ENPM-662

Introduction to Robot Modeling

Final Project – Reach Robot

(Shoulder – Elbow Rehabilitation Robot)

Rajeshwar N S

116921237

Table of contents

Abstract:

Due to raised incidences of stroke, paralysis, or other diseases along with dramatic increment of life expectancy, the number of patients with movement disability has been increasing continuously. Repetitive and intensive voluntary movements in physical therapy are important factors that facilitate significant improvement for motor-impaired patients. The emergence of rehabilitation robotic devices has stimulated the development of physical therapy. Here the REACH Robot is a rehabilitation/assistive device that is used to aid patients who are not able to use their shoulders and elbows to an extent a perfectly healthy individual does. The goal of the project is to model the human arm as RRP manipulator observing its planar motion when coupled with end effector. Also, the project discusses about the interaction between the patient's digits (fingers) and robot's gripper during simulation.

Motivation:

A lot of college going teenagers often meet with accidents, injuring their shoulders and elbows. So, this project focuses on the population and the anthropological data of lengths of parts like elbows and shoulders are taken from [1] in which every link (ie. Elbow or shoulder) length is calculated as a multiple of height. Generally, Rehabilitation for Shoulder muscles contain two parts. a. Flexibility and b. Strength. Flexibility generally includes Range of motion, stretching exercises which are static and dynamic. ROM is desired degrees of range of motion of movement pattern that is pain free and maybe assisted with a stick/ towel or unassisted. Static stretching are stretches that are pain free and held for 20-30 seconds and repeated 3-6 reps. Dynamic stretching are stretches that are pain free and held for 1-5 seconds and repeated 10-15 reps. The Strength aspect comprises of lifting weights, tubing etc. This project focusses more on the flexibility over the strength ie. providing stretching exercises to the arm.

Robot Description:

The REACH robot consists of a simple hand gripper that is constrained to move along a single axis (Y). The gripper is linearly actuated by a roh'lix motor. The upper and lower bounds (Y_f and $Y_{o)}$ for the gripper movement are constrained by design. The distance between the Human subject and the lower bound Y_o is D1, which is fixed.

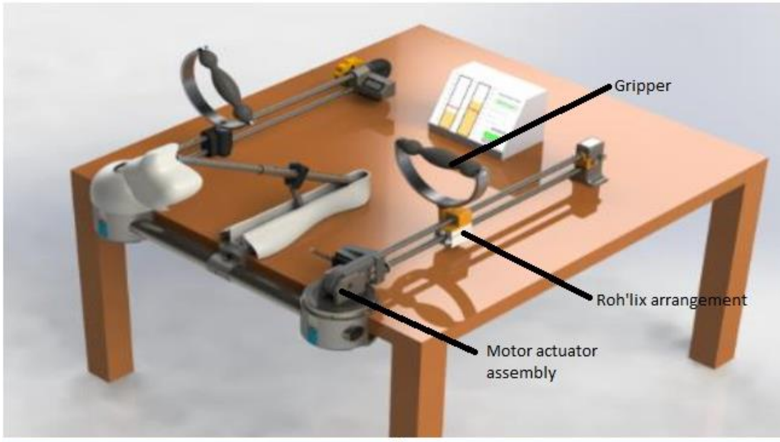


Fig 1: Reach Robot

Problem statement:

For the given chosen desired path y(t), Compute the required actuator force profile $F_a(t)$ required to power the robot end effector to aid in assisting the patient. Also, model the human arm as a 7-DOF machine and show its interaction with the robot, which eventually constrains the arm to move along the same plane as the robot's base. Verify that the obtained joint angles during the simulation matches with the analytically solved angles using Inverse Kinematics. Further, simulate the force of interaction between human digits and the robot end effector, based on the different instantaneous acceleration values obtained for the gripper.

Assumptions:

- The Human wrist is connected to the robot end effector, which moves only along the Y axis. Only yaw is allowed for the human wrist.
- Shoulder and elbow are modelled as torsional springs with torsional stiffness Ks=100Nm/rad and Ke=60Nm/rad. Values taken from Journal of Neurosciences.
- Link lengths L1 and L2 are constant for a given patient and will be assumed as a constant during the simulation stage.
- The entire human arm lies on the XY plane.
- Both the shoulder and elbow angles are bounded as follows.

$$\theta_S \epsilon\ [0,120],\ \theta_E \epsilon\ (0,180)$$

- For the gripper shown in the model, the gripper is moving in an orthogonal direction to the normal forces at the tips of digits when connected to the gripper. So, this negates the problem of slippage for our consideration.

Approach to Performing the task:

- Compute the D-H parameters using appropriate frame diagrams.
- Obtain the transformation matrices from forward kinematics and calculate the Jacobian matrix of the manipulator.
- Using Inverse Kinematics, calculate the joint angles ($\theta_{S,}\theta_{e,}\theta_{w,}$) by geometry as shown in MATLAB code.
- Model the manipulator in V-Rep using the given link lengths and the obtained joint variables.
- Simulate the model with all joints being in Inverse Kinematics mode.
- Validate the joint angles obtained during the simulation process with the analytically solved values.
- On Obtaining the joint angles, find the Rotational force ie. Torque at each joint using torsional stiffness values.
- Calculate the force required to actuate the gripper handle using Jacobian matrix and Joint angle values. Study the interaction between the Human digits and end effector and understand the gripping action of human hand on the end effector for different values of actuator force.

Obtaining the Data:

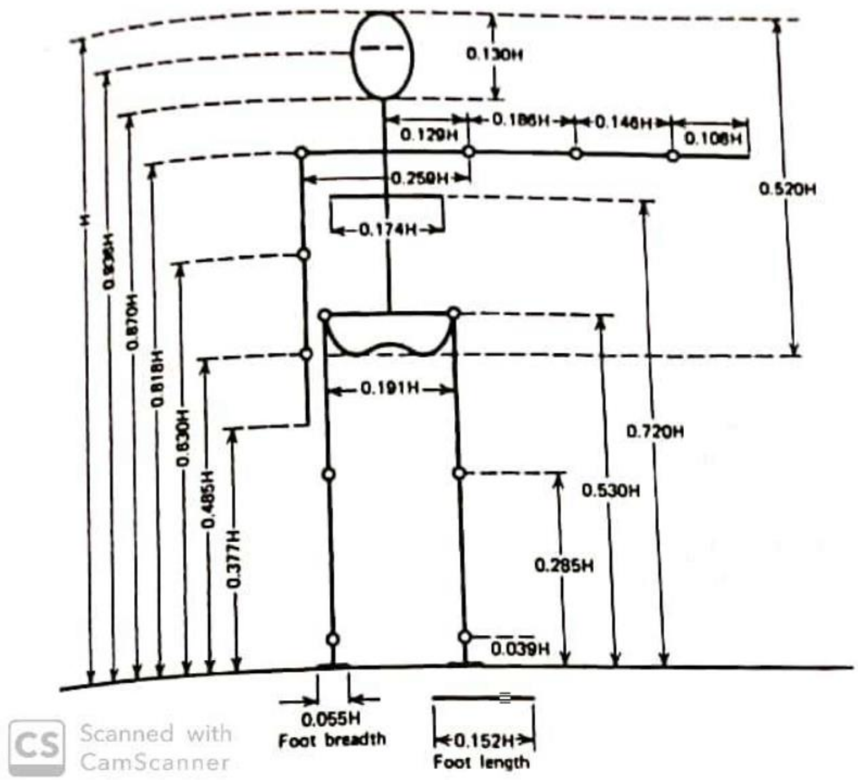


Fig 2: Anthropological Data of Average Human parts

First, the anthropological data of the Elbow and shoulder lengths [1] are taken as shown in the Fig.2. The average adult human stands 171cm. Thus, H = 1.71m. Using this we calculate the link lengths as :

Human body dimensions	In metres
Mid Riff to Shoulder	0.129H = 0.22059
Shoulder to Elbow	0.186H = 0.31806
Elbow to Wrist	0.146H = 0.24966
Wrist to Fingers	0.108H = 0.18468

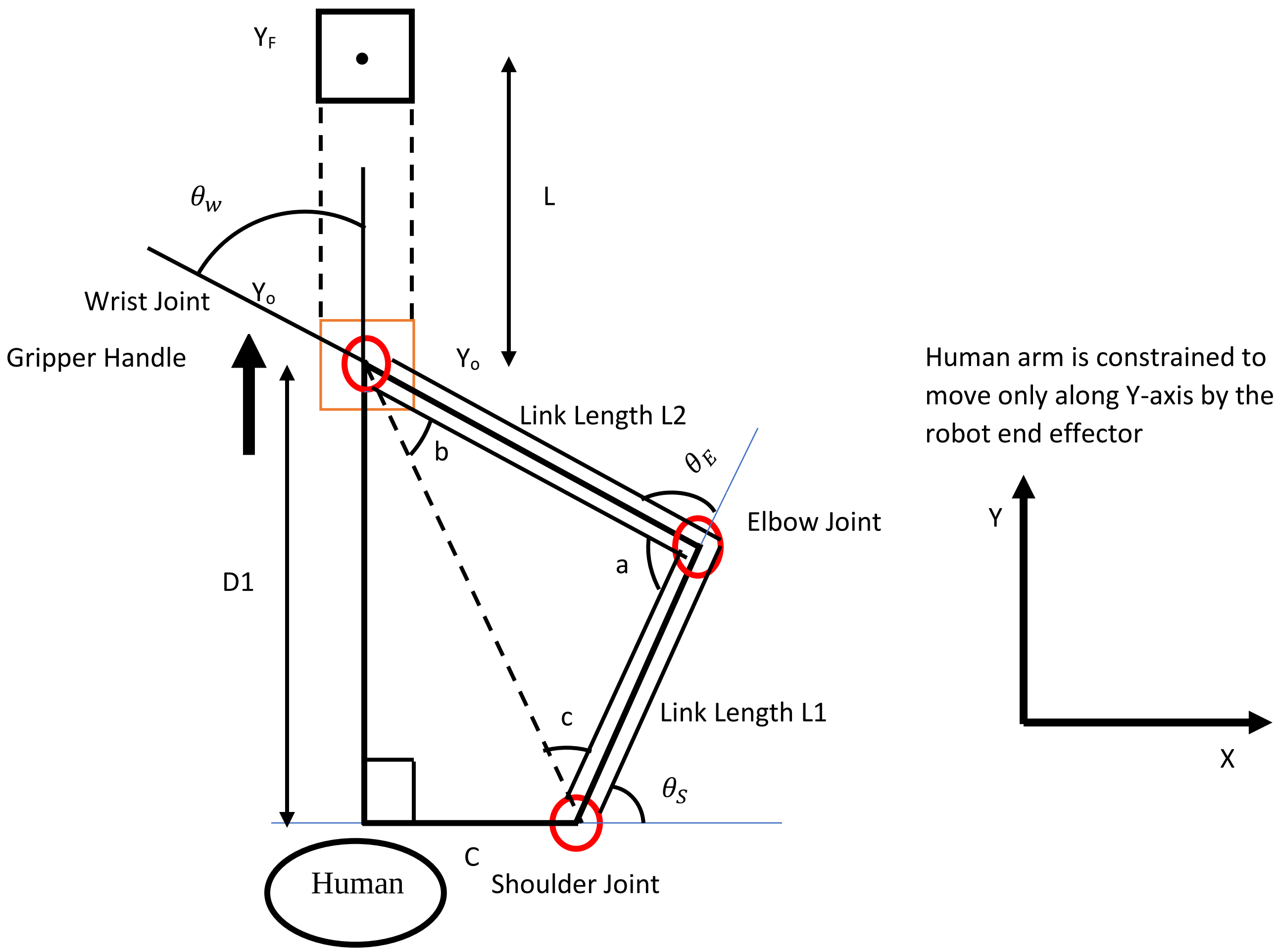


Fig 3. Reach Robot as a RRRP Manipulator

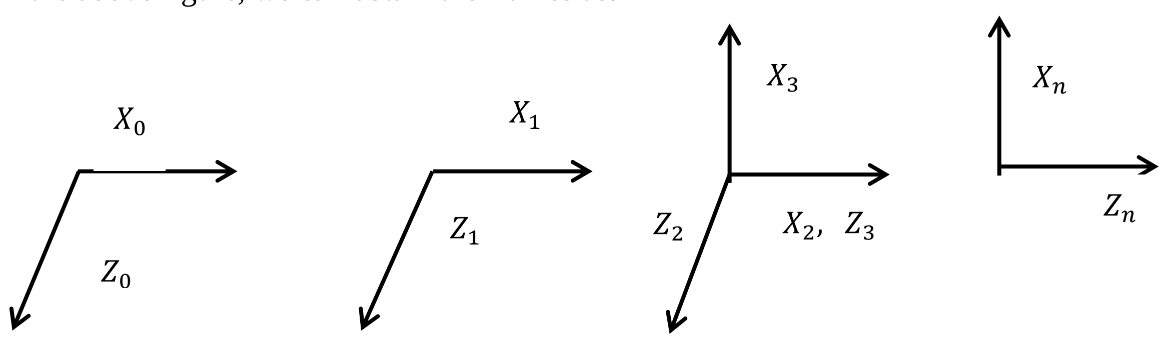
Also, the distance between the human and the gripper handle (D1), the distance between midriff to the shoulder joint (C) and the distance traversed by the gripper handle in the Y-plane (L) are given [2]. Given these values, we find the initial orientation of the Elbow and shoulder links (θ_S and θ_E) by geometry. Applying cosine rule, we obtain the initial orientation of both the links. The calculation of the joint angle values is shown in the attached MATLAB code.

```
l1 = 0.31806; %Shoulder-Elbow portion's link length
l2 = 0.24966; %Elbow-Wrist portion's link length

% D is the distance between the Human midriff and
% the REACH robot's lower boundary.
D = 0.35; %in m

% Distance between midriff and shoulder joint
C = 0.22059; %in m

% Expanding on the cosine rule mentioned in the report to find
% the angles given the lengths of the triangle, constituted
% by the links L1, L2 and hypotenuse of the C and D components.
A = sqrt(C^2 + D^2); % hypotenuse

% angle opposite hypotenuse =>
a = acosd((l1^2 + l2^2 - A^2)/(2*l1*l2));

% angle opposite shoulder-elbow part =>
b = acosd((A^2 + l2^2 - l1^2)/(2*A*l2));

% angle opposite elbow-wrist part =>
c = acosd((A^2 + l1^2 - l2^2)/(2*A*l1));
```

```
% computing the initial joint angles (Shoulder, Elbow, Wrist)
% for the gripper position at lower boundary condition
theta_e = 180 - a; %elbow joint angle - initial
theta_s = 180 - (atan(D/C)+c); %shoulder joint angle - initial
theta_w = atan2d(C,D)+b; %wrist joint angle - initial
```

D-H Frames/Tables:

From the above figure, we can obtain the frames as:



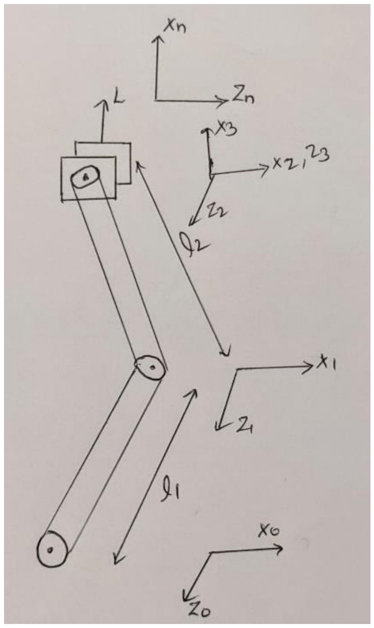


Fig 4: D-H Frame Representation

The DH table could be constructed as:

$\boldsymbol{\theta_{i-1}}$	$\boldsymbol{d_{i-1}}$	$\boldsymbol{x_{i-1}}$	$\boldsymbol{\alpha_{i-1}}$
θ_S	0	L1	0
θ_E	0	L2	0
$\theta_w + 90$	0	0	90
0	L	0	0

<u>Forward Kinematics and Jacobian :</u>

The Transformation matrices can be calculated as shown in the attached MATLAB code.

```
function T = rots(theta, d, a, alpha)
T=[cosd(theta) -sind(theta)*cosd(alpha) sind(theta)*sind(alpha) a*cosd(theta); sind(theta) cosd(theta)*cosd(alpha) -cosd(theta)*sind(alpha)
    a*sind(theta); 0 sind(alpha) cosd(alpha) d; 0 0 0 1];
end
```

```
A1_0=rots(theta_s1,0,l1,0); %First row of DH table
A2_0=rots(theta_e1,0,l2,0); %Second row of DH table
A3_0=rots(theta_w1+90,0,0,90); %Third row of DH table
A4_0=rots(0,D1-D,0,0); %Fourth row of DH table

% Corresponding transformation matrices
H1_0=A1_0; %For Shoulder (revolute) joint
H2_0=H1_0*A2_0; %For Elbow (revolute) joint
H3_0=H2_0*A3_0; %For Wrist (revolute) joint
H4_0=H3_0*A4_0; %For gripper (prismatic) actuator

% Extracting R and O matrices to obtain Jacobian matrix
O0=[0 0 0]'; % Origin coordinates
R1=H1_0(1:3,1:3);
O1=H1_0(1:3,4);
R2=H2_0(1:3,1:3);
O2=H2_0(1:3,4);
R3=H3_0(1:3,1:3);
O3=H3_0(1:3,4);
R4=H4_0(1:3,1:3);
O4=H4_0(1:3,4);

% Calculating the values of Z for Jacobian matrix
Z0=eye(3)*[0 0 1]';
Z1=R1*[0 0 1]';
Z2=R2*[0 0 1]';
Z3=R3*[0 0 1]';
```

The final transformation matrix H4 is given by

H4

4×4 double

	1	2	3	4
1	-0.5153	0	-0.8570	-0.3663
2	-0.8570	0	0.5153	0.6077
3	0	1.0000	0	0
4	0	0	0	1.0000

From these rotation matrices, Calculate Z and O matrices to compute the Jacobian of the manipulator as shown below:

```
% Computing the Jacobian Matrix componenents and framing J
J11=cross(Z0,(O4-O0)); %Jv component of Shoulder jt
J12=cross(Z1,(O4-O1)); %Jv component of Elbow jt
J13=cross(Z2,(O4-O2)); %Jv component of Wrist jt
J14=Z3; %Jv component of Gripper prismatic jt
J21=Z0; %Jw component of Shoulder jt
J22=Z1; %Jw component of Elbow jt
J23=Z2; %Jw component of Wrist jt
J24=[0 0 0]'; %Jw component of Gripper prismatic jt
J=[J11 J12 J13 J14;J21 J22 J23 J24]; %Jacobian matrix created for the same.
```

The resultant Jacobian matrix is

J ×

6×4 double

	1	2	3	4
1	-0.6077	-0.3050	-0.0876	-0.8570
2	-0.3663	-0.2686	-0.1457	0.5153
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	1.0000	1.0000	1.0000	0

Inverse Kinetics:

- Torque:

Using Jacobian matrix, compute the torque of each revolute joints using the formula

$$\boxed{\tau = K.\theta}$$

```
% Now, we compute the values for joint torques, using the
% joint angles and joint tortional stiffness values. The torsional
% stiffness values for the same are given as follows:
K_s=100; %Torsional stiffness for shoulder = 100Nm/rad.
K_e=60; %Torsional stiffness for elbow = 60Nm/rad.
K_w=40; %Torsional stiffness for wrist = 40Nm/rad.
K_gripper=50; %Stiffness for gripper modeled as a spring = 50N/m
T_shoulder=K_s*deg2rad(theta_s1);
T_elbow=K_e*deg2rad(theta_e1);
T_wrist=K_w*deg2rad(theta_w1);
F_gripper=K_gripper*(D1-D);
```

- Force:

From the obtained torques using the equation

$$\boxed{\tau = J^T.F}$$

```
% Creating the new joint Torque matrix:
T= [T_shoulder;T_elbow;T_wrist;F_gripper];
F=pinv(J')*T;
```

Since the Jacobian matrix obtained is not invertible, we obtain the pseudo inverse of the matrix using pinv function in MATLAB. The Torque and Force matrix are shown below:

T

4×1 double

	1
1	188.2975
2	12.1518
3	20.5884
4	8.5000

F

6×1 double

	1
1	-197.4260
2	-307.8687
3	0
4	0
5	0
6	-72.2413

Validation:

(For validation, please find the attached video)

Validating Part 1:

- For REACH horizontal plane movement, use Modeling_Part1 file.
- Run using CopelliaSimEdu.
- Run the simulation and note the parameter for Gripper Actuator (revolute joint), it says 0.17. That is the higher boundary limit for the gripper, lower being zero.
- Correspondingly, note down the values for wrist joint, elbow joint and shoulder joint. Here, D1=D+0.17
- Verify with the values displayed in MATLAB for the same.
- Tabulating the same here and comparing:

Joint	**Value in MATLAB**	**Value in CopelliaSim**
Shoulder	22.7326	-22.74
Elbow	75.6285	75.64
Wrist	-52.8959	-52.90

- Now, rerun the simulation and try to pause the video at a random value of displacement for gripper actuation, before simulation ends.
- For this point, let us try to change the corresponding D1 value in MATLAB, as obtained in our recording. Here, D1 = D+0.0846.
- Now, obtaining the calculated values from MATLAB for joint angles and comparing with the obtained values from simulation.

Joint	**Value in MATLAB**	**Value in CopelliaSim**
Shoulder	-4.8099	-4.81
Elbow	25.02	25.04
Wrist	-20.21	-20.23

-

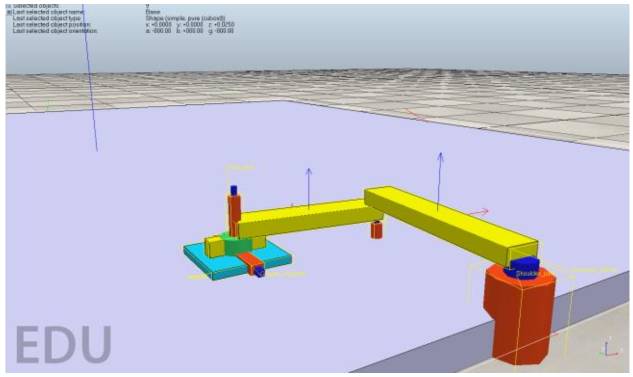
¶ 

Fig 5 – Validation of RRRP Manipulator Joint Variables

Validating Part 2:

¶ The Robot was also able to trace a circular path as shown in the video. This enables increasing the degree of freedom in the vertical plane with minimal jerk. A smooth trajectory was implemented for the same along the vertical plane and a gripper wrist assembly was simulated to follow this path. For Further Integration, we could add the shoulder and elbow joints as multiple revolute joints or individual spherical joints.

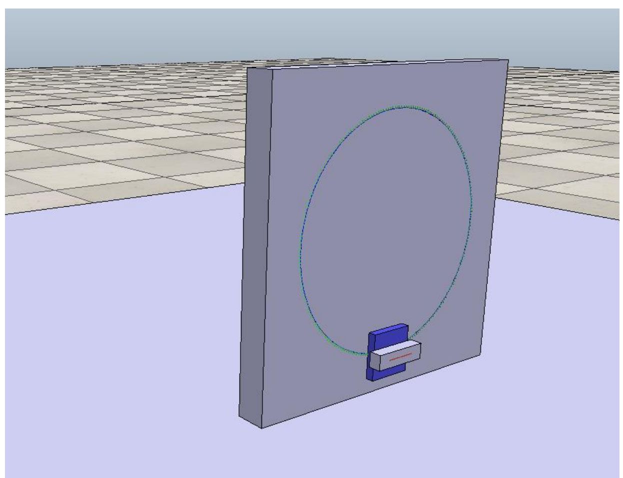


Fig 6 – Validation of movement of the manipulator in the vertical plane

Conclusion:

We have designed, modelled and simulated a 2D planar (horizontal) rehabilitation device, the REACH robot. The goal of this device is to provide shoulder and elbow rehabilitation to the human patient repetitively, over multiple sessions. The goal of using a robot for rehabilitation is to minimise dependency on manpower to carry out repetitive tasks, with additional safety factors taken into consideration. We have successfully validated the joint angles for a simulation of a human arm, interacting with the end effector of the robot. A video has been attached alongside this report, for clarity. We have used the values of joint angles to calculate joint torques (by relating them with torsional stiffness values), which can further be combined with Jacobian for the designed RRRP manipulator (Human arm model) to calculate the end effector wrench. Also, as an extension of our project, we have extended rehabilitation to the third dimension as well, by simulating a condition where rehabilitation is being provided in vertical plane, where we can observe the gripper to move along with the wrist along a predetermined path, on a vertical platform. Looking forward, these are the possible ways we are planning to improve the project:

1. Extending rehabilitation to the vertical plane by modelling the shoulder and elbow joints as spherical joints or a series of revolute joints. We have modelled the simulation environment already (defined the path that the gripper must follow), but the modelling of human hand is yet to be realised due to lack of time.
2. Looking from the control systems perspective, we can possibly improve the interaction of the robot with an end user by using force or impedance control.
3. Integrating with interactive software- to improve the satisfaction of the patient's therapy session. This is a clinically proven result (motivation plays a key role in rehabilitation). Potential areas of work- Computer vision and interfacing REACH robot as the controller for any interactive software/game.

References:

[1] Biomechanics and Motor control of Human movement, D.A. Winters, 4th edition, John Wiley and sons – 2009

[2] Professor Anindo Roy et al, Department of Neuroscience, University of Maryland

[3] Spong, M. W., Hutchinson, S., & Vidyasagar, M. (2006). *Robot modeling and control* (Vol. 3, pp. 187-227). New York: wiley.

[4] Franklin et al, Journal of Neuroscience, Vol – 27, Issue – 29, Page 7705 - 7716