Design of a low cost robotic system to aid in the rehabilitation of stroke patients

Matthew Bowers, Nathaniel Goldfarb, Aishwary Jagetia, Rishi Khajuriwala,

Akshay Kumar, Brandon Lam, and Nishant Shah

December 12, 2017

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

A stroke is a cerebrovascular accident (CVA) or the damage to the central nervous system (CNS) caused by the sudden disturbance in the blood supply to the brain tissue which leads to partial loss of brain function or impaired motor control. It is not a disease but a clinical syndrome characterized by the rapid onset of focal neurological signs, lasting more than 24 hours or eventually leading to death of the patient with presumed vascular cause. It can be caused by a few different pathologies which all result in a sudden onset focal cerebral damage. It is the third most common cause of death worldwide, second leading cause of dementia and the major cause of depression in the elderly [16].

1.2 Types of Strokes

Strokes can be classified in two types Ischaemic stroke and Haemorrhagic stroke.

- Ischaemic strokes makes up 80% of strokes and are primarily due to caused due to cerebral Ischaemic, which is a restriction in blood supply to tissues, causing a shortage of oxygen and glucose needed for cellular metabolism. The most common causes are large artery disease, small vessel disease and cardioembolism. However, a large number of other diseases can occasionally cause Ischaemic stroke too.

- Haemorrhagic Stroke makes up 20% of strokes is typically caused by Subarachnoid haemorrhage, the spontaneous extravasation of blood into the subarachnoid space when a blood vessel near the surface of brain leaks.

1.3 Risk Factors

There are many causes which may lead to a stroke, the most common are: age, gender, blood pressure, smoking, diabetes mellitus, cholesterol, body mass index and physical exercises, alcohol, ethnicity, homocysteine.The chance of a person having a stroke rises exponentially with age and about 25 % of men and 20% of women who live to 85 years of age can expect to suffer a stroke. However, 25% of strokes occur in individuals of working age [16].

1.4 Effects of a Stroke

A stroke may damage our body parts which may consequently affect some of the body, its functions and activities we perform in our day to day life as listed below:

- Body Parts most affected by Stoke:
  - Brain
  - Cardiovascular system
  - Legs and arms
  - Shoulder
- Body functions most affected by Stroke:
  - Temperament and personality
  - Energy and drive
  - Sleep, attention, and memory
  - Psychomotor and perceptual
  - Cognitive and seeing
  - Proprioception and touch
  - Voice and articulation
  - Ingestion, defecation, urinary, and sexual
  - Mobility and stability of joints
  - Muscle power, tone, and reflexes
  - Muscle endurance
  - Control of involuntary movement
  - Gait pattern functions
  - Consciousness orientation and intellectual
- Daily Activities affected by Stroke:
  - Reading, writing, and calculating
  - Solving problems
  - Undertake single and multiple tasks

- Transferring oneself
- Maintaining body position
- Mobility
- Toileting
- Dressing
- Washing and self-care
- Hand and arm use
- Eating and drinking
- Preparation of meals
- Recreation and leisure

1.5 Rehabilitation

Rehabilitation is a therapy or the process of restoring and regaining physical strength and function after being affected by a Stroke. The hope is that through physical exercise a stroke patient can restore some of the lost or weakened function. This is a long process that shows the best results when done within the first six month of the stroke, Figure 1.1 shows a timeline of the stages of rehabilitation. The rehabilitation process consists of several steps that are listed below.

1. Assessment to identify and quantify the patient's needs
2. Goal setting to define realistic and attainable goals for improvement
3. Intervention, to assist in the achievement of goals
4. Reassessment to assess progress against agreed goals

   The stroke rehabilitation can be broken down in three stages:

1. Acute stage:(1st Week) This is a time for convalescence. This is usually done in the hospital. Therapists will typically focus on helping the patient what patients can do.
2. Subacute phase: (2nd Week to 3 Months) Most recovery will take place during this stage. The brain is "primed" to recover.
3. Chronic phase: (3 months and beyond) This is the stage where the patients have to perform the activities by themselves. This phase of recovery is the most difficult stage the patient will face but once the threshold is crossed there will be important gains.

Because of the brain's amazing ability to rewrite itself, essential progress can be made during the stroke rehabilitation.

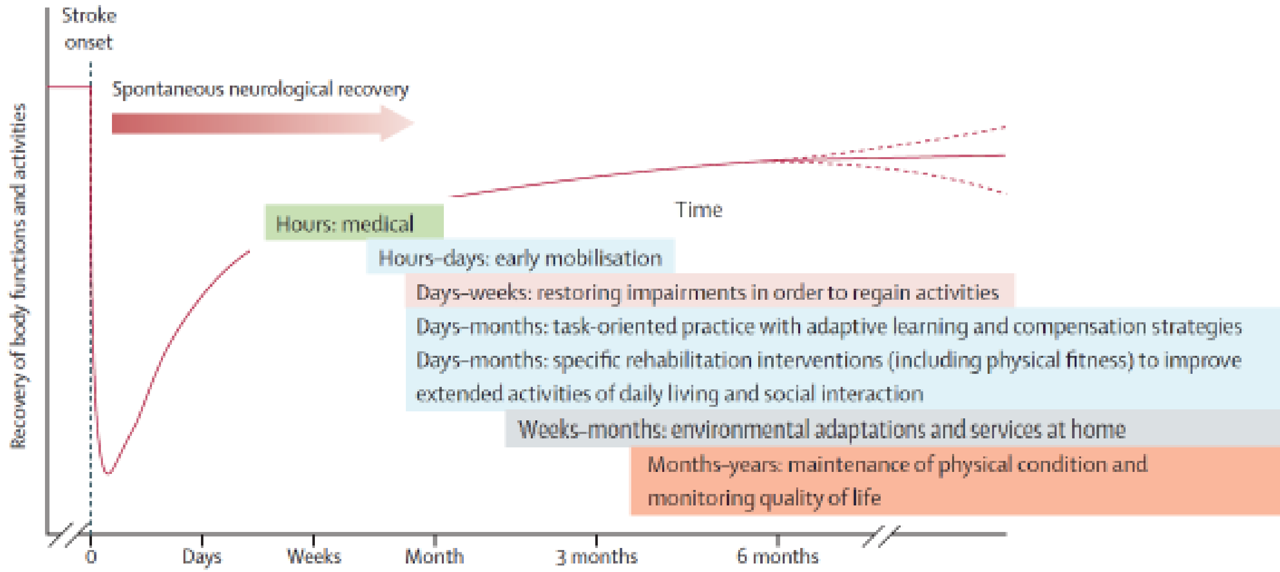


Figure 1.1: Stages of Rehabilitation

1.5.1 Traditional Treatments of Stroke

Many different methods have been explored to aid in the of recovery of stroke patients. These treatments range from traditional physical therapy to the current methods that involve the use of high tech robotic platforms. Even with recent advancements the most common treatment is still physical exercise. This method of treatment requires the least amount of equipment, but often requires the physician or therapist to help the patient through different exercise during multiple physical therapy sessions. In addition to placing a large cost on the physician's time and physical/mental energy, this treatment method often requires the patients to travel to a clinic over and over to receive personalized treatment. This leads to a missed opportunity with dangerous consequences. Studies have shown that when treatment is received within the first six month the patient has the greatest chance of recovery. This created a need to make treatment faster and easier to provide. Researches have developed different methods to help ease the burden on the therapist and increase the patient's physical exercise within those first 6 months. These exercises often include stretching and holding the stretch, small motor functions that physical feedback, active-passive bilateral therapy, or some kind of robotic assisted method [25].

1.5.2 Exercises for upper body rehabilitation

After stroke, the joints affected in the upper limb are the shoulder, the elbow and the wrist joints. Rehabilitation typically starts by straightening the arm, curling the arm, and performing some motion with it. While these motion seem simple, they are very difficult for a person after they have been affected by stroke. Usually the sequence for the recovery of joints is 1) the shoulder, 2) the elbow, and 3) the Wrist.

The rehabilitation for shoulder joint starts after the uncurling of the hand. Shoulder flexion and extension is the first exercise. It is performed by keeping the patient's elbow straight, their whole arm starts in front of them is moved slowly upwards above their head, then back down as shown in Figure 1.2. Shoulder abduction is the focus of the next exercise, in it the patient's arm starts at the side of the body and is moved upwards slowly, once the arm is above shoulder height it should be made sure that the palm is facing up. Then the patient's arm is brought above their head and then lowered as shown in Figure 1.3.

Figure 1.2: Shoulder Flexion and Extension

Figure 1.3: Shoulder Abduction

For Elbow joint, the exercises focus on elbow abduction as well as flexion and extension. For elbow abduction, the elbow joint is given support and slowly the arm is moved inwards and outwards in a direction perpendicular to the body with the thumb pointed upwards. For elbow flexion and extension, the patient's elbow is slowly bent until their hand is touching their shoulder, then the arm is fully extended and straighten the arm down slowly, as seen in Figure 1.4.

After some recovery, there are exercises which helps both the joints called Proprioceptive Neuromuscular Facilitation. There are two exercises that are useful for stroke patients.

In the first exercise the patient moves their right arm from their lower left side to the patient's right shoulder. The second exercise requires the user to take their right arm from lower right side of their body to their left shoulder. These exercises are good for both the elbow and shoulder joints. The best exercise for the patients to use is to move their arm in

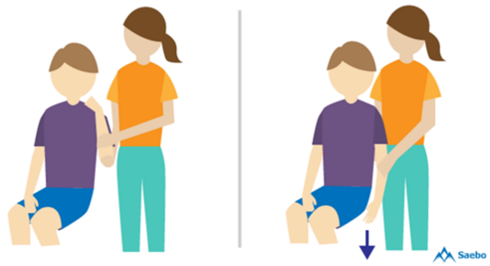


Figure 1.4: Elbow Flexion and Extension

circular motion, but this should only be done only after some recovery is done. An example of these exercise can be seen in Figure 1.5, the patients are asked to move objects from the strong side of the body to the weaker side, this exercise helps patients for vision recovery and joint recovery.

Figure 1.5: Exercise 1

1.5.3 Robot-assisted Rehabilitation

Rehabilitation therapists spend a large amount of time on conventional techniques, which are inconsistent due to therapist fatigue and other human factors. Robotic-assistance is a more consistent approach due to programmable robotic devices, reproducable force outputs and replicable training provided by the therapist. Also, the robotic-assistive device provides patients the opportunity to perform the rehabilitation tasks independently with minimal supervision in an environment of their choosing, typically will be their homes. Thus, these manipulations may ultimately enhance the speed recovery of the patients beyond current possibilities using the robotic-assistance. This type of treatment has also been more efficient

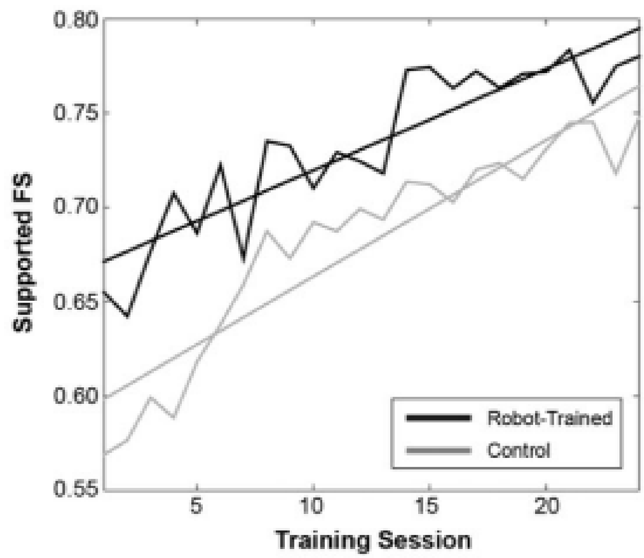


Figure 1.6: Robot-assisted Rehabilitation

for therapists as mitigates human factors due to fatigue, unpredicted actions, and other factors.

During the ARM Guide study [6], subject completes a task either using traditional rehabilitation methods or robotic assistive rehabilitation methods. The results are then compared and displayed in Figure 1.6 and Figure 1.7. This figure shows that the subjects who received robotic assistance improved much more quickly than those who did not get assistance.

Similarly, studies shows that the patients who received robotics therapy had a reduced impairment compared to the conventional therapy. Some works also indicate that supplemental robotics therapy can improve recovery in acute and chronic stroke patients. While robot assistance is shown to increase the rate of recovery it does have limitations, such as potentially allowing the user to slack during their exercises. These limitations must be taken into consideration before developing a robotic system.

1.5.4 Robotic Therapy Development

The main motive of a therapy is to increase the patient's engagement in rehabilitation activities. There are three different strategies that could be implemented:

- Assist-as-needed: The rehabilitation activities are predefined and thus the end effector follows a specific desired trajectory chosen by the therapist. Various modeling and estimation methods are used to determine the appropriate amount of support required as well as machine learning techniques can be used to adjust task difficulty in order to overcome problems like slacking.

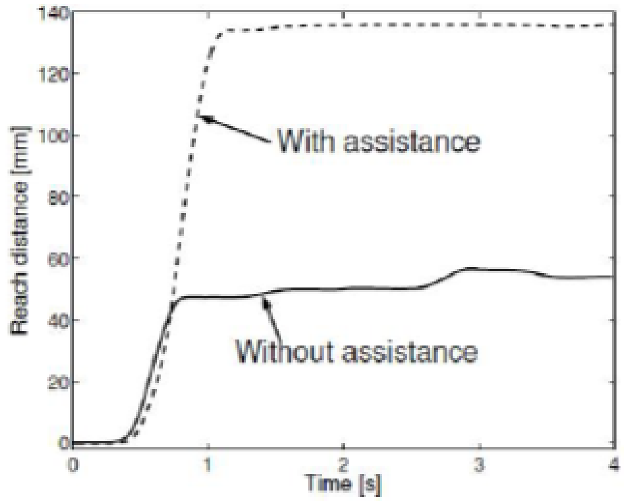


Figure 1.7: Robot assistance vs Without assistance

- Detection of patient intent to move: This type of technique is particularly used for impaired patients who have trouble in completing movements. With the help of haptic devices which can determine the Forces, Velocity, Time threshold, EMG and EEG signals we can trigger the robot assistance.
- Virtual reality games for a more immersive experience

Robotic-assistance devices are extremely useful in the field of rehabilitation. Most of these devices often fall under 2 different categories of either an end-effector device or an exoskeletons (Figure 1.8)

While both of these devices have been shown to improve a patient's ability to recover from their injuries there are many factors that distinguish the two form each other. Below lists some of the advantages and disadvantages of using either an end-effector device or an exoskeleton:

- End-effector:
  - Simpler structure/ control
  - Easy to adjust to patient
  - Limb posture not fully determined
  - Limited force / position data
  - Risk of joint injury
- Exoskeleton:

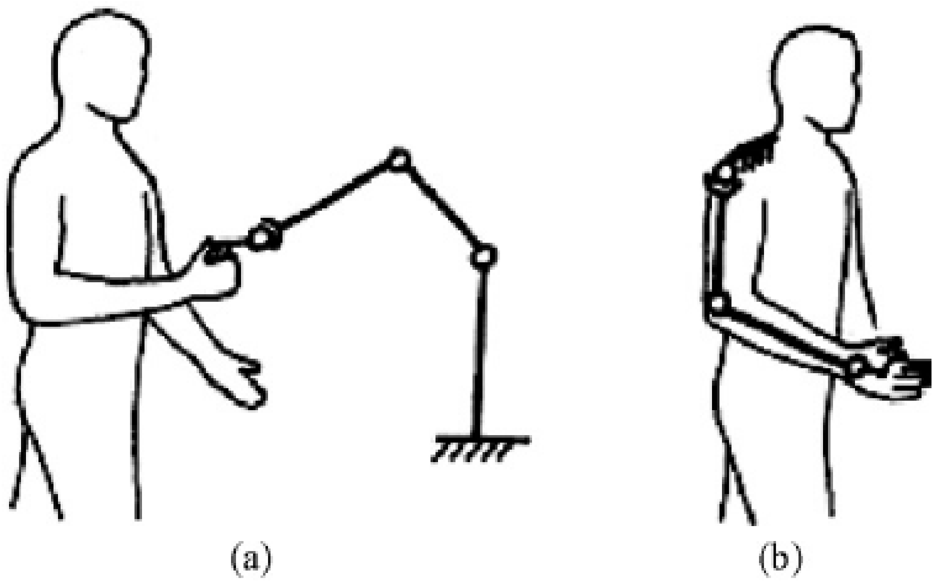


Figure 1.8: End-effector devices(a) Vs Exoskeletons(b)

- Joint axes fully determined
- Physiological movements
- Force and position data of each joint
- Robot axes have to align with anatomical axes
- Longer setup time
- Challenging anatomical constraints

End-effector devices can be further classified as either being unilateral Vs bilateral. An example of both of these can be seen in Figure 1.9.

Exoskeletons can be further classified as Partial or Full (a)(b), Wire-based (c), Single or Multi robot (d), as shown in Figure 1.10.

1.5.5 Comparison of Robot-assisted Therapy and Conventional Therapy

Conventional therapy has been the most commonly used method of treatment, even though studies have shown that robot-assisted therapy can greatly improve the speed of recovery. Below in Table 1.1 shows a comparison of the 2 different types of therapies.

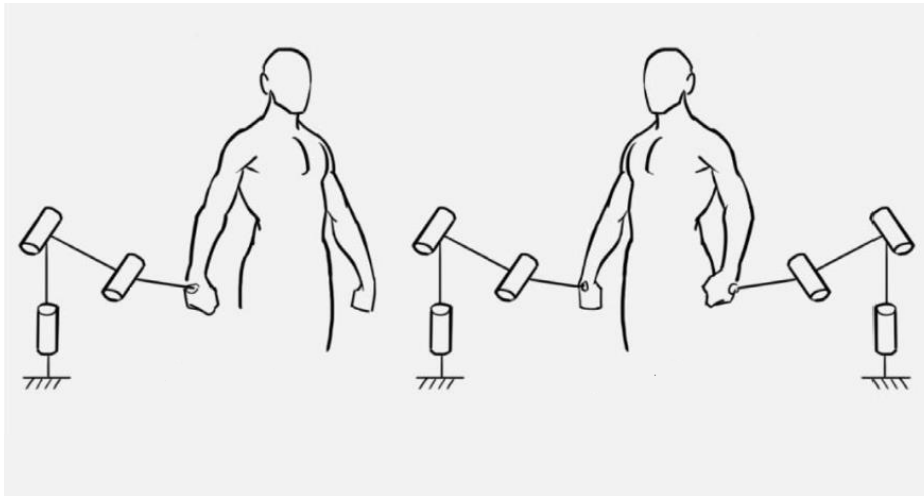


Figure 1.9: Unilateral vs Bilateral

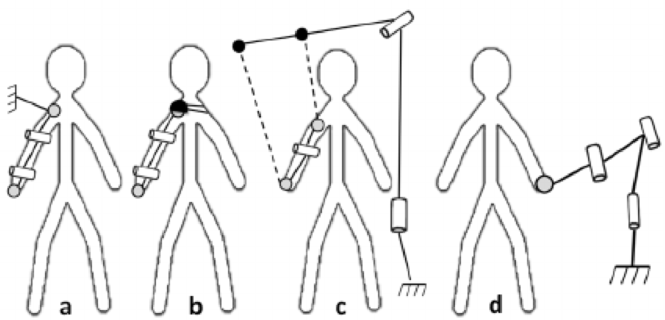


Figure 1.10: Partial(a) or Full(b), Wire-based (c), Single or Multi robot(d)

1.5.6 Integrating Video Games in Stroke Rehabilitation

Convention rehabilitation is very time consuming and labor and resource intensive. It is not available everywhere as one needs a good doctor to help with the therapy. Effects are delayed and very minimal in some of the patients as they get bored and avoid doing the exercises as suggested by the doctor. Also, the patients are needed to go every time for session with the doctor, so they need to go to special centers for the rehabilitation.[32] One method that has provided much success is the use of games to help assist in the recovery of the patient. Most of these methods place the patient in a virtual environment by having them control a character or complete some task. These are often through video game consoles such as an Xbox Kinect, the Wii, or other commonly used systems. These make the solutions more user friendly as well as affordable for many patients. Studies have shown that patient's recovery has a strong correlation with the inclusion of games in the recovery treatment [29],[23], [14]. Some of these treatments include an assistive tool to help track the patient's motions. Video game rehabilitation provides a natural or real-life environment, patients can forget about

	Conventional Therapy	•Robot Assisted Therapy
Advantages	•Less costly •No big space requirements •No risk of obsolescence •Treatment of any body part •High degree of feedback or flexibility •Communication dependent on therapist	•Supports therapist •High reliability •High amounts of repetition/intensity •Individually adjustable assistance •Quantifiable and objective assessments •Highly motivating for some •Detailed and timed sensory feedback •New interventions
Limitations	•Limited by manpower •Poor movement repeatability •Limited number of repetitions •Difficult for severely affected patients •Imprecise or subjective assessments •Motivation dependent on patient therapist relationship •Subjective feedback	•Costly •Space consuming •Risk of obsolescence •Devices specific for individual limbs •Lacks degree of feedback or flexibility •Limited communication

Table 1.1: Therapy Comparison

their surroundings and focus directly on the task in the simulated environments. It provides more motivation to the patients, as they play games and tend to have fun, so they are more motivated to continue therapy. Rehabilitation services can be provided by a doctor from a distance. A major obstacle to recovery post discharge is for patients to continue with home exercises and therapy sessions, this can be achieved by using video game rehabilitation as to tend to motivate patients to do the exercises. Video game rehabilitation also has some challenges as the cost of all the consoles and systems used are very expensive and the operation usually requires a technical expertise.[33] Many games are developed for upper stroke rehabilitation taking into consideration factors such as social context, motion type, cognitive challenge and many others. Many games such as Frog Simon, Under the Sea, Dirt Race, Baseball Catch, Catch the Kitty, Pong, Frogger, Helicopter are developed, tested and improved. [34]

1.5.7 Recovery Measurement Techniques

Physicians have used many different methods of determining a patient's severity of stroke as well as their recovery. Some of these methods include the NIH Stroke Scale, the Fugl-Meyer Assessment of Motor Recovery, the Barthel Index of Activities of Daily Living, The Physical Function Index of the SF-36, and the Modified Rankin Outcome Scale [quality of life among stroke survivors, management of adult stroke, early stroke treatment associated with better outcome].

The NIH Stroke scale is a that is used to determine how severe a stroke is. It does this by evaluating the effects of acute cerebral infarction on the levels of consciousness, language, neglect, visual-field loss, extraocular movement, motor strength, ataxis, dysarthria, and sensory loss.A trained physician, or therapist is able to duct these tests from the patient's bedside and is meant to be a simple reliable to tool that can give therapists a gauge on how their patient is doing [11].

The Fugl-Meyer Assessment of Motor Recovery is targeted towards stroke survivors who have been effected on one side of an upper limb. It can be used to both evaluate and measure the patient's recovery. It tests and scores the patients on 5 different domains: motor function, sensory function, balance, joint range of motion, and joint pain. It tests 226 items and requires a few items to successfully conduct the test [2].

The Barthel Index of Activities of Daily Living tests a patient on how they are able to conduct activities deemed necessary for everyday living. These categories include: feeding, bathing, grooming, dressing, bowel control, bladder control, toilet use, transfers, mobility on level surfaces, and their use of stairs. The patient is ranked in these areas and are added compiled into a single score. Many of these rankings rank if the patient is able to carry out the task independently or if they require any kind of assistance and are meant to be the patient's self report or as observed by another person [9].

The Physical Function Index of the SF-36 is 36 item report/survey of patient health, used for evaluating individual patients health status. In addition it is also useful for researching the cost-effectiveness of a treatment and monitoring and comparing disease burden. This is a shortened version of the SF-60 which includes 60 items to test. Scoring is done in the areas of stability, physical functioning, bodily pain, general health perceptions, physical role functioning, emotional role functioning, social role functioning, and mental health [28].

The Modified Rankine Outcome Scale ranks a patient off a simple 0-6 scale. These rankings are defined as from having no symptoms to being dead. Between these 2 rankings are a range of disability the patient may experience [27]. These assessments offer a wide range of categories that therapists and physicians check when a patient is unwell. While some of these categories are not directly affected by a stroke they do offer insight into what physicians and therapists consider important for daily functions and what a normal person should be able to do.

As mentioned in Section 1.4.2, robot-assisted rehabilitation has greatly decreased the patient's time of recovery and allows for a more targeted recovery. The previously mentioned recovery measurement techniques can be used to evaluate the patient's current state of recovery whether they are using a traditional recovery method such as stretching or with a robotic-assisted method.

Chapter 2

Related Work

Several other researches have investigated this topic. Many of the systems that have been developed are similar. They all aim to assist a stroke patient through some movements. Each system accomplishes this in their own way. A brief overview of the main systems that are currently being studied or are in current use will be discussed below. This field is not yet complete, there are still problems that have to be asked and problems to be solved. The overarching challenge is to solve the automation movement therapy problem. This motivation of this project is to find the optimal balance between what the user and robot should be doing [5 pg. 1693].

2.1 MIT-MANUS

One of the earliest projects that looked into a solution to this problem was conducted by Hogan et al at MIT with the development of the MIT-MANUS. Between 1994 and 2004, they have treated 250 patients on there system [12]. Over the years the system has undergone many improvements and refinements. The overarching goal of was to investigate if brain damage can be repaired through task space training.

The MANUS robot is a SCARA manipulator that allows for 2 degrees of motion in the horizontal plane. There is then an wrist support mechanism attached to the end of the arm of the robot. This mechanism has an additional three degrees of motion. Meaning in total the MANUS is a 5 DoF robotics system. The MANUS can be fixed to a desk, these allows the height of the system be adjusted to the user [10].

The system was designed so that it can be backdrivable, these is so that a user can move the system. This design feature how every contradicts another necessary, that the motors can produce enough torque to move the the user limbs. The arm is capable of delivering 45N of force to the the user. The joints are powered by motors rated for 9.65Nm. This positions and velocities of the links are measured with a 16-bit absolute encoder and a 1.8V/rad/sec tachometer. A virtual impedance of 4.2N/mm was setup, this is the smallest amount of impedance that can be sensed by a person. The workspace of the robot was 15"x18" and worked the shoulder abduction between the range of 45 and 65, It also worked the flexion of the shoulder between 30 and 90.

To add functionality to the system a vertical actuator was added in MANUS. The vertical actuator was constituted of a lead screw powered by a brushless motor. To engage the patients in the system the MIT-MANUS is used to play a video game. The patient as to move around a o cursor on the screen to different location. This makes the the rehabilitation process more interactive [13]. The robotic system is then treated as an input to the computer. Where the sensors find the position and the motors are used to provide assistants to the patient to achieve the trajectories.

The MANUS device recently underwent a rigorous study that compared traditional rehabilitation therapy and that of MANUS therapy. The conclusion of this therapy was the the patients who underwent the therapy with the robot showed a more improvement than the people who were treated in the traditional way. Additional, the robot assisted therapy could be provided at a lower cost than traditional therapy conducted with a therapist. The significant result of this project was that robotic rehabilitation for stroke patients is possible and can show positive results in treatment.

2.2 CBM-MOTUS

The design of the CBM-Motus addresses specific requirements related to the tele-rehabilitation of elderly, typically post-stroke, patients. In this scenario a high degree of portability and robustness are requested. For this reason the robot has been designed to be light, compact and robust to be moved to and easily mounted at the patient's site. Also, the low cost is one of the requirements that the machine tries to address through the simplicity of its mechanical system. Moreover, the machine has been conceived to optimize the dynamic behaviour in the interaction with the patient by addressing requirements of high levels of safety and dependability. The CBM-Motus is regarded as a cartesian manipulator with two linear joints d1 and d2.The design of this robot is based on the portability and robustness in order to use it for home based stroke rehabilitation or it can be teleoperated as per the requirement [1].

Design of this robot is such that it achieves back-drivability i.e low friction, low and isotropic apparent inertia when back driven. A large workspace to allow the administration of several rehabilitative treatments (target: ¿ 500 mm x 500 mm) and interaction forces up to 50 N. The two modules are actuated using DC servomotors fixed to the frame (Aerotech BM 250) with rated torque of 2 Nm and peak torque of 5 Nm. Being R = 25 mm the radius of the pulleys, the maximum force which the robot is able to exert is 80 N (peak force: 200 N). The planar workspace is square in shape with a side of 550 mm. The overall dimensions of the robot frame are 830 x 820 x 110 mm. The total mass (frame and motors included) is about 30 kg [31].

2.3 GENTLE/A (Haptic Master)

The Haptic Master (HM) robotic arm was developed at FCS Control Systems in the Netherlands. This was the third iteration of the arm and unlike the previous models featured a 7 DoF arm. The purpose of this arm was to aid in the rehabilitation of stroke patients, like other platforms it was not mobile. It has been used for Rehab in the Gentle/A system [5]. It is an RPP Robot also known as a Cylindrical robot because it has a cylindrical workspace. Though this one has constricting limits to its workspace so it is more of a pie wedge shaped workspace, with a volume of 80 L, as seen in the figure below [26].

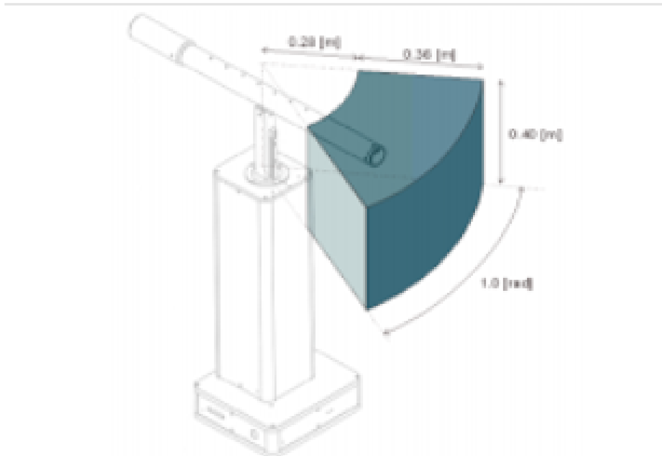


Figure 2.1: HapticMaster workspace [26]

Workspace	$80 \cdot 10^{-3}$ [m^3]
Position resolution†	$4 \cdot 10^{-6}$-$12 \cdot 10^{-6}$ [m]
Stiffness†	$10 \cdot 10^{3}$-$50 \cdot 10^{3}$ [N/m]
Nominal/max force	100/250 [N]
Minimal tip inertia	2 [kg]
Maximum velocity	1.0 [m/s]
Maximum deceleration	50 [m/s^2]
Force sensitivity	0.01 [N]

†depending on the degree of freedom and the position

Table 2.1: HapticMaster specs [26]

The above table lists the specs of the HapticMaster, and we can see it has a Nominal/Max Force of 100/250 N. Additionally with an update rate of 2500 Hz it is an order of magnitude greater than the “maximal human discrepancy value” [26], allowing for a smooth haptic experience.

2.4 NeReBot and MariBot

The NeReBot was developed at the Robotics Laboratory of the Department of Innovation in Mechanics and Management at the University of Padua in Italy. The MariBot was designed as the next step in the evolution of this system. While both of the robots are cable driven the NeReBot only had 3 DOF compared to the improved 5 DOF in the new MariBot.

Figure 2.2: NeReBot

In the NeReBot the auditory feedback consists of a beep and the beginning and end of point-to-point exercises. The motors have a max winding speed of 100mm/s and their encoders have a resolution of 1000 ppr [17].

The MariBot has a similar functionality in the cable driven part but adds a 2 DOF RR planar arm [17]. The joint motors can provide 2 Nm of continuous torque. While It does not have the direct drive pulley-motors of the NeReBot, that allow wire tension estimation from the current, it uses deformable elements with strain gauges to measure the wire tension.

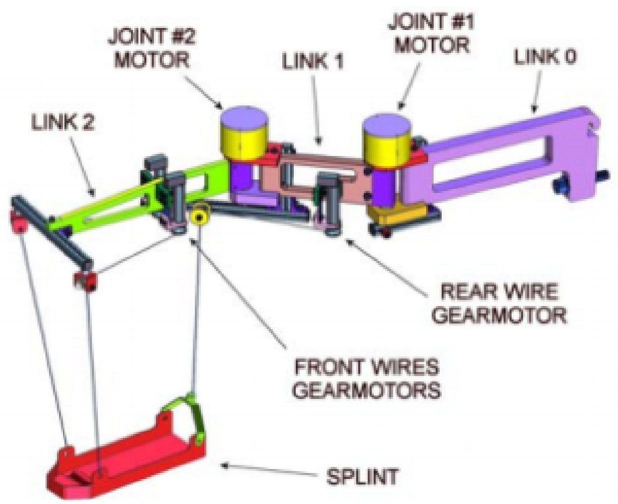


Figure 2.3: MariBot [21]

2.5 UL7

The UL7 exoskeleton was developed at Bionic labs at UCLA by Jacob Rosen. This was the third iteration of the arm and unlike the previous models featured a 7 DoF arm. The purpose of this arm was to aid in the rehabilitation of stroke patients, like other platforms it was not mobile.

This system contained a lot of features similar systems did not. The arm allows for rotational movement along all three axis of the shoulder and wrist as well as a single axis rotation for the elbow. The mechanical design of this system was based on activities of daily living (ADL) motion. This data was used to find the placements of all the axis of rotation. This arm is a high DoF linkage, this leads to the existence of singularities. While, this points cannot be avoided, the team at UCLA managed to place them in unreachable location. The mechanical actuation of this system was designed to have minimum backlash and be low weight. To achieve this a cable driven system was used to power the joints. The cables were kept in constant tension through the use of pulleys. The shoulder and elbow were powered by 6.2Nm maxon motor while the wrist was powered by a 1 Nm maxon motor [20].

A novel controller was to provided stable support of the arm was created. This controller lead to semi-global asymptotic stability. The controller proposed was based on a linear PID controller. This controller does not need a full dynamic model to find a set of tuning parameters that lead to a stable system [30].

2.6 ARM GUIDE

Developed with the motive of making a comparative study between unassisted and assisted rehabilitation, ARM Guide's findings were rather discouraging and called for enhanced en-

Figure 2.4: Bi-Manu-Track

gagement devices. Since the major stimulus to recovery of an damaged motor system is the patient's efforts itself, ARM Guide's full assistance in motion caused slacking rather than improvement in motion range/ease. The proposed design had linear rails with adjustable orientation in space. In also incorporated 6 axis force sensors to measure the interaction forces between the user and the device. Owing to the bulky size, lack of user intent detection and poor flexibility, it could not serve as a promising setup for therapeutic rehabilitation or scientific research in similar fields.[]

2.7 GEOMAGIC PHANTOM SERIES

Developed by Geomagic, Phantom is multi-purpose haptic device with high precision, large workspace, adjustable stiffness force feedback as well, all while still providing considerable amount of force. This series of devices from the company may have various specifications, but is designed aptly to replicate the human arm motion at shoulder,elbow and the wrist. The underlying construction has a passive gimbal with a gimbal as the end-effector. This provides the necessary position feedback for rotation along all the axes. With the size being as small as a small backpack, this device offers appreciable precision with enough power to serve for later stages of rehabilitation where accuracy and repeatability is more sought after, than restoration of power in preliminary stage.

It provides torque feedback from 3 revolute(roll, pitch and yaw) joint as well as force feedback along three linear degrees of motion. The imperative limitation of such compact systems for rehabilitation is always the upper limit for their force output and instability in case of overloading. Moreover, given the small links lengths; this setup can function well for lower arm/forearm exercises, but not for the upper arm.

2.8 T-WREX

"Therapy-Wilmington Robotic Exoskeleton was developed as the doctoral dissertation research of Dr. Robert Sanchez and was work supported by NIDRR. A 5-DOF system using elastic bands to overcome the human arm weight. Being a completely passive device,T-WREX provided only the least possible assist force using the elastic bands and was largely instrumental in measuring the performance of the patient 's hands - their reach, speed of motion and precision using sensors. It was essentially a non-robotic device with the main motive to assess the Fugl-Meyer score the patients and determine their recovery progress; but since it did provide partial support, it does suggest use of passive elements like springs and elastic bands for rehabilitation setups [22].

2.9 UHD

The Universal Haptic device is one of its kind 2-DOF haptic rehabilitation device that is suitable for am as well as wrist functionality restitution. Impedance control was implemented with proportional force control scheme. A passive universal joint locking/unlocking scheme decides arm/wrist exercise mode. Two motors actuate the two actuated bars in perpendicular directions using strings and pulleys for power transmission. Use of springs and strings makes the motion of the setup, smooth and devoid of unwanted jerks. Given the multi-mode operation; this setup could target particular exercises for the wrist or the arm like pronation/supination and flexion/extension [19].

2.10 MIME

Mirror Image Movement Enabler (MIME) robotic device is specially for shoulder and elbow neurorehabilitation in subacute stroke patients. MIME incorporates a PUMA 560 robot (Staubli Unimation Inc, Duncan, South Carolina) that applies forces to the paretic limb during unilateral and bilateral movements in three dimensions. Robot-assisted treatment (bilateral, unilateral, and combined bilateral and unilateral) was compared with conventional therapy. Combined unilateral and bilateral robotic training had advantages compared with conventional therapy, producing larger improvements on a motor impairment scale and a measure of abnormal synergies. However, gains in all treatment groups were equivalent at the 6-month follow-up. Combined unilateral and bilateral training yielded functional gains that were similar to the gains from equivalent doses of unilateral-only robotic training, although the combined group had more hypertonia and less movement out of synergy at baseline. Robot-assisted treatment gains exceeded those expected from spontaneous recovery. These results are discussed in light of the need for further device development and continued clinical trials [3].

Compared to MIT-MANUS, the device allows more naturalistic motion of the arm because of its six degrees of freedom (DOF), but must rely on force feedback so that the patient can drive the robot arm. Four control modes were developed for MIME. In the passive mode, the patient relaxes and the robot moves the arm through a desired pattern. In the active assist mode, the patient initiates a reach toward a target, indicated by physical cones on a table top, which then triggers a smooth movement of the robot toward the target. In the active-constrained mode, the device acts as a sort of virtual ratchet, allowing movement toward the target, but preventing the patient from moving away from the target. Finally, in the mirror image mode, the motion of the patient's less impaired arm is measured with a digitizing linkage, and the impaired arm is controlled to follow along in a mirror symmetric path [15].

2.11 Comparison

The device attributes are compared in Table 2.2 and Table 2.3. The decision for which device to use must depend on what you hope to accomplish.

Table 2.2: Device Comparison Part 1

System	MIME	ARM-Guide
DoF Assisted	3 Active DOF, 3D space	1 Active DOF, 2D space
Upper Limb Segment Used	Shoulder + Elbow	Shoulder + Elbow
Type of System	Single-point + digitizer bilateral	Single-point, unilateral
System	Bi-Manu-Track	Gentle/S
DoF Assisted	1 Active DOF at one time	3+1 Active, 2 Passive DOF, 3D space
Upper Limb Segment Used	Forearm + wrist	Shoulder + Elbow + Forearm + wrist
Type of System	Multi-robot, bilateral	Single-point + wire-based, unilateral
System	MIT-MANUS (InMotion2)	MIT-MANUS (InMotion3)
DoF Assisted	2 Active DOF, 2D space	3 Active DOF
Upper Limb Segment Used	Shoulder + Elbow	Forearm + wrist
Type of System	Single-point, unilateral	
System	NeReBot	T-WREX
DoF Assisted	3 Active DOF, 3D space	5 passive DOF (3 shoulder, 2 elbow)
Upper Limb Segment Used	Shoulder + Elbow	Shoulder + Elbow
Type of System	Wire-based, unilateral	Fixed exoskeleton, unilateral

Table 2.3: Device Comparison part 2

System	Novint Falcon
DoF Assisted	3(All Active)
Upper Limb Segment Used	Wrist
Type of System	Single Point, Unilateral
Feedback	Haptic + Visual
Payload	>2 lbs
System	Geomagic Phantom
DoF Assisted	6 DOF(3 Active + 3 Passive))
Upper Limb Segment Used	Forearm + wrist
Type of System	Single Point, Unilateral
Feedback	Haptic
Payload	1.8 lbs
System	Universal Haptic Device
DoF Assisted	2 DOF(Active)
Upper Limb Segment Used	Arm + Wrist
Type of System	Fixed Mounted Platform
Feedback	Haptic
Payload	16.86(Arm Mode), 4.49(Wrist Mode)

Chapter 3

Methodology

Overview

In this section, the materials, equipment, tools and resources that are used in the project will be discussed and presented with the logic behind each decision. The electrical, mechanical, and software portions of the robotic manipulator are discussed in greater detail below.

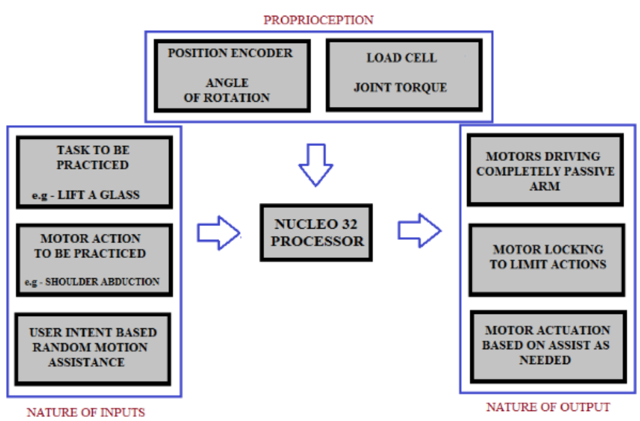


Figure 3.1: Workflow Diagram

The flow-chart above shows the basic framework of the proposed robotic manipulator based rehabilitation device. The workflow consists of the modes of operation, their corresponding outputs, and the inputs that drive the manipulator.

3.1 Electronics

3.1.1 Processor Board

A large amount of processing power was needed for the manipulator arm due its need to collect sensor data in real time, give actuator commands and implement real-time control sequences. Because of this, the STM32 Nucleo 144 development Board powered by a 32-bit STM32F746ZG MCU, a 32 bit micro-controller, was used. This is a high-performance low power micro-controller packed with 1 MB flash memory, USB/RJ45 Ethernet connector and umpteen number of GPIO pins.

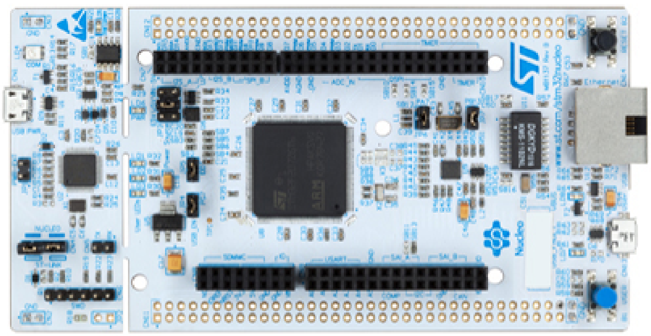


Figure 3.2: Board

3.1.2 Sensors

In order for the device to achieve the responses needed to make the robotic arm usable for haptic feedback, sensors need to be implemented to detect the angular position of the joints as well as the forces being applied on the joints. To solve this we will be using encoders and load cells at the motorized joints.

Encoders

The encoders used in this device are AS5055A Position Sensors from AMS. This are hall effect position sensors that are able to measure the orientation of a magnetic field. These sensors are placed a couple millimeters from a magnet that is attached to the shaft of the joint. The sensor reads the orientation of the magnetic field with a 12-bit resolution and then translate that into an electrical signal. This signal is typically very small in amplitude and so requires an amplifier to help increase the signal into readable ranges. The amplifier used for this robotic arm is a INA826 Supply Instrumentation Amplifier from Texas Instruments. Both of these electronic components were chosen because they were able to fulfill the necessary requirements and were readily available.

Load Cell

The load cell chosen for this robotic device was a 10kg Straight Bar Load Cell from Sparkfun. This load cell works well with the device because of its relatively small size and weight. The manipulator arm will need to assist a person through their arm motions and so the more lightweight the arm is the more effectively the motors can apply a force at the end effector.

The load cells were also a good fit for our device because stroke patients will be targeted audience for the device. Stroke patients, especially in the early stages of their recovery, are only able to apply small amount of force and so a load cell that can detect high forces is not required.

3.2 Mechanical

3.2.1 Actuation

In order to assist a patient in their movement a source of motion and power must be applied. This can be achieved using motors at each of the robotic arm's joints. In order to have a smooth motion when the robot arm must apply enough torque from its motors. This can be determined by the amount of force needed at the end effector to pull a human arm. Another desirable quality for motors is backdrivability. This allows for an external force to freely rotate the shaft of the motor without damaging the internal mechanisms. It is basically impossible to find a motor with both of these requirements at a low cost. In order to work around this issue, a good control scheme based on external stimulus must be implemented. With this a strong non-backdrivable motor can be made into a backdrivable motor, by sensing the input forces and powering the motor with the respective polarity. For this device the JX Servo PDDO-HV5932MG motor is being used. This decision is mainly based on the current availability of motors and time frame of the project. In order to get the appropriate to torque needed for the device, 2 motors will power each joint doubling the strength of 1 motor. The motors have relatively decent backdrivability, but will be made better once a force-based control scheme is implemented.

3.2.2 Hardware Design Structure

The design presented in Figure 3.3 is used as a reference for building the rehabilitation robotic arm. The reference arm has 3 3D printed links with 1 yaw and 2 roll joints with constraints in angles of rotation. The Joint restrictions of the arm can be seen in Table 3.1.

Table 3.1: Joint Limits

Joint Name	Minimum Angle of Rotation	Maximum Angle of Rotation
Base Yaw(J1)	$-60\,\text{deg}$	$90\,\text{deg}$
Base Roll(J2)	$-18\,\text{deg}$	$110\,\text{deg}$
Elbow Roll(J3)	$30\,\text{deg}$	$210\,\text{deg}$

Changes had to be made to accommodate the desired application. It was found that the reference arm could not provide the appropriate power, move in the necessary workspace,

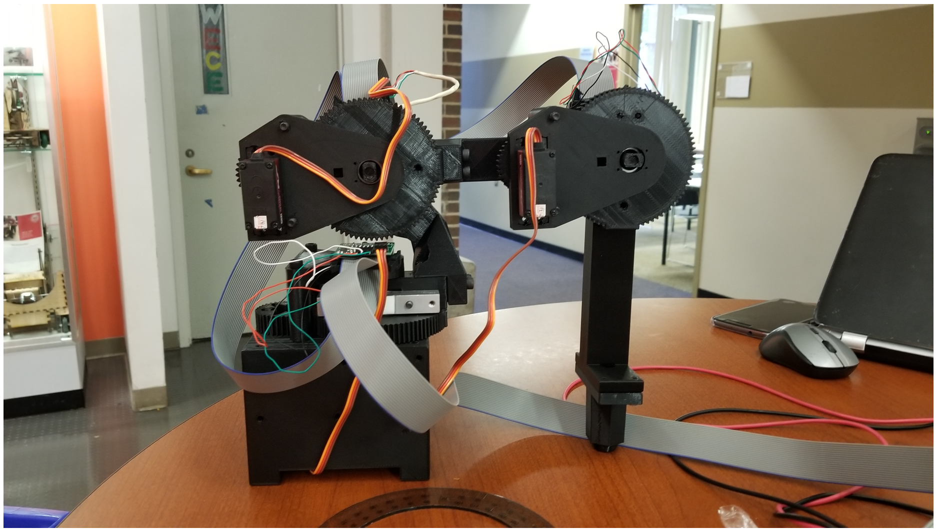


Figure 3.3: Reference Design of the Robotic Arm

or have an appropriate end-effector. To increase the power of the joints, and to support the payload at the end effector, an extra motor was introduced at every joint as shown in Figure 3.4. The design also considered balancing the weight along the central axis of the link to avoid torque due to motor weight

.

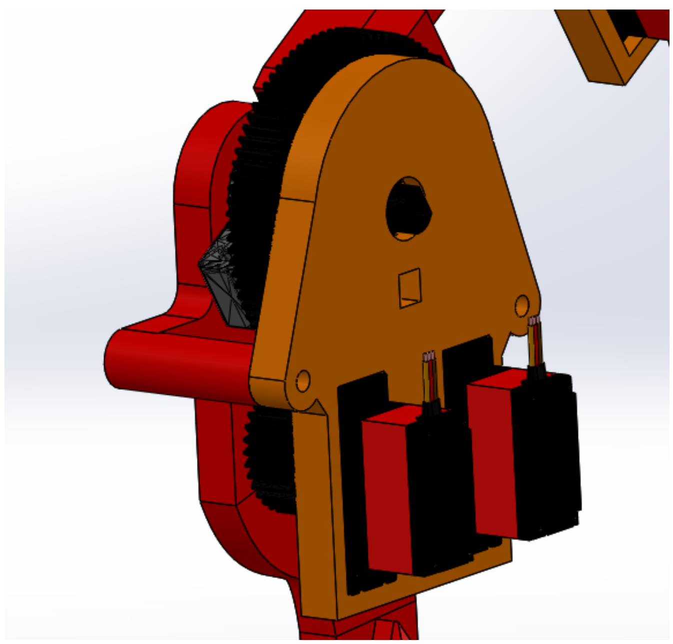


Figure 3.4: Double Motor Support

In order to increase the workspace an the flexibility of the robot arm the lengths of the joints were scaled by 1.5 to the arm in Figure 3.5 and a semicircular track, shown in Figure 3.6, was developed.

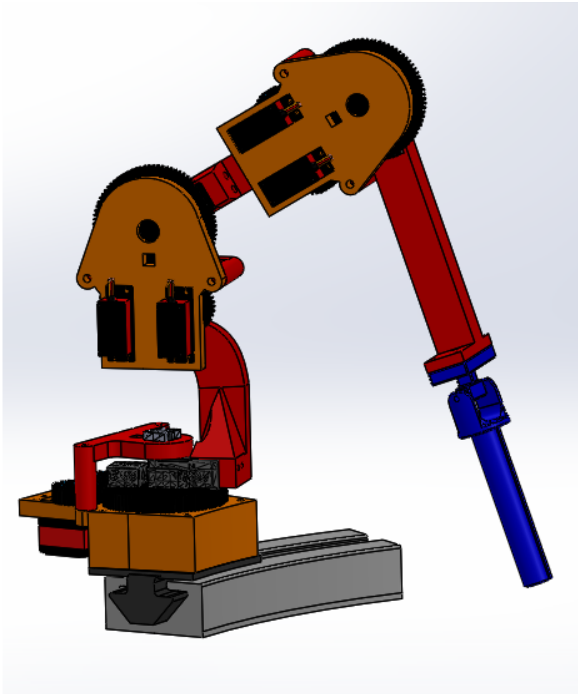


Figure 3.5: Design of the New Robotic Arm

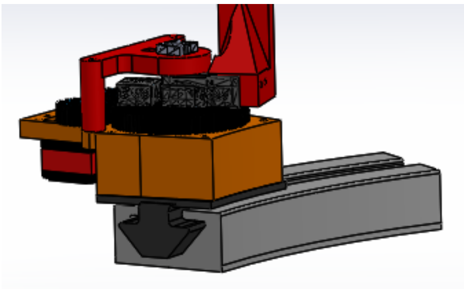


Figure 3.6: Semicircular Track for Robotic arm

This track allows the robotic arm to be mounted and the workspace to be manipulated to the desired orientation for the appropriate rehabilitation exercise, thus increasing the workspace even more. To overcome the limitation at the lower link joint of having less angular freedom at the joint we have come up with the modification in the orientation of the lower joint as shown in Figure 3.7.

The arm is connected to an wrist joint end-effector with an adapter, this end effector is discussed more in the following section. This arrangement provides the device with 3 active and 3 passive degree of freedom.

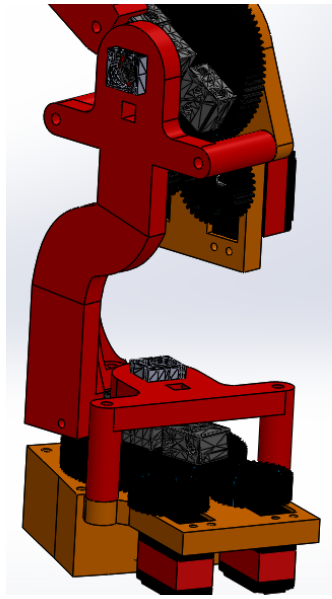


Figure 3.7: Lower Link Modification

3.2.3 Three Degrees of Freedom Human Interface Handle with Vibrational Feedback Support

A 3 DoF wrist with a handle was created so that user can grab on to it while the robot is moving. Several redesigns were made to reduce weight and combat singularities. The wrist was created to provide a roll, pitch, roll rotation without restricting the natural movement of the users wrist. To avoid singularities, the wrist was designed so that all of its 3 axis passed through the same point. To allow for smooth friction 5mm free rotation ball bearing were used as an interface between the bolts and rotational pieces. They were analyzed to ensure they could handle the expected loads of up to 60 lbs. A vibration motor was incorporated into the end-effector so that the user could be given vibration feedback whenever necessary. Finally a rubber grip was used to provide better grip of the handle. The resulting end effector can be seen below in Figure 3.8 .

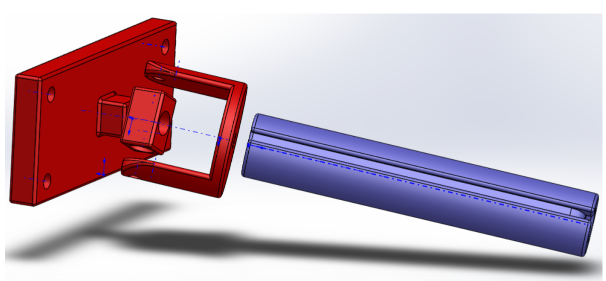


Figure 3.8: Assembled end effector

This design is compact and lightweight, allowing for the user to move more freely and for more torque to be dedicated to moving user's arm.

3.3 Software

3.3.1 System Architecture

User Datagram Protocol (UDP) is a connection-less protocol and used in applications like SNMP (Simple Network Management Protocol), DNS (Domain Name System) where data packets arriving out of order, unreliability are not of concern but immediate send through of the data packet matters. Since UDP does not involve connection establishment, therefore applications like communication between the haptic device and the GUI where connection establishment delays need to be avoided, UDP is preferred over TCP. If TCP packet is lost, it will be resent. That is not handy for applications that rely on data being handled in a specific order in real time.

We referred to the architecture of Novint in which they developed a low-level driver software named Haptic Device Abstraction Layer (HDAL). It handles the low-level communication between the Falcon and the computer. The software layer above HDAL, called Haptic Effects (HFX), which is used for creating force effects on games.

We looked at the number of programming languages and platform available for GUI for the high-level control such as XAML, Python with Qt Framework, Java, C++ with wxWidgets library, Citrus specifically used for game frameworks, PyGUI, wxPython. Some of these libraries or framework are cross platform while some are not. We found that CHAI3D and H3D are the open source platforms/frameworks which works using the C++ and Python languages supports a wide variety of commercially available three, six and seven degree of freedom Haptic devices, and makes it simple to support new custom force feedback devices.

3.3.2 Communication

The existing software was written in Java use HID (Human Interface Device) [4] to communicate with the Nucleo board. The original code was written in Apache Groovy and was integrated into Bowler Studio. This made it very difficult to work with and modify. Java is also not a widely used language so additional communication channel will be set up to send the message to a more commonly used language, Python. This will be accomplished through a UDP (User Datagram Protocol). More details will be discussed in Chapter 4.

3.3.3 Graphical User Interface - Video Game

Many of the research projects have found that using games is one of the best way to incentivize stroke patients to continue in their exercises, as the patients tend to procrastinate and eventually stop their exercise all together. It has also been discovered that video game consoles such as the Playstation and the Wii are useful, but only during in later stages of recovery. In initial and intermediate stage, the games should be simple and the level of game should increase as the patients get use to a certain level. Some of the games developed by other researchers working in this field are a Baseball catching game, Helicopter, Pong, Catch the kitty, dirt race and many other games which helps the patient to exercise their shoulders, elbows and wrist during recovery

A number of platforms are available to develop rehabilitation games, such as Chai3D, H3D, Pygame package, Processing environment, PyQt package and many other softwares and packages. Out of this list, H3d and Chai3D are open source softwares for developing games and are used by Novint, Geomagic and other Haptic Device companies. These software programs were specifically designed for developing 3D games which would lead to more complexity than needed for this project. This led to development of a simple 2D game that will be used for initial and intermediate stages of rehabilitation. A Pong game, where the user was forced to exercise his/her shoulder joint, was decided upon due to its simplicity. The motion was restricted only to the x-axis so that the patient could only exercise his/her shoulder joint at a time. The Pong game is flexible to allow for the changing of direction for the paddle and for another joint to be mapped to the paddle, if another joint is being trained. The main idea was to develop this game in the Processing IDE and interface the game with the arm developed.

3.3.4 Trajectory Planning

The robot will be guiding the user through different motions that are needed for everyday exercises and motions. In order to determine which trajectories are important, a motion capture (Mocap) system was implemented to monitor a subject's motions. Once the trajectories they can used as a reference for a dynamic motion primitive (DMP) algorithm. The exercises chosen were a Shoulder flexion and Circular motion exercise. First, we used the full body Plug-in-Gait available in the nexus vicon system as seen in Figure 3.9. This diagrams gives the locations if full body motions were to be recorded. Since the project focused on the upper body, only the markers associated with the upper body were used when creating a template for the mocap system.

Once, the template was made, seen in Figure 3.10, data was taken from three different test subjects. First, the subjects were prepared for the motion capture by attaching the markers on their body and were asked to perform the two exercises described above. Static data was first collected and then dynamic data to be used for generating trajectories. Subject preparation was different for all the subjects because their differing body builds which made some of the markers not visible on the mocap system, this can be see in Figure 3.11.

Only one of the three data sets collected was usable. The other data sets had some missing data points in it, which prevented the generation of a complete trajectory. Trajectories were successfully generated from the usable data set, seen in Figure 3.12 and were used as templates for the DMPs.

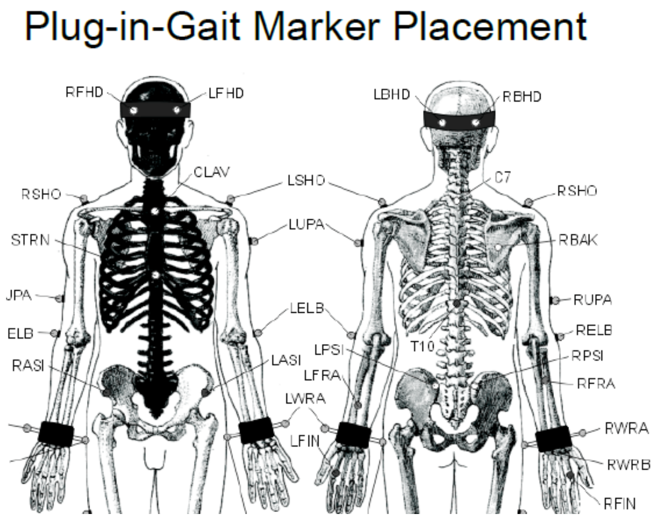


Figure 3.9: Mo-cap Marker Positions on the Human Body

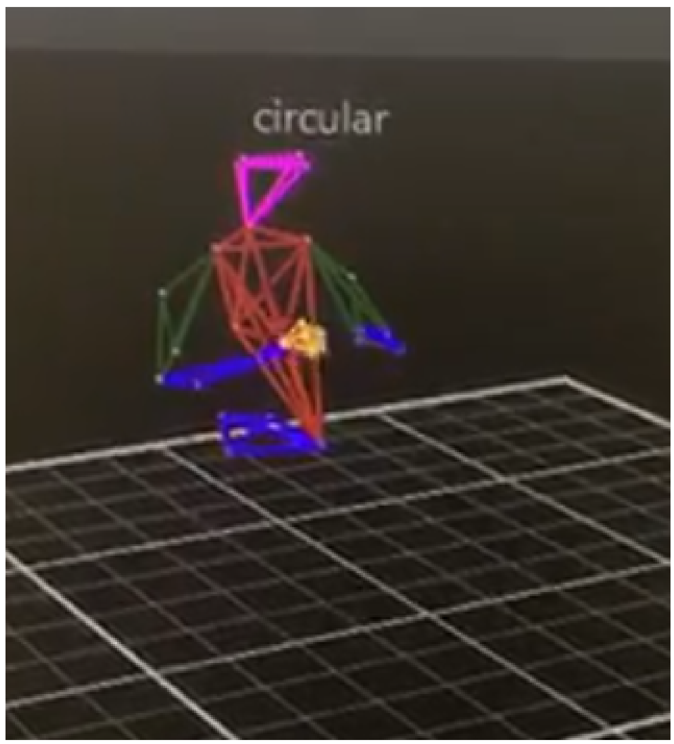


Figure 3.10: Mocap Upper Body Template

3.3.5 Dynamic Movement Primitives

Often times in learning by demonstration, it is required to reposition the start or end or modify other trajectory parameters like velocity and acceleration as well as avoid dynamic

Figure 3.11: Mocap Subject Preparation

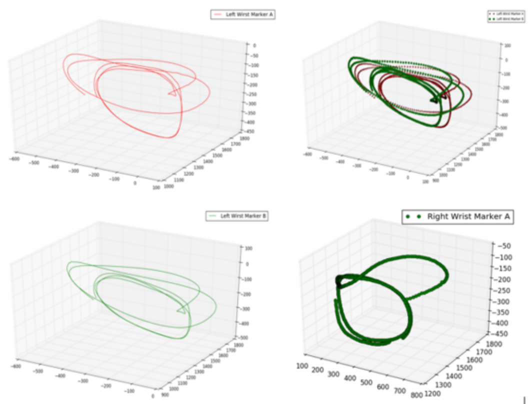


Figure 3.12: Trajectories from Mocap Data

obstacles. Such a large pool of challenges can all be solved by Dynamic Movement Primitives (DMPs). This is a method of representing complex motor actions that are flexible and can be adjusted without manual tuning. DMPs are mathematical formalization of motion primitives. Each DMP is a non-linear system in itself. It basically is the combination of a stable known behavior component combined with a forcing term f that makes the complete system capable of tracking desired trajectories. DMPs only need one demonstration of the trajectory/action to learn and re-execute. These will be helpful in creating a mode that pulls the user along a pre-planned trajectory of an exercise.

Chapter 4

Software

Overview

The software was built in to be as modular as possible. The individual components could be tested in isolation. This allowed for the system to be unit tested at every layer and for abstraction to be added at each level. This allowed the layers to be black boxed as they were built and tested. It also allows for the piece of software to be replaced without having to rewrite the entire software stack.The layers of the software are illustrated in Figure 4.1. The systems are shown in *pink*, the servers are shown in *green*, and the external inputs (user) are shown in purple. This was done to show the different types of layers the are present in the system and how the they are all connected to each other. A more detailed explanation of each layer will be discussed through out this chapter.

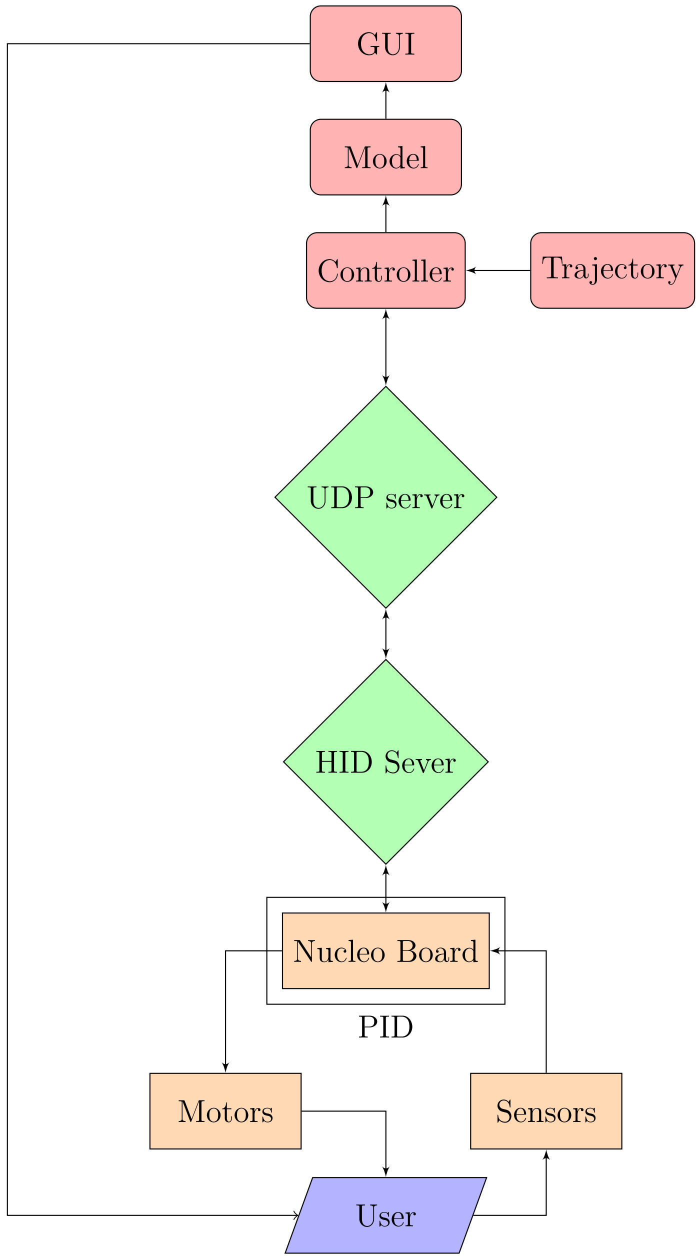


Figure 4.1: System Overview

4.1 Nucleo Board

The bottom layer is the user themselves. They are physical attached to the robotic system. They can move the robot around or the robot can move them. There are two connections to the Nucleo board. The motors feed into the user by moving the arm around, while the sensors allow for feedback communication. The low level control of the motors and the sensor acquisition are handled on the Nucleo board. There is a low level PID controller on the board. This ensure that the servos reach their setpoint. The software on the board was developed by Kevin Harrington [8]. It uses an object oriented approach to the low level code. There is an abstract class that can be extended to execute different commands when a packet of data is sent down from the computer. This class can be used to create multiple different actions. They can be distinguished from each other from a command bit that is set in when the class is created.

There were four different classes setup to handle commands. Two of the classes set up configuration parameters. The other two are used to control the robot. The first command class is used to set the PID values for set point control. The second command class is used to set up the PID constance for the velocity controller. The two control classes are used to do set-point control and velocity control respectively.

4.2 Servers

The Nucleo talks to the computer over HID, this communication interface requires no drivers to connect to the computer [4]. An array of 15 floats are sent up and down from the board. This is known as a *packet* This commands array can be filled to numerous ways to send commands to and read data from the board. To send the commands up and down stream, the float array is convert into a byte array and sent over HID.

There are two different "servers" on the board. The first server is written in Java. This server talks to the board over HID. It relies only on the HID4java library [7]. However it was desired to use python for the majority of the high level control. To do this a second server was setup that communicates over UDP. This connects the Java code to the Python code the Java UDP $\rightarrow$ HID server waits for a command to come over the UDP channel, when it receives a packet it immediately sends it down to the board and waits for a reply over to come back over HID. It then sends the received packet back up over UDP to the python code. The Java server does not interact or change the packet. It just passes the packet from HID to UDP and UDP to HID. The entire communication loop take on average 2ms to send and receive a message. This is 10x faster then the previous HID-matlab method used.

4.3 Controller

The high level language is written in Python and used a combination of functional and object oriented programming. This was done so that segments of the code could be reused more easily. It allowed for the physical parameters robot to be specified in one class and it would be distributed through the reset of the program. This was achieved by creating a *robot* class that contained parameters that can be set on run time to specify the physical parameters of the system. This class is then passed into other functions where the needed parameters are extracted and used for what ever application that function is achieving. The details of how the controller is implemented will be discussed in Chapter 5.

4.4 Model

To test the dynamics and control of the robot a dynamic model was created to see how the system moves before it is applied to the physical robot. This ensured that the robot was not broken during the testing phase of the controller development. The test is a simple stick model that only represents the position and angles of the joints. The plot can be seen in Figure 4.2.

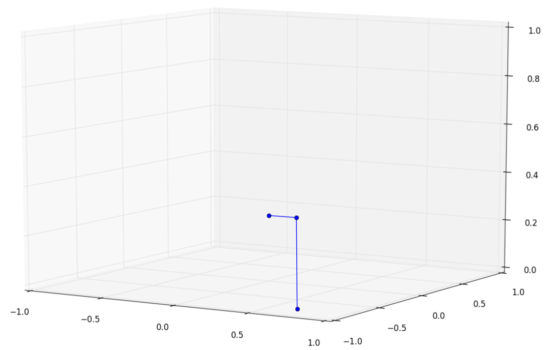


Figure 4.2: Live plot of the arm

4.5 Graphical User Interface (GUI)

The top layer of the system is the graphical user interface (GUI). This layer is a game of Pong, displayed in **??**, that will prompt the user to move the robot around in order to control the actions within the game. It was created in a Processing Python Mode to use the rich and simple to use library to create a GUI and then converted as a .py file from .pyde file. The user can control the paddle to hit the ball back and forth, as it can be seen in the figure below it restricts movement of the paddle in up and down direction that is the y-axis only based on the preliminary exercise to be carried out by the patient. It can also be seen in the figure that there is only one paddle as it is only a single player game and hence the ball will bounce back from the wall. The speed of the paddle and the ball can be changed depending on the motion required by the user. As mentioned in some of the research papers, the joint which can be recovered first is the shoulder joint. Because of this motion required does not need to go from side to side movement, making it a good exercise to be matched to the vertical movement of the paddle in the game. The movement of the paddle can be changed if the end user wishes to exercise in x-axis or the horizontal plane. The score has not been displayed on the display as some of the papers suggested that it would discourage the patients in the initial stage. The game was designed in such a way that the patient would not feel that discouragement and may have some fun in the process of rehabilitation.

Figure 4.3: Pong Game

4.6 Dynamic Movement Primitives (DMPs)

DMPs are broadly classified as rhythmic and discrete. Rhythmic DMPS represents periodic non-terminating motions and discrete DMPS are used to formulate terminating motions. Discrete DMPs allow for the changing of start/end positions and other parameters while the system shall track the changes maintaining the same nature of motion. Rhythmic DMPs allow changing the time-period.

Discrete DMP have a point attractor as the base system while Rhythmic DMP have limit cycles. The equation below shows the mathematical representation of a DMP, where the first term is a stable system and forcing term f makes the real-time changes in the system to effectuate tracking. The forcing term f is composed of multiple non-linear systems distributed over time. A canonical system that defines the forcing term.

$$\tau \ddot{y} = \alpha_z(\beta_z(g - y) - \dot{y}) + f$$

Motion capture was used to collect data for exercises involving shoulder rotation(motion of wrist in transverse and frontal plane) for the right and left arms of a unaffected person. This data helped ascertain the demonstration trajectories needed for certain rehabilitation exercises. This gives an example trajectory for the end-effector of our rehabilitation device. Our DMP was trained on this demonstration to derive the trajectory parameters, i.e. the position, velocity and acceleration for all the points on the path. Our MOCAP data was only based on thee left and right wrist markers. Other markers were used for reference and the ones on the fingers and thumb were occluded from the camera view during the the demonstrations.

Despite given constraints, the trajectories along each individual axes(X, Y and Z) were obtained. The trajectory was derived in Cartesian coordinates and required the use of inverse kinematics to transform these values to the joint space. Finally, once the angular positions, angular velocities and angular accelerations for all the three joints were calculated, they can be sent and implemented on the rehabilitation device.

Chapter 5

Dynamics and Controller

5.1 Kinematics

The robot can be treated as a *RRR* manipulator. A kinematic diagram of this arm is shown in Figure 5.1.

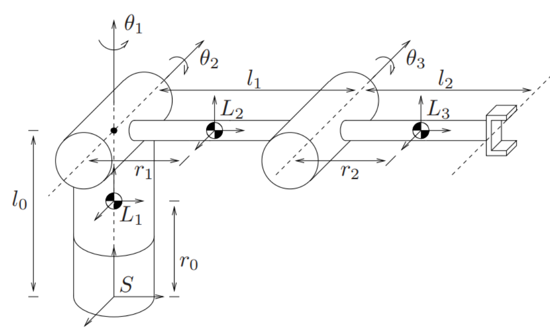


Figure 5.1: Kinematic model of the arm
[18]

The forward kinematics for this arm are well known and can be solved geometrically. They are given in Equation 5.1. These equations allow for the location of the end effector to be calculated based on the joint angles.

$$\begin{aligned} x &= (l_1 c_2 + l_2 c_{23}) c_1 \\ y &= (l_1 c_2 + l_2 c_{23}) s_1 \\ z &= l_0 + l_1 s_1 + l_2 s_{23} \end{aligned} \tag{5.1}$$

An exact equation for the inverse dynamics can also be found. This allows for the joint angles to be found from the location of the end effector. This can be shown in Equation 5.3.

$$\begin{aligned}
\theta_1 &= atan2(y, z) \\
\theta_3 &= -acos(\frac{(x^2 + y^2 + (z - l_0)^2 - l_1^2 - l_2^2)}{2l_1 l_2} - 0.5 * \pi \\
\theta_2 &= atan2(z - l_0, \sqrt{x^2, y^2}) - atan2(l_2 s_3), l_1 + l_2 c_3)
\end{aligned} \tag{5.2}$$

To get the velocity of of the end effector the Jacobian of the arm needs to be calculated. This can be found by taking the derivative of the joint positions [18].

$$J = \begin{bmatrix} -l_2c_2 - r_2 - r_2c_{23} & 0 & 0 \\ 0 & l_1s_3 & 0 \\ 0 & -r_2 - l_1c_3 & -r_2 \\ 0 & -1 & -1 \\ -s_{23} & 0 & 0 \\ c_{23} & 0 & 0 \end{bmatrix} \tag{5.3}$$

5.2 Dynamics

The dynamic equations of motion were worked out using the Euler Lagrange method, the equation for Euler Lagrange is given by Equation 5.4 [24].

$$\tau = \frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{q}} - \frac{\partial \mathcal{L}}{\partial q} = M\dddot{q} + C\ddot{q} + G \tag{5.4}$$

Solving out the Euler Lagrange for this arm according to the method specified in [18] leads to definition of the variables found in equation Equation 5.5 The equation for $C(q, \dot{q})$ and G can be found in the literature.

$$M = \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix} \tag{5.5}$$

where,

$$
\begin{aligned}
M_{11} &= I_{y2}s_2^2 + I_{y3}s_{23}^2 + I_{z1} + I_{z2}c_2^2 + I_{z3}c_{23}^2 + m_2r_1^2c_2^2 + m_3(l_1c_2 + r_2c_{23}) \\
M_{12} &= M_{13} = M_{21} = M_{31} = 0 \\
M_{22} &= I_{x2} + I_{x3} + m_3 * l_1^2 + m_2 * r_1^2 + m_3r_2^2 + 2m_3l_1r_2c_3 \\
M_{23} &= I_{x3} + m_3r_2^2 + m_3l_1r_2c_3 \\
M_{32} &= I_{x3} + m_3r_2^2 + m_3l_1r_2c_3 \\
M_{33} &= I_{x3} + m_3r_2^2
\end{aligned}
\tag{5.6}
$$

The parameters for the robot are the following. The mass, link length and centroid are given in each column. The I_{xx}, I_{yy}, and I_{zz} for each link are given in the rows of the inertia.

$$
\begin{aligned}
m &= \begin{bmatrix} 1.01992 & 0.3519 & 0.22772 \end{bmatrix} \\
l &= \begin{bmatrix} 0.25107 & 0.191 & 0.37843 \end{bmatrix} \\
r &= \begin{bmatrix} 0.10424 & 0.14550 & 0.203 \end{bmatrix} \\
[I_x, I_y, I_z] &= \begin{bmatrix} 0.006757 & 0.0006036 & 0.0015514 \\ 0.001745 & 0.0005596 & 0.00006455 \\ 0.00706657 & 0.0006254 & 0.0015708 \end{bmatrix}
\end{aligned}
\tag{5.7}
$$

5.3 Compliant Controller

A controller was designed to both compensate for gravity and allow user to move the end effector around in task space. This controller would be used when the patient or user is moving the rehabilitation arm while it is passive, this is useful for stroke patients that have some strength, but may still need som assistance making particular movements. The the controller is a modified compliance controller [24] where it does not take in a reference trajectory and because of that it does not feed back the joint angles. For the purpose of design of the controller, the input to the system will be a force vector located at the tip of the end effector. The Jacobian can then be used to find joint torques. The controller is outlined in Figure 5.2.

To make this work on the physical arm several changes had to be made. Since we did not have direct access to the joint torques, they had to be measured from the load the cells on the joints. This values was then scaled and feed back into the systems as torque. To smooth the path the joint next joint angle was predicted by first order approximation. As detailed in Equation 5.8. This allowed the control to estimate the next position of the arm. This smoothed the trajectory of the arm and allowed the low level PID to operate as intended.

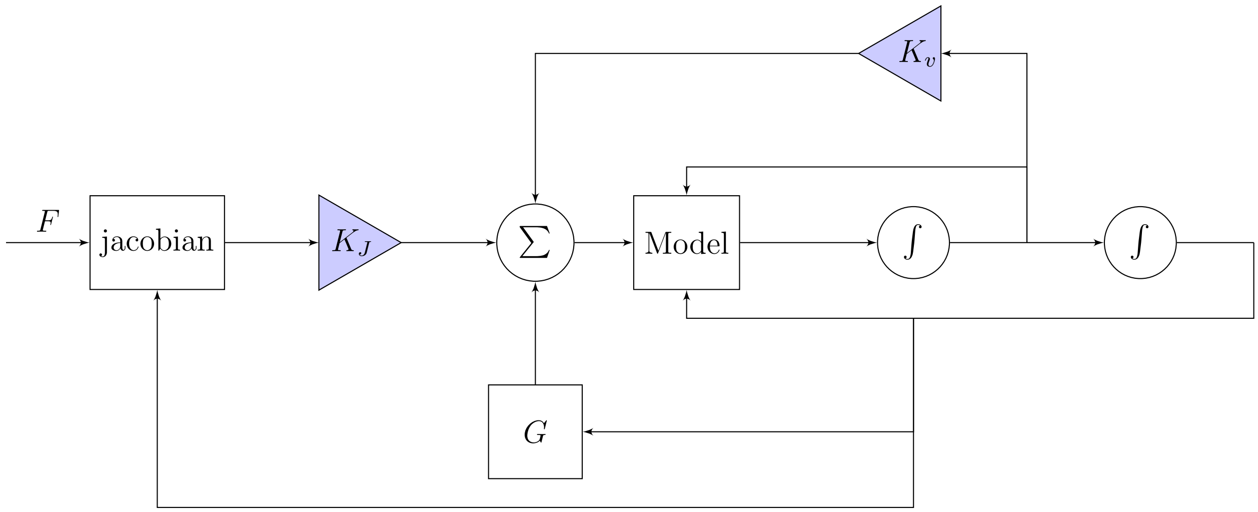
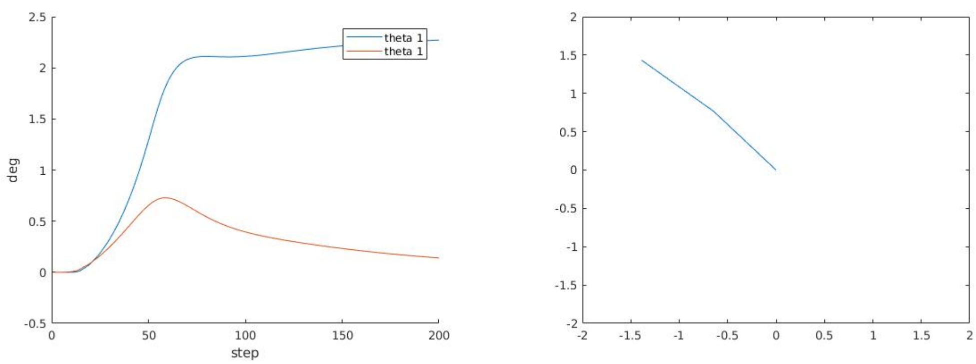


Figure 5.2: Compensation Controller



(a) Joint Profile

(b) Resting configuration of arm

Figure 5.3: Task and Joint space for simplified arm

$$q(k)_n = q(\dot{k} - 1_n \delta t + q(k-1) \tag{5.8}$$

To test this controller a simpled model was created in Simulink. This was done on a simplified two DoF arm. A force vector of $F = [-1; 1]$ was applied to the tip of the end effector. This caused the robot to move in the direction of the end effector. The joint angle profile and the reseting location of the arm is shown in Figure 5.3b.

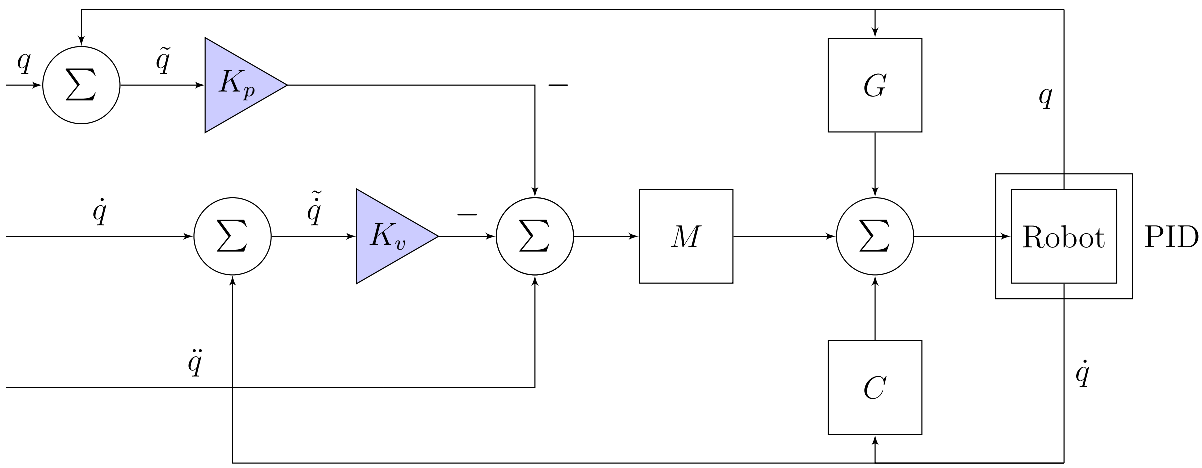


Figure 5.4: Inverse Dynamics Controller

5.4 Inverse Dynamics Controller

To ensure that the arm moves through the trajectories that are set by the DMP an inverse dynamics controller was implemented. This controller was designed to control the arm in joint space, allowing for patients with very little arm strength to be able to have their arm moved in the desired trajectories, discovered using motion capture. An inverse dynamics controller was chosen because of its simplicity. It allows for the linearized feedback of the state and a non-linear decoupling of the system dynamics. The controller is shown in Figure 5.4. The results of PD control tuning and a trajectory comparison can be seen in **??** and Figure 5.6.

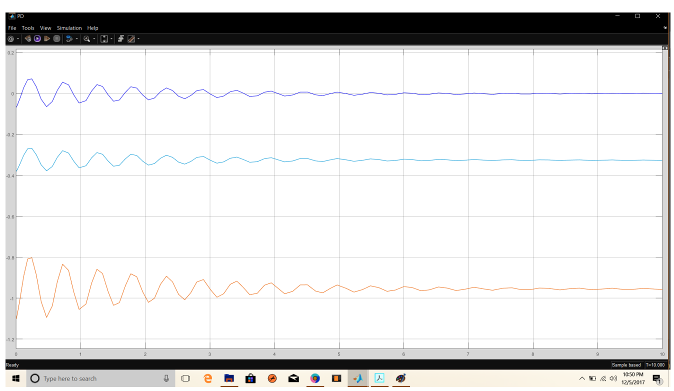


Figure 5.5: PD control tuning

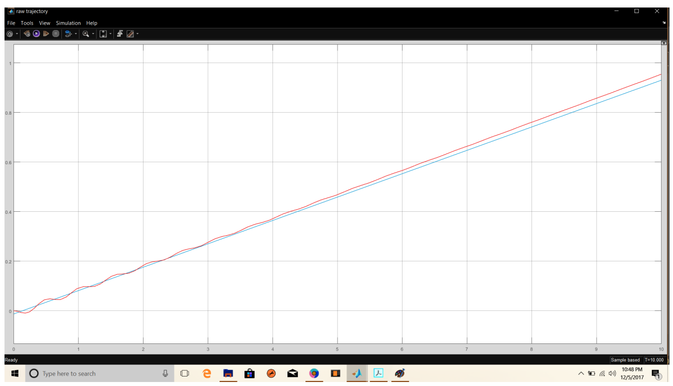


Figure 5.6: Desired trajectory (Orange) and Tuned Trajectory(blue)

Chapter 6

Results

6.1 Mechanical

6.1.1 Changes to Original Arm

The modifications made in the arm fulfilled the required application. These modifications not only provide with the increase in the power but also significantly increased in the workspace. In order to support the load at the end effector of the rehabilitation patients, an additional motor has been added at every joint of the arm. This improvement addressed the payload requirements. In order to increase the angular motion of the lower joint, the joint (J1) plates have been modified by increasing the angular range from 150 deg to more than 180 deg. Also, using the same modified joint plate at the joint(J2) the angular motion range increased from 128 deg to more than 180 deg. The link lengths were increased by a multiple of 1.5, thus increasing the reach. The track also allows for workspace to be further increased, as the arm could be positioned in multiple locations. These modifications significantly increased the workspace of the haptic arm. Due to these modification, the arm was able to perform rehabilitation exercises with a very good accuracy. The final arm can be seen in Figure 6.1.

6.1.2 End-Effector

The rubber grip allowed for good grip of the end effector. However for those that are suffering from stroke a strap may also need to be implemented to secure the grip. The Vibrational motor allowed for feedback to the game whenever the ball hits the paddle. While, the design while light and small, it does suffer from both gimble lock and singularities in particular positions. The bearings have allowed for smooth actuation, but the weakest point of the structure was determined to be the prongs on the single axis rotational component. In the future these could be enlarged or reinforced with bolts.

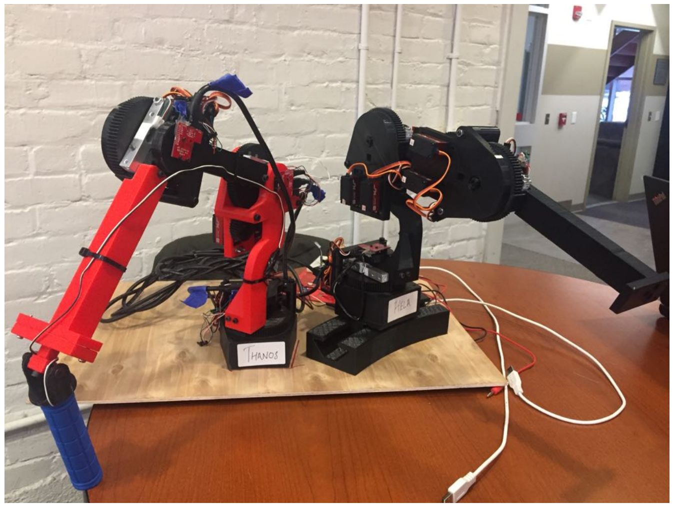


Figure 6.1: Bilateral End Effector Arms

6.2 Electronics

6.2.1 Sensors

The load cells being used are 10kg load cells. Initially there was concern regarding the ability to measure the range of forces that will be applied on the robotic arm. The load cells were able to accomplish the goal of measuring the loads that are placed on the arm. The data collected from these sensors were instrumental in the control of the arm and allows us to accurately take in information. The encoders used for the arm were effective in accurately measuring the angle of the 3 joints. The high resolution allowed for fine tune control to be obtained.

6.3 Software: Control

6.3.1 Communicate with arm

One of the major problems was the speed at which the computer talked to the board. Using the Matlab framework the average up and down communication time was 20ms. This did not allow for smooth trajectories to be achieved and caused the dynamic controllers to become unstable. To fix this problem a new communication frame was built as discussed earlier. This decreased the average communication time to 2.9ms. This allowed the controller to perform with real time communication and work properly.

6.3.2 Controller Arm

Two high level controller were created. One of these controllers was an inverse dynamics control that is used for position control for when the robot is following a trajectory. The second controller was a compliance controller. This was used to help the user move as they apply force tho the handle. This was applied to each of the joints so that it help the user move.

6.3.3 Dynamics of the Robot

The dynamic equations of the 3 DOF arm was derived for the morphology of the robotic arm. The inertia, coriolis and gravity matrix were obtained from the CAD model designed in Solidworks. Unfortunately the PLA material, used in printing the arm, was not available in Solidworks and so mass of each arm was calculated using the density of the material and the volume reported from Solidworks. From there inertia matrix was calculated. The coriolis matrix was obtained from Solidworks for individual link. Using the given information regarding the robot the jacobian, forward and inverse kinematics were derived and then implemented in the Dynamics code.

6.3.4 Integration of Arm with the Game

The game was developed in Python using the pygame package. The arm was integrated with the game, by mapping the task space to actions in the game. Continuous values of the end-effector position were obtained from the forward kinematics and then given as an input that controlled the position of the paddle. A basic AI was created to play against the patient. The AI tacks the position of the ball and moves the opposing paddle accordingly, this was done to give the patient an interactive feel. The game provided the feedback to the patient by vibrating the motor whenever the patient's paddle hit the ball creating a haptic feedback for the patient in the robotic device.

6.3.5 Implementation of DMP

The trajectory obtained from the DMP's are essentially in the human frame of reference with origin at shoulder joint. The trajectories were first scale down by half using the generalization feature from DMPs. Further transformation to the robot's Cartesian coordinate frame using an educated guess resulted in the same trajectory.

Using inverse kinematics, the joint angles of all the individual joints needed to complete the trajectories were derived. Before testing these set of joint angles on the robot, they were simulated in Python. The results of the simulations, as seen in Figure 6.2 were quite promising.

However, the joint angles in the inverse kinematic equations do not account for the joint constraints of our robot. When these constraints are taken into consideration, a crooked

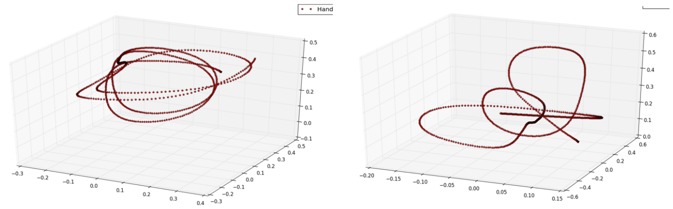


Figure 6.2: Final Arm Trajectory in Robot Frame

version of the trajectory is formed. The trajectory has the features of the original demonstrations, but is truncated on some edges.

Chapter 7

Future Work

There are many robotic systems available in the market for Stroke Rehabilitation targeted towards the upper limb, but all of the systems are based on the already available industrial robots such as the famous MIT-MANUS, MIME. This makes them very expensive, not easily accessible to everyone, and not easy to use as they need an expert to run the robot. This shows the need a robotic device with haptic Feedback at minimum cost possible, so that it can be affordable to everyone. As a proof of concept, two small sized robotic arms with haptic feedback were developed for bilateral rehab exercises. they have 3 active DOF and 3 passive DOF on the end effector, enabling the patients to perform shoulder and elbow exercises. The arm is also interfaced with a simple 2-D Pong game to visualize the movements of the patient's arm and make the exercises less tiresome to patients. While the Proof of Concept for the arm was created, there is still much that can be done to improve the system.

7.1 End Effector

One area of improvement lies in the end effector of the robot. The current end effector has threads grinding into the bearings used for rotation. While this is acceptable for a short term, it will inevitably cause the arm to be less durable and sustained. One way this can be fixed is by modifying the 2 axis joint to allow the handle to be attached using a single bolt, this will prevent necessary damage to the bearings. Another improvement of the design would account for the singularities in motion as well avoid the chances the handle will experience a gimble lock.

7.2 Control

The robot is able to execute its actions effectively, though there are jitters during different points of execution. This issue can be resolved through more accurate tuning of the PID controller. Ideally there would be a GUI from which the user can modify specific gains to change the torque, speed, and feedback of the arm. This will allow for a more customized and effective training session for the patient. This can also be a source of providing resistance and increase the level difficult of the treatment to match the patient's ability to move.

7.3 Rehabilitation Game

Currently the game developed for the rehabilitation exercise is minimal. The game could be developed more and an improved GUI could me made so that patients can better interact with it using the robotic arm. The way the arm interfaces with the game could also use improvement. Ideally there would a system that allows for custom mapping of different joints to different actions in the game or a game that uses all 3 degrees of motion could be developed, but due to time constraints this was left undone.

7.4 Dynamic Motion Primitives

DMPs are an excellent way for the arm to lead a patient through different trajectories. In order for the arm to carry out the trajectories, inverse kinematics are employed. Currently the inverse kinematics results in an unnatural trajectory to be created. An improvement to the system would be to ensure that the inverse kinematics is produce a natural trajectory when it has constrained joint angles.

Chapter 8

Contributions

8.1 Matthew Bowers

The majority of the work that I contributed to the project was the development of Three Degrees of Freedom Human Interface Handle with Vibrational Feedback Support. I also worked on previous designs of the wrist mechanism as well as various adapter plates. Designed the final 3DOF handle including CAD of 3D printed parts plus selection of vibrational motors, rubber grips, and rotational bearings based on design requirements. Assembled the 3DOF Handle components as well as parts of the arm itself. I also had a role in testing and debugging the hardware and electronics of the arm. I did research and wrote about several of the related systems. Also the initial creation of the Report using Latex through Overleaf and was editor of the progress report.

8.2 Nathaniel Goldfarb

The majority of the work that I contributed to the project was the development of the software stack and the controllers. I wrote developed and wrote most of the software that was used on the robotic platform. I also reconfigured existing code for the new platform. This involved removing the hid communication code from Bowler Studio and writing a hid server to handle the communication to the board without the Bowler Studio over head. I also had to write a UDP bridge between Java and Python so that Python could be used to develop the control architecture. I then developed the organization of the high level code. I wrote several different pieces of code to efficiently pass information around to obtain calculate the control inputs and the current state of the system. I also wrote a real time plotting system to show the state the robot in real time. I also bridged the gap between all the layers of code from hardware to the GUI. This also includes fixing the game integration to the hardware. I had to rewrite how the game was integrated into the system so that it would work with out code base.
I also tested and debugged the hardware, electronics, software of the arm. The electronic of

the arm are very sensitive and required additional hardware to get working reliably. This also required the tunning of the low level PID and the joint offsets. The high level PID also had to be tuned as well as the entire code base tested and debugged. This involved both working on the high level code and low level code to ensure that the arm had the desired movement. I was able to get the entire system working together. I could send commands generated from DMP to move the arm and able to generate compliance control commands to play a game.
I also designed and printed a prototype of a wrist mechanism. Then advised on a the second design iteration. I also aided in the printing of the arm and the installation of the electronics and sensors. To get the DMP for the arm, I arranged the times to use the Mocap system. I also integrate a vibrating motor into the handle.
One of my major responsibilities the management of the team. I arranged meeting times for the group to work on and discuss the project. A big part of this was teaching and training the team on various robotic and programming concepts. I also wrote the sections of the paper pertaining to my contribution.

8.3 Aishwary Jagetia

I have researched about the rehabilitation for stroke patients and the ways to recovery. Based on the power requirements and certain important modifications we had to remodel the entire design. My work was focused on the Hardware side of the project. I learnt Solidworks software for modeling the new arm. I modeled all the required components for our project. Converting soft files from stl's to Solidworks compatible files was a very time consuming task. To name a few specific contributions, I replicated the original arm model from scratch while also magnifying the size to 1.5 times the original. I modified the lower link to increase the angle at the lower joint of the arm. I made alterations in the brackets at every joint to accommodate for double motor for power requirements. I also designed the track and base for the new haptic arm. I made minor adjustments in the end effector joint as the bracket was quite thin earlier and it broke during assembly. I made sure the new design for more flexible in the sense that it could accommodate for varying load cells (For both 50kg and 10kg) as we had to do last minute change in the load cell. Since, I was responsible for the modeling of the parts, I governed the 3D printing for the all the parts of the arms including track and end effectors which was cumbersome but easy task. Not only did I model the parts, I took charge for the physical assembly of the arm.
I also helped Rishi Khajuriwala in creating a new template for upper body using MOCAP. Being the test subject I also learned about the MOCAP system and its working. Finally I took care of the entire Hardware/Mechanical failures with its debugging and played a small role in electronics and code implementation.

8.4 Rishi Khajuriwala

The project goal for me was to learn different robotics concepts and implement them in the work, first task assigned me was to find the rehabilitation exercises that are needed for upper body rehab, so I worked on that and got the exercises and decided which can be implemented on our project. Second task assigned to me was to learn the mocap system and know how it works and then get trajectories from the motion capture. In the mocap system I made an upper body plug in gait template from scratch and used that in getting the trajectory. Then I was given the task to help Nishant with the Pong game, so I worked with him and learned a little about developing a simple 2-D game in python. Then I was given the task to get the PD controller values in MATLAB, I was not able to do that without Nathaniel helping me understand the control and dynamics concept. I helped in tuning the PID values at low level and so helped for degugging the software of the robot. I helped in assembling the robot to Aishwary by assembling the parts and soldering some of the electrical parts and also helped in testing and debugging the hardware and software of the robot. From the project I have learned many new things like Mocap system, control and dynamics of the robot, PD tuning, and also little about creating a simple 2-D game.

8.5 Akshay Kumar

My contribution to this project spread over several domains. I started with exploring the Nucleo board, its operational features and firmware but had to left that midway as it was extraneous and we already had the database ready for it. I explored the various existing end-effector rehabilitation devices and created a data-sheet comparing the existing Hephaestus arm with the likes of Novint Falcon and Geomagic Touch. This data sheet helped ascertain the features and limitations of the existing arm that we expand upon. I worked out the workspace of the Hephaestus arm in 3D, to visualize the out of reach zones and ensure that our new design considers that. I worked on the theoretical aspects of Gravity compensation for the robotic manipulator

I worked with Rishi Khajuriwala to use data from the MOCAP system. I extracted the trajectories for the left and right wrist markers and using Dynamic Movement Primitives to generalize and mathematically formulate the trajectories of motion for the end-effector. I wrote the communication section for the Pong game to interface it with the other Python codes that run the various robot associated scripts like for dynamics and controls. I also worked with Aishwary Jagetia and others while assembling and debugging both the arms.

8.6 Brandon Lam

My main focus on this project were the hardware components of this project. I had worked heavily on understand the different sensors, motors,and different components that were in the original reference arm.This involved creating a data-sheet, with Akshay, that documented the parts that were used and their specifications. From there I worked with other members of the team to ensure the parts we had would work for our application and if they did not then I worked on acquiring them, by handling the orders for materials. I also worked on coming up with a suitable design for our end effector as well as the building and rebuilding of our arm as needed. I had worked with Aishwary to ensure that the newly printed parts would work with our electrical components, debug any designs, prepare and print the parts on the 3D printer, assemble the arms, and make last minute modifications.

While my focus was on the hardware, it was important for me to understand the other areas of the project, even if it was in only in a broad general sense. I played a part in helping create the template for the motion capture by being a subject and determining all the different difficulties that come with collecting usable data, with Rishi. I had worked on the control scheme for the controllers and looked into which would be suitable for the arm and helped in the development of the controller, with Nathaniel. While not an official role, my desire to have an understanding in every area helped keep the team on track, since I would often ask questions or make connects between different fields to ensure that the different areas of our project would be compatible with one another.

I had also taken up the role of editing, rewriting, and formatting the final report. This involved ensuring that every aspect of the project was covered sufficiently and that it would make sense for the reader. Latex was not the easiest to work with and learn, but it did provide a good learning experience.

8.7 Nishant Shah

I tried to help in every domain possible. My main aim of the project was to get familiar with some of the robotic concepts and in that Nathaniel helped me a lot, he helped me understand some of the basic concepts of Kinematics and Dynamics of the robot. I was able to use my knowledge from my Dynamics course and help derive the properties of the robot such as inertia, centripetal and gravity matrix of the robot. Then I helped Nathaniel to write program for the 3D link representation of the robot which helped us to visualize and test the dynamics, control and movement of the robot.

I was assigned to search an appropriate game and I came across some of the programming softwares specifically dedicated to make Haptic Games such as Chai3D and H3D which are being used by Novint, Geomagic and many other companies. But those softwares were for developing a 3D game which would become hard for me to develop a game using those softwares. So, I decided to write the Pong game program in Processing Python Mode but we were not able to import the Processing Python game file into our main programming environment and Akshay helped me out with this problem.

I made some contributions in MOCAP part as well by volunteering as a test subject two times and helped other team members in soldering and assembling the robot. Also, I helped Nathaniel in debugging the software and tune the PID for the robot.

Bibliography

[1] D. Accoto, L. Zollo, D. Formica, and E. Guglielmelli. Design of a planar robotic machine for telerehabilitation of elderly patients. *Gerontechnology*, 7(2), 2008.

[2] D Ali and J Moore. Rehab measures: Fugl-meyer assessment of motor recovery after stroke, 2010.

[3] Charles G Burgar, Susan L Garber, HF Machiel Van der Loos PhD, OTR Deborah Kenney MS, et al. Robot-assisted upper-limb therapy in acute rehabilitation setting following stroke: Department of veterans affairs multisite clinical trial. *Journal of rehabilitation research and development*, 48(4):445, 2011.

[4] Universal Serial Bus. Device class definition for human interface devices (hid). *Version*, 1:1996–2001, 2001.

[5] Radhika Chemuturi, Farshid Amirabdollahian, and Kerstin Dautenhahn. Adaptive training algorithm for robot-assisted upper-arm rehabilitation, applicable to individualised and therapeutic human-robot interaction. *Journal of NeuroEngineering and Rehabilitation*, 10(1):102, Sep 2013.

[6] Leonard E Kahn, Michele L Zygman, William Rymer, and David J Reinkensmeyer. Robot-assisted reaching exercise promotes arm movement recovery in chronic hemiparetic stroke: a randomized controlled pilot study. 3:12, 02 2006.

[7] gary rowe. hid4java.

[8] Kevin Harrington. Rbe3001nucleofirmware, 2017.

[9] Irene Hartigan. A comparative review of the katz adl and the barthel index in assessing the activities of daily living of older people. *International journal of older people nursing*, 2(3):204–212, 2007.

[10] N. Hogan, H. I. Krebs, J. Charnnarong, P. Srikrishna, and A. Sharon. Mit-manus: a workstation for manual therapy and training. i. In *[1992] Proceedings IEEE International Workshop on Robot and Human Communication*, pages 161–165, Sep 1992.

[11] Scott E Kasner. Clinical interpretation and use of stroke scales. *The Lancet Neurology*, 5(7):603–612, 2006.

[12] Hermano I Krebs, Mark Ferraro, Stephen P Buerger, Miranda J Newbery, Antonio Makiyama, Michael Sandmann, Daniel Lynch, Bruce T Volpe, and Neville Hogan. Rehabilitation robotics: pilot trial of a spatial extension for mit-manus. *Journal of NeuroEngineering and Rehabilitation*, 1(1):5, 2004.

[13] Hermano Igo Krebs, James Celestino, Dustin Williams, Mark Ferraro, Bruce Volpe, and Neville Hogan. *24 A Wrist Extension for MIT-MANUS*, pages 377–390. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[14] Kate Laver, Stacey George, Susie Thomas, Judith E Deutsch, and Maria Crotty. Virtual reality for stroke rehabilitation. *Stroke*, 43(2):e20–e21, 2012.

[15] Peter S Lum, Charles G Burgar, Machiel Van der Loos, Peggy C Shor, et al. Mime robotic device for upper-limb neurorehabilitation in subacute stroke subjects: A follow-up study. *Journal of rehabilitation research and development*, 43(5):631, 2006.

[16] Hugh Markus. Stroke: causes and clinical features. *Medicine*, 40(9):484–489, 2012.

[17] Stefano Masiero and Mario Armani. Upper-limb robot-assisted therapy in rehabilitation of acute stroke patients: focused review and results of new randomized controlled trial. *Journal of rehabilitation research and development*, 48(4):355, 2011.

[18] Richard M Murray, Zexiang Li, S Shankar Sastry, and S Shankara Sastry. *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[19] J. Oblak, I. Cikajlo, and Z. Matjacic. Universal haptic drive: A robot for arm and wrist rehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(3):293–302, June 2010.

[20] J. C. Perry, J. Rosen, and S. Burns. Upper-limb powered exoskeleton design. *IEEE/ASME Transactions on Mechatronics*, 12(4):408–417, Aug 2007.

[21] Giulio Rosati, Paolo Gallina, Stefano Masiero, and Aldo Rossi. Design of a new 5 dof wire-based robot for rehabilitation. In *Rehabilitation Robotics, 2005. ICORR 2005. 9th International Conference on*, pages 430–433. IEEE, 2005.

[22] R. Sanchez, D. Reinkensmeyer, P. Shah, J. Liu, S. Rao, R. Smith, S. Cramer, T. Rahman, and J. Bobrow. Monitoring functional arm movement for home-based therapy after stroke. In *The 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, volume 2, pages 4787–4790, Sept 2004.

[23] Gustavo Saposnik, Robert Teasell, Muhammad Mamdani, Judith Hall, William McIlroy, Donna Cheung, Kevin E Thorpe, Leonardo G Cohen, Mark Bayley, et al. Effectiveness of virtual reality using wii gaming technology in stroke rehabilitation. *Stroke*, 41(7):1477–1484, 2010.

[24] Mark W Spong, Seth Hutchinson, and Mathukumalli Vidyasagar. *Robot modeling and control*, volume 3. Wiley New York, 2006.

[25] A. Stuart. Arm and hand exercises for stroke rehab. 2012.

[26] Richard Q Van der Linde, Piet Lammertse, Erwin Frederiksen, and B Ruiter. The hapticmaster, a new high-performance haptic interface. In *Proc. Eurohaptics*, pages 1–5, 2002.

[27] JC Van Swieten, PJ Koudstaal, MC Visser, HJA Schouten, and J Van Gijn. Interobserver agreement for the assessment of handicap in stroke patients. *Stroke*, 19(5):604–607, 1988.

[28] John E Ware Jr and Cathy Donald Sherbourne. The mos 36-item short-form health survey (sf-36): I. conceptual framework and item selection. *Medical care*, pages 473–483, 1992.

[29] Sung H You, Sung Ho Jang, Yun-Hee Kim, Mark Hallett, Sang Ho Ahn, Yong-Hyun Kwon, Joong Hwi Kim, and Mi Young Lee. Virtual reality–induced cortical reorganization and associated locomotor recovery in chronic stroke. *Stroke*, 36(6):1166–1171, 2005.

[30] W. Yu and J. Rosen. A novel linear pid controller for an upper limb exoskeleton. In *49th IEEE Conference on Decision and Control (CDC)*, pages 3548–3553, Dec 2010.

[31] L. Zollo, D. Accoto, F. Torchiani, D. Formica, and E. Guglielmelli. Design of a planar robotic machine for neuro-rehabilitation. In *2008 IEEE International Conference on Robotics and Automation*, pages 2031–2036, May 2008.