robotics and augmenting our natural strengths to superhuman levels as is evident in the existence of many sci-fi TV shows exploring the possibilities of robotics, e.g. Ironman. However, robotics is not limited to fanciful interpretations and for the last few decades, research has gone into exploring possible applications of robotic devices to improve quality of life. The hand exoskeleton designs judged for the present endeavor yielded the following conclusions for implementation in the design process. The hand exoskeleton was decided to be underactuated instead of

fully actuated. Underactuated mechanisms are lighter and more economically viable. Economic viability becomes a concern when the commercialization of the product is considered. The problem inherent in underactuated mechanisms is the non-conformity of the mechanism towards the natural path of the finger. This issue was compensated for by employing a differential evolution algorithm to optimize the dimensions of the links to closely follow the natural path of a human finger. The program that gave the dimensions used equations from the theory of machines and gave the required dimensions.

Synthesis and Analysis

The dimensions so obtained were used to make a 2D model on the software, 'Linkages'. The simulation run on Linkages confirmed the feasibility of the design for the next stage, which involved making a 3D design of the hand exoskeleton. With the 3D design done, the next step required a study to test its structural integrity. Normally static analyses are done on Finite Element Analysis software. The design was reproduced in MSC Adams, where a dynamic analysis was carried out. The exoskeleton had ten joints – two on each finger. The dynamic analysis run on MSC Adams gave the forces on each of the ten joints in the x and y axes for all the possible positions that the mechanism may take during a single periodic motion of its actuators. Instinct would dictate that there must needs arise 10 such instances corresponding to each of the joints, for moments of maximum forces exerted on them. These ten instances are grouped into three instances. The snapshots of the mechanism in those three instances were observed. The snapshot of the mechanism at each of the three critical points was analyzed in COMSOL for static analysis and the results were recorded. The design at this point is deemed sufficient to proceed to development.

Development of Prototype and Potential for Commercialisation

The authors have entered NUST's flagship FICS (Finding Innovative and Creative Solutions) competition for start-ups and has so far been successful in clearing the third stage. The future commercialisation of the project depends on its economic viability.

Commercialization is certainly going to play a good part in making more implementable decisions. The project was limited to being a Final Year Project of mechanical engineering graduates, hence the constraints of budget weighed in the favour of material strong enough to manufacture a prototype. Actual application may require more resilient materials. The general appearance of the prototype is cluttered and chaotic, with wires dangling out of it. It will be in commercial interests to work on its cosmetic character and give it a look that makes it acceptable if not appealing, for use. The implantation of a wireless system to establish a connection between the cyberglove and the hand exoskeleton could also declutter the system and add to its user-friendliness. Such a system is expected to require another Arduino to be installed.

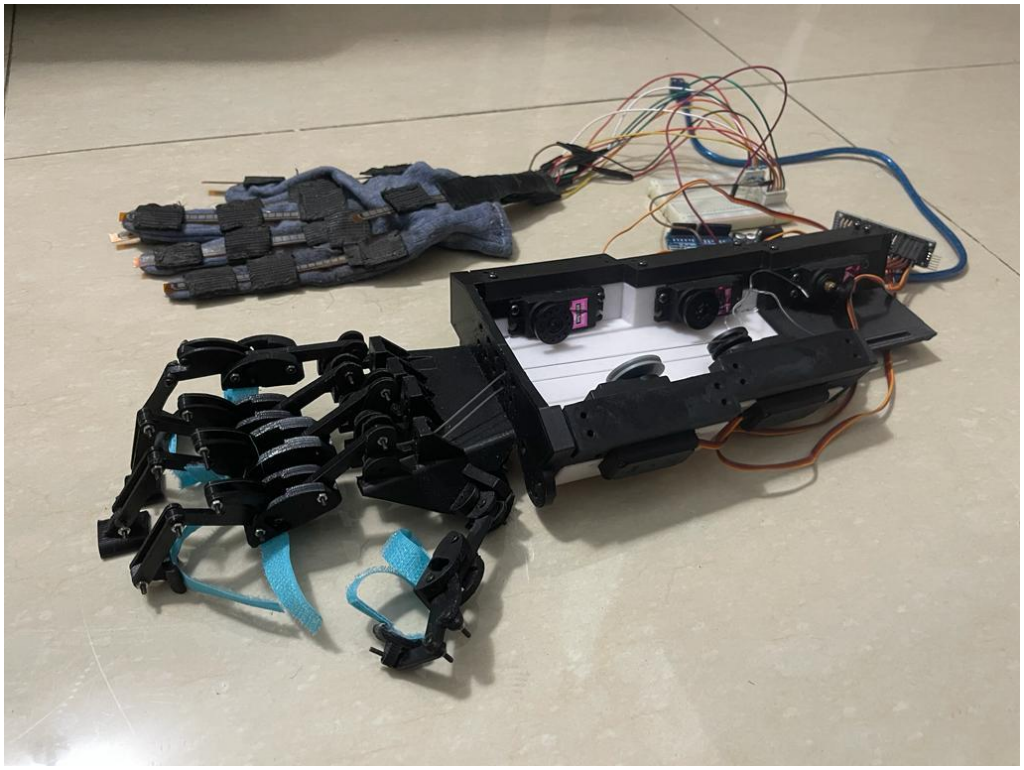


Figure 54 Cyberglove and the Exoskeleton

The figure represents the present state of the prototype. The cyberglove and the hand exoskeleton form an independent and portable system that can be worn.

Challenges faced in designing the Control System

While the mechanism continues to develop, the ways to achieve control are still being explored. The ideas at present include EEG or EMG-based, or motion-based control. EEG or EMG-based control are very attractive options; however, the extraction of the signal is a major issue that comes up. The only viable option left then is to employ motion sensors for motion-based control. The process of deciding how to achieve control is going to be a very testing evaluation of the trade-offs of each of the three methods. While one may seem more viable right now, the actual problem at hand, when the prototype is ready will pave the way for a more decided decision. The present literature review has focused on the mechanisms of hand exoskeletons. Consideration of how the exoskeletons achieve control should warrant another literature review documenting finer details relating to signal extraction and control systems. A potential future project that could stem from this final year project could involve collaboration with hospitals for the deployment of the hand exoskeleton. It should be noted that consent should be obtained from patients to make the individuals involved, aware of all the agreements that they may enter, once they consent to participation in the study involving the clinical trials of the hand exoskeleton.

REFERENCES

[1]. V. Moreno-SanJuan, A. Cisnal, J.C Fraile, J. Pérez-Turiel, E. de-la-Fuente, Design and characterization of a lightweight underactuated RACA hand exoskeleton for neurorehabilitation, Elsevier Robotics and Autonomous Systems (2021)

[2]. An Exoskeleton System for Hand Rehabilitation Driven by ShapeMemory AlloyTe Tang, Dingguo Zhang ∗, Tao Xie, and Xiangyang Zhu

[3]. An EMG-driven Exoskeleton Hand Robotic Training Device on Chronic Stroke Subjects N.S.K. Ho, K.Y. Tong, Senior Member, IEEE, X.L. Hu, K.L. Fung, X.J. Wei, W. Rong, E.A. Susanto

[4]. Development and pilot testing of HEXORR: Hand EXOskeleton Rehabilitation Robot Christopher N Schabowsky, Sasha B Godfrey, Rahsaan J Holley & Peter S Lum

[5]. A. Chiri, F. Giovacchini, N. Vitiello, E. Cattin, S. Roccella, F. Vecchi, M.C. Carrozza,
HANDEXOS: Towards an exoskeleton device for the rehabilitation of the hand, in 2009 IEEE/RSJ Int. Conf. Intell. Robot. Syst, IEEE, 2009,

[6]. C.D. Takahashi, L. Der-Yeghiaian, V.H. Le, S.C. Cramer, A Robotic Device for Hand Motor Therapy after Stroke, in 9th Int. Conf. Rehabil. Robot. 2005. ICORR 2005., in: IEEE, n.d.

[7]. A. T. Laliberté, L. Birglen, C. Gosselin, Underactuation in robotic grasping hands, Mach. Intell. Robot. Control 4 (2002) 1–11,

[8] S. Cobos, M. Ferre, M.A. Sanchez Uran, J. Ortego, C. Pena, Efficient human hand kinematics for manipulation tasks, in 2008 IEEE/RSJ Int. Conf. Intell. Robot. Syst, IEEE, 2008,

[9] J. Nycz, T. Butzer, O. Lambercy, J. Arata, G.S. Fischer, R. Gassert, Design and characterization of a lightweight and fully portable remote actuation system for use with a hand exoskeleton, IEEE Robot. Autom. Lett. 1 (2016)

[10].Material Parameters of VeroWhitePlus RGD835, http://www.stratasys. com/materials/polyjet//media/29592222B80C489BAC28803DB08C10E5

[11] Yang, S.-H.; Koh, C.-L.; Hsu, C.-H.; Chen, P.-C.; Chen, J.-W.; Lan, Y.-H.; Yang, Y.; Lin, Y.-D.; Wu, C.-H.; Liu, H.-K.; et al. An Instrumented Glove-Controlled Portable HandExoskeleton for Bilateral Hand Rehabilitation. Biosensors 2021, 11, 495. https://doi.org/10.3390/bios11120495

[12]. A. Lince et al., "Design and testing of an under-actuated surface EMG-driven hand exoskeleton," 2017 International Conference on Rehabilitation Robotics (ICORR), 2017, pp. 670-675, doi: 10.1109/ICORR.2017.8009325.

[13] Image Reference: https://mikeyjoedonohue.wordpress.com/

[14] K. Aho, P. Harmsen, S. Hatano, J. Marquardsen, V. Smirnov, and T. Strasser, “Cerebrovascular disease in the community: results of a WHO collaborative study,” *Bull. World Health Organ.*, vol. 58, no. 1, p. 113, 1980.

[15] E. L. Miller *et al.*, “Comprehensive overview of nursing and interdisciplinary rehabilitation care of the stroke patient: a scientific statement from the American Heart Association,” *Stroke*, vol. 41, no. 10, pp. 2402–2448, 2010.

[16] D. Zhao *et al.*, “Epidemiological transition of stroke in China: twenty-one–year observational study from the Sino-MONICA-Beijing Project,” *Stroke*, vol. 39, no. 6, pp. 1668–1674, 2008.

[17] G. J. Snoek, M. J. IJzerman, H. J. Hermens, D. Maxwell, and F. Biering-Sorensen, “Survey of the needs of patients with spinal cord injury: impact and priority for improvement in hand function in tetraplegics,” *Spinal Cord*, vol. 42, no. 9, p. 526, 2004.

[18] T. E. Twitchell, “The restoration of motor function following hemiplegia in man,” *Brain*, vol. 74, no. 4, pp. 443–480, 1951.

[19] K. W. Krigger, “Cerebral palsy: an overview.,” *Am. Fam. Physician*, vol. 73, no. 1, 2006.

[20] V. Gracia-Ibáñez, M. Vergara, J. L. Sancho-Bru, M. C. Mora, and C. Piqueras, “Functional range of motion of the hand joints in activities of the International Classification of Functioning, Disability, and Health,” *J. Hand Ther.*, vol. 30, no. 3, pp. 337–347, 2017.

[21] D. Lloyd-Jones *et al.*, “Heart disease and stroke statistics—2009 update: a report from the American Heart Association Statistics Committee and Stroke Statistics Subcommittee,” *Circulation*, vol. 119, no. 3, pp. e21–e181, 2009.

[22] https://www.karger.com/Article/FullText/519554

[23] J. Iqbal, N. G. Tsagarakis, A. E. Fiorilla, and D. G. Caldwell, “A portable rehabilitation device for the hand,” in *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, 2010, pp. 3694–3697.

[24] I. Jo and J. Bae, “Design and control of a wearable and force-controllable hand exoskeleton system,” *Mechatronics*, vol. 41, pp. 90–101, 2017.

[25] T. Tang, D. Zhang, T. Xie, and X. Zhu, “An exoskeleton system for hand rehabilitation driven by shape memory alloy,” in *Robotics and Biomimetics (ROBIO), 2013 IEEE International Conference on*, 2013, pp. 756–761.

[26] F. Zhang, L. Hua, Y. Fu, H. Chen, and S. Wang, “Design and development of a hand exoskeleton for rehabilitation of hand injuries,” *Mech. Mach. Theory*, vol. 73, pp. 103–116, 2014

[27] M. Troncossi, M. Mozaffari Foumashi, C. Mazzotti, D. Zannoli, and V. Parenti-Castelli, “Design and manufacturing of a hand-and-wrist exoskeleton prototype for the

rehabilitation of post-stroke patients," *Quad. DIEM–GMA Atti Della Sesta Giorn. Studio Ettore Funaioli*, pp. 111–120, 2012.

[28] P. Ben-Tzvi, J. Danoff, and Z. Ma, "The Design Evolution of a Sensing and Force-Feedback Exoskeleton Robotic Glove for Hand Rehabilitation Application," *J. Mech. Robot.*, vol. 8, no. 5, p. 051019, 2016.

[29] A. Wege and G. Hommel, "Development and control of a hand exoskeleton for rehabilitation of hand injuries," in *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, 2005, pp. 3046–3051.

[30] J. Arata, K. Ohmoto, R. Gassert, O. Lambercy, H. Fujimoto, and I. Wada, "A new hand exoskeleton device for rehabilitation using a three-layered sliding spring mechanism," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, 2013, pp. 3902–3907

[31] J. Yang, H. Xie, and J. Shi, "A novel motion-coupling design for a jointless tendon-driven finger exoskeleton for rehabilitation," *Mech. Mach. Theory*, vol. 99, pp. 83–102, 2016.

[32] Y. Hasegawa, Y. Mikami, K. Watanabe, and Y. Sankai, "Five-fingered assistive hand with mechanical compliance of human finger," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, 2008, pp. 718–724

[35] V. Gracia-Ibáñez, M. Vergara, J. L. Sancho-Bru, M. C. Mora, and C. Piqueras, "Functional range of motion of the hand joints in activities of the International Classification of Functioning, Disability, and Health," *J. Hand Ther.*, vol. 30, no. 3, pp. 337–347, 2017.

[36] Robert L. Norton – Design of Machinery: An Introduction to the Synthesis and Analysis of Mechanisms and Machines, Fifth Edition

[37] V. L. Roger *et al.*, "Heart disease and stroke statistics—2012 update: a report from the American Heart Association," *Circulation*, vol. 125, no. 1, pp. e2–e220, 2012.

[38] C. D. Takahashi, L. Der-Yeghiaian, V. Le, R. R. Motiwala, and S. C. Cramer, “Robot-based hand motor therapy after stroke,” *Brain*, vol. 131, no. 2, pp. 425–437, Feb. 2008.

[39] V. M. Parker, D. T. Wade, and R. L. Hewer, “Loss of arm function after stroke: measurement, frequency, and recovery,” *Int. Rehabil. Med.*, vol. 8, no. 2, pp. 69–73, 1986

[40] J. H. Carr and R. B. Shepherd, *A motor relearning programme for stroke*. Aspen Pub, 1987.

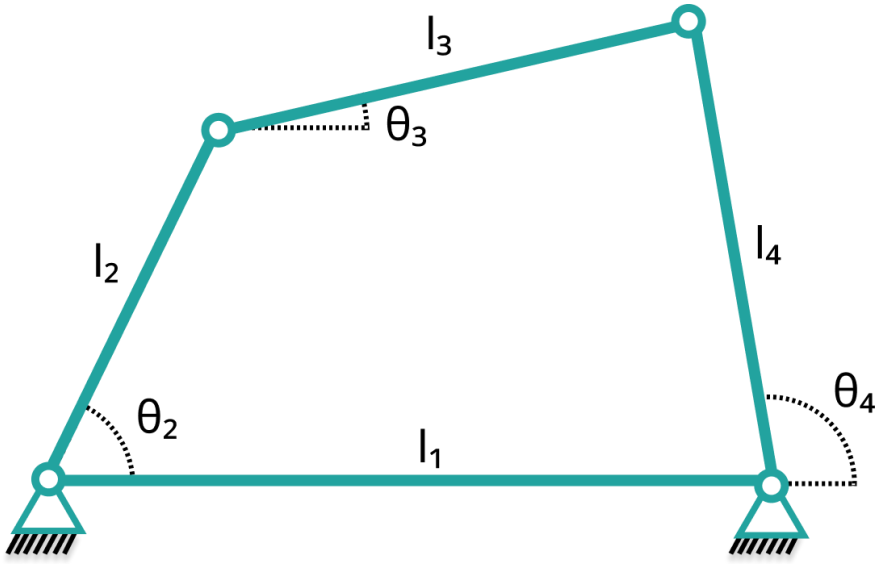
[41] P. Polygerinos, Z. Wang, K. C. Galloway, R. J. Wood, and C. J. Walsh, “Soft robotic glove for combined assistance and at-home rehabilitation,” *Robot. Auton. Syst.*, vol. 73, pp. 135–143, 2015.

[42]https://www.mayoclinic.org/diseases-conditions/stroke/symptoms-causes/syc-20350113#:~:text=Ischemic%20stroke%20occurs%20when%20a,most%20common%20type%20of%20stroke.

[43] Murray Longmore, Ian B. Wilkinson, Andrew Baldwin, Elizabeth Wallin – Oxford Handbook of Clinical Medicine, Ninth Edition

APPENDIX I: FOURBAR MECHANISM EQUATION

Equations from the theory of machines for the analysis of four-bar linkages [36]:



$$K_1 = \frac{l_1}{l_2}, \qquad K_2 = \frac{l_1}{l_4}, \qquad K_3 = \frac{l_2^2 - l_3^2 + l_4^2 + l_1^2}{2 l_2 l_4}$$

$$A = K_1 - K_2 \cos\theta_2 + K_3, \qquad B = -2\sin\theta_2$$

$$C = K_1 - (K_2 + 1)\cos\theta_2 + K_3$$

$$\theta_4 = 2\tan^{-1}\frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$

$$K_4 = \frac{l_1}{l_3}, \qquad K_5 = \frac{l_4^2 - l_1^2 - l_2^2 - l_3^2}{2 l_2 l_3}$$

$$D = \cos\theta_2 - K_1 + K_4\cos\theta_2 + K_5, \quad E = -2\sin\theta_2$$

$$F = K_1 + (K_4 - 1)\cos\theta_2 + K_5$$

$$\theta_3 = 2\tan^{-1}\frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$

Note: The square root of the discriminant is added in the equations for theta shown to find the output angles for the open configuration.

APPENDIX II: CODE FOR DIMENSIONAL SYNTHESIS OF THE MECHANISM

```
import numpy as np
from scipy.optimize import differential_evolution

theta_PH1 = [-2.2, -9.7, -12.9, -17.3, -20, -22.4, -26.1, -29.6, -31.8, -34.5, -36.4, -38.4, -40.4, -42.6, -46.9, -50.7]

def fourbar(parameters, angles, mechNumber, args):
    """
    Compute the joint angles of a four-bar linkage mechanism given a set of parameters and input angles.

    Args:
    - parameters (array): Array of parameters for the four-bar linkage mechanism. The size and meaning of this array
      depend on the value of `mechNumber`.
    - angles (array): Array of input joint angles for the four-bar linkage mechanism.
    - mechNumber (str): String indicating the number of four-bar linkage mechanism. This can be "1", "2", or "3".
    - args (tuple): Tuple of additional arguments required for computing the joint angles. The extracted values from this tuple depend on the value of `mechNumber`.

    Returns:
    - theta3 (array): Array of joint angles for the third joint of the four-bar linkage mechanism.
    - theta4 (array): Array of joint angles for the fourth joint of the four-bar linkage mechanism.
    """

    parameters = np.array(parameters)
    proximal, middle, distal, t1, t2, t3 = args
    if mechNumber == "1":
      try:
        psi1 = parameters[3, :]
        psi2 = parameters[4, :]
        angles = parameters[-16:, :]
        angles = angles + psi1
        l1 = parameters[0, :]
        l2 = parameters[1, :]
        l3 = parameters[2, :]
        l4 = np.divide(t1,np.sin(psi2))
```

```
    except:
      psi1 = parameters[3]
      psi2 = parameters[4]
      angles = parameters[-16:]
      angles = angles + psi1
      l1 = parameters[0]
      l2 = parameters[1]
      l3 = parameters[2]
      l4 = np.divide(t1,np.sin(psi2))
  elif mechNumber == "3":
    try:
      psi1 = parameters[3, :]
      psi2 = parameters[4, :]
      angles = parameters[-16:, :]
      angles = angles + psi1
      l1 = parameters[0, :]
      l2 = parameters[1, :]
      l3 = parameters[2, :]
      l4 = np.divide(t1,np.sin(psi2))
    except:
      psi1 = parameters[3]
      psi2 = parameters[4]
      angles = parameters[-16:]
      angles = angles + psi1
      l1 = parameters[0]
      l2 = parameters[1]
      l3 = parameters[2]
      l4 = np.divide(t1,np.sin(psi2))
  elif mechNumber == "2":
    try:
      phi1 = parameters[3, :]
      phi2 = parameters[4, :]
      angles = np.pi - (angles + phi1 - 0.05)
      l1 = 2.77
      l2 = parameters[0, :]
      l3 = parameters[1, :]
      l4 = parameters[2, :]
    except:
      phi1 = parameters[3]
      phi2 = parameters[4]
      angles = np.pi - (angles + phi1 - 0.05)
      l1 = 2.77
```

```
        l2 = parameters[0]
        l3 = parameters[1]
        l4 = parameters[2]
    K1= np.divide(l1, l2)
    K2= np.divide(l1, l4)
    cosine = np.cos(angles)
    sine = np.sin(angles)
    K3 = np.divide((np.square(l2) - np.square(l3) + np.square(l4) + np.square(l1)),(2 * np.multiply(l2,
l4)))
    A1 = cosine-K1-(np.multiply(K2, cosine))+K3
    B1 = -2*sine
    C1 = K1 - np.multiply((K2+1),cosine) + K3
    theta4 = 2 * np.arctan(np.divide(-B1-np.emath.sqrt(np.square(B1) - (4*np.multiply(A1,C1))) ,
(2*A1)))
    K4 = np.divide(l1,l3)
    K5 = np.divide((np.square(l4) - np.square(l1) - np.square(l2) - np.square(l3)),(2 * np.multiply(l2,
l3)))
    D1 = cosine-K1+(np.multiply(K4,cosine))+K5
    E1 = -2*sine
    F1 = K1 + np.multiply((K4-1),cosine) + K5
    theta3 = 2 * np.arctan(np.divide((-E1-np.emath.sqrt(np.square(E1) - (4*np.multiply(D1,F1)))),
(2*D1)))
    return theta3, theta4

def alphaError(alpha, proximal, middle, distal, para):
    x1 = proximal*np.cos(alpha)
    y1 = proximal*np.sin(alpha)
    Fx = np.array([[9.003, 8.233, 7.775, 7.057, 6.5, 5.254, 4.363, 3.139, 1.839, 1.301, 0.0734, 0.0417,
0.0167, -0.054, -0.0349, -0.0545]]).transpose()
    Fy = np.array([[-0.079, -3.178, -3.924, -4.76, -5.26, -5.798, -6.151, -6.09, -5.529, -5.254, -4.706, -
4.302, -4.011, -3.646, -3.271, -2.957]]).transpose()
    theta_PH1 = np.array([[-2.2, -9.7, -12.9, -17.3, -20, -22.4, -26.1, -29.6, -31.8, -34.5, -36.4, -38.4, -
40.4, -42.6, -46.9, -50.7]]).transpose()
    theta_PH2 = np.array([[-9.6, -30.2, -37.5, -46.1, -51.8, -65.5, -73.7, -87.4, -103.3, -111.6, -121.5, -
128.5, -136, -141.6, -151.6, -159.9]]).transpose()
    theta_PH3 = np.array([[-6.9, -39.7, -48.8, -62.2, -70.4, -90.6, -103.2, -122, -148.4, -160.4, -175.5, -
186.2, -194.2, -202.7, -213.5, -221.5]]).transpose()
    theta_PH2_rad = theta_PH2 * (np.pi/180)
    theta_PH3_rad = theta_PH3 * (np.pi/180)
    Fx = Fx - np.cos(theta_PH3_rad)*distal - np.cos(theta_PH2_rad)*middle
    Fy = Fy - np.sin(theta_PH3_rad)*distal - np.sin(theta_PH2_rad)*middle
    E1 = np.square(x1-Fx) + np.square(y1-Fy)
```

```
    E2 = np.square(alpha - ((theta_PH1*np.pi)/180))
    M1 = 10
    M2 = 1000000
    cost = M1*np.sum(E1, axis=0)+ M2*np.sum(E2, axis=0)
    try:
      ordered = lambda x: (np.diff(x, axis=0)<=0).all(axis=0)
      array = ordered(para[-16:,:])
      cost[np.where(~array)] += 1000000
      return cost
    except:
      ordered = lambda x: (np.diff(x, axis=0)<=0).all(axis=0)
      array = ordered(para[-16:])
      cost[np.where(~array)] += 1000000
      return np.sum(cost)

def betaError(beta, proximal, middle, distal, para):
    alpha = np.array([[-0.03434395, -0.16856848, -0.2241437,  -0.30062929, -0.34750948, -
0.38938777, -0.45407241, -0.51420633, -0.55167307, -0.59847426, -0.63355711, -0.66782388, -
0.70216166, -0.73971795, -0.81372956, -0.87926912]]).transpose()
    x1 = proximal*np.cos(alpha) + np.cos(alpha+beta)*middle
    y1 = proximal*np.sin(alpha) + np.sin(alpha+beta)*middle
    Fx = np.array([[9.003, 8.233, 7.775, 7.057, 6.5, 5.254, 4.363, 3.139, 1.839, 1.301, 0.0734, 0.0417,
0.0167, -0.054, -0.0349, -0.0545]]).transpose()
    Fy = np.array([[-0.079, -3.178, -3.924, -4.76, -5.26, -5.798, -6.151, -6.09, -5.529, -5.254, -4.706, -
4.302, -4.011, -3.646, -3.271, -2.957]]).transpose()
    theta_PH1 = np.array([[-2.2, -9.7, -12.9, -17.3, -20, -22.4, -26.1, -29.6, -31.8, -34.5, -36.4, -38.4, -
40.4, -42.6, -46.9, -50.7]]).transpose()
    theta_PH2 = np.array([[-9.6, -30.2, -37.5, -46.1, -51.8, -65.5, -73.7, -87.4, -103.3, -111.6, -121.5, -
128.5, -136, -141.6, -151.6, -159.9]]).transpose()
    theta_PH3 = np.array([[-6.9, -39.7, -48.8, -62.2, -70.4, -90.6, -103.2, -122, -148.4, -160.4, -175.5, -
186.2, -194.2, -202.7, -213.5, -221.5]]).transpose()
    theta_PH1_rad = theta_PH1 * (np.pi/180)
    theta_PH2_rad = theta_PH2 * (np.pi/180)
    theta_PH3_rad = theta_PH3 * (np.pi/180)
    Fx = Fx - np.cos(theta_PH3_rad)*distal
    Fy = Fy - np.sin(theta_PH3_rad)*distal
    E1 = np.square(x1-Fx) + np.square(y1-Fy)
    E2 = np.square((beta) - (theta_PH2_rad-theta_PH1_rad))
    M1 = 100
    M2 = 100000
    cost = M1*np.sum(E1, axis=0)+ M2*np.sum(E2, axis=0)
    try:
```

```
        ordered = lambda x: (np.diff(x, axis=0)<=0).all(axis=0)
        array = ordered(para[-16:,:])
        cost[np.where(~array)] += 1000000
        return cost
    except:
        ordered = lambda x: (np.diff(x, axis=0)<=0).all(axis=0)
        array = ordered(para[-16:])
        cost[np.where(~array)] += 1000000
        return np.sum(cost)

def functionGeneratorError(theta4, output_angles_2, para):
    try:
        theta4_ = theta4 - para[4, :]
    except:
        theta4_ = theta4 - para[4]
    E2 = np.square(output_angles_2 - theta4_)
    M2 = (180/np.pi)**2
    cost = M2*np.sum(E2, axis=0)
    try:
        return cost
    except:
        return np.sum(cost)

def functionGeneratorObjectiveFunction(para, *args):
    x, x1 = args
    output_angles_2 = x1["x"][-16:]
    theta3, theta4 = fourbar(x["x"], x["x"][-16:], "1", args=(5.27, 2.86, 1.844, 1.75, 1.75, 1.75))
    input_angles_2 = theta3 + x["x"][4]
    try:
        phi1 = para[3, :]
        phi2 = para[4, :]
        angles = np.zeros((16, para.shape[1]))
    except:
        phi1 = para[3]
        phi2 = para[4]
        angles = np.zeros((16))
    theta3, theta4 = fourbar2(para, np.array([input_angles_2]).transpose(), "2", (x, x1))
    cost = error2(theta4, np.array([output_angles_2]).transpose(), para)
    try:
        cost[np.iscomplex(cost)]*=1000000
    except:
        if np.iscomplex(cost):
```

```
        cost *= 1000000
      if para.shape == (20,):
        np.sum(cost)
    return abs(cost)

def pathGeneratorObjectiveFunction(para, *args):
    proximal, middle, distal, t1, t2, t3 = args
    try:
      psi1 = para[3, :]
      psi2 = para[4, :]
      angles = np.zeros((16, para.shape[1]))
    except:
      psi1 = para[3]
      psi2 = para[4]
      angles = np.zeros((16))
    theta3, theta4 = fourbar(para, angles, "1", (proximal, middle, distal, t1, t2, t3)) # Modify the
mechanism number to correspond to the mechanism being optimized
    theta4_ = np.pi - theta4
    alpha = np.pi - (psi1 + psi2 + theta4_)
    cost = error(alpha, proximal, middle, distal, para) # Modify the name of the error function to
corresponding to the mechanism being optimized
    try:
      cost[np.iscomplex(cost)]*=1000000
    except:
      if np.iscomplex(cost):
        cost *= 1000000
    if para.shape == (21,):
      np.sum(cost)
    return abs(cost)

bounds = [(2,4), (1,4), (2, 5), (np.pi/6, np.pi/2),(np.pi/6, np.pi/2),(0, np.pi/2),(0, np.pi/2),(0, np.pi/2),(0,
np.pi/2),(0, np.pi/2),(0, np.pi/2),(0, np.pi/2),(0, np.pi/2),(0, np.pi/2),(0, np.pi/2),(0, np.pi/2),(0,
np.pi/2),(0, np.pi/2),(0, np.pi/2),(0, np.pi/2),(0, np.pi/2)]

x = differential_evolution(function, bounds, args = (5.27, 2.86, 1.844, 1.75, 1.75, 1.75), popsize=100,
mutation=(0, 0.4), recombination=0.5, maxiter=400000, vectorized=True, disp=True) # change the
function passed to the differential evolution algorithm to correspond to the type of mechanism being
optimized.
print(x)
output_angles_2 = x["x"][-16:]
theta3, theta4 = fourbar(x["x"], x["x"][-16:], "1", args=(5.27, 2.86, 1.844, 1.75, 1.75, 1.75))
psi1 = x["x"][3]
```

```
psi2 = x["x"][4]
theta4_ = np.pi - theta4
alpha = np.pi - (psi1 + psi2 + theta4_)
print("Angle Alpha generated:")
print(alpha*180/np.pi)
print("Error in angle Alpha in degrees:")
print(theta_PH1 - (alpha*180/np.pi))
theta_PH2 = [-9.6, -30.2, -37.5, -46.1, -51.8, -65.5, -73.7, -87.4, -103.3, -111.6, -121.5, -128.5, -136, -141.6, -151.6, -159.9]
theta3, theta4 = fourbar(x["x"], x["x"][-16:], "3", args=(5.27, 2.86, 1.844, 1.75, 1.75, 1.75))
psi1 = x["x"][3]
psi2 = x["x"][4]
theta4_ = np.pi - theta4
beta = np.pi - (psi1 + psi2 + theta4_)
print("Angle beta generated:")
print(beta*180/np.pi)
print("Error in angle beta in degrees:")
print(((beta+alpha)*180/np.pi)-theta_PH2)
```