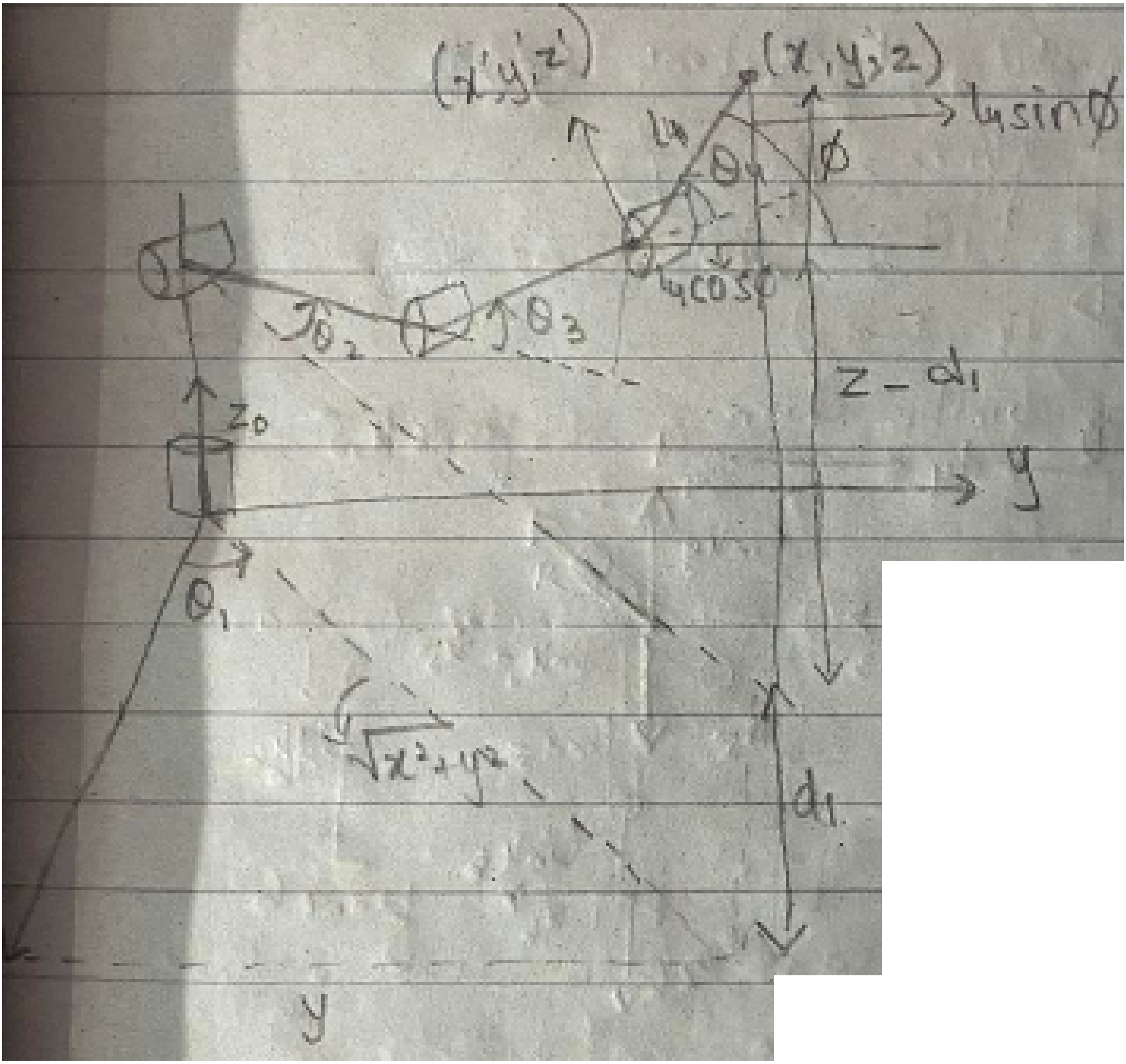
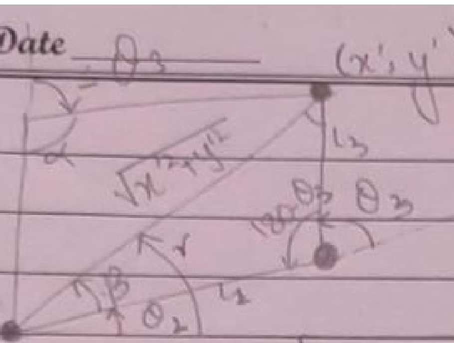
Task 5.1 Inverse Kinematics Solutions (40 points)

Given a desired position, (x, y, z), of the end-effector and orientation, ϕ, find mathematical expressions for all solutions to this inverse kinematics problem. Show all steps and specifically state how many solutions exist? Assuming that direction of $\hat{x}$ of end-effector is along the length of the last link, ϕ is the angle it makes with the x-axis of frame 1, i.e. $\phi = \theta_2 + \theta_3 + \theta_4$. When the gripper is parallel to the base board, then $\phi = 0°$[a].

[a] See the remarks below for further explanation



Date θ_3



50, 50, 20, 0

50, 30, 20, 0

$\theta_1 = \tan^{-1}(y/x)$

$\theta_1 = \tan^{-1}(y/x) - pi$

where $x' = \sqrt{x^2+y^2} - l_4\cos\phi$

$y' = (z - d_1) - l_4 \sin\phi$

using law of cosines,

$$x'^2 + y'^2 = l_2^2 + l_3^2 - 2l_2l_3\cos(\underbrace{180-\theta_3}_{\alpha}) - (i) \quad \because \cos(a-b) = \cos a\cos b + \sin a \sin b$$

$$x'^2 + y'^2 = l_2^2 + l_3^2 - 2l_2l_3\cos 180 \cos\theta_3$$

$$x'^2 + y'^2 = l_2^2 + l_3^2 + 2l_2l_3\cos\theta_3$$

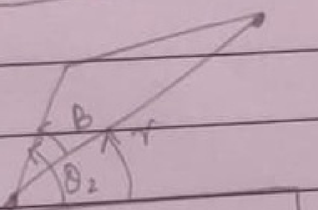
$$\cos\theta_3 = \frac{x'^2 + y'^2 - l_2^2 - l_3^2}{2l_2l_3}$$ $\Rightarrow$ matches the solution in lab.

To get two solutions we get

elbow up

$\alpha - \theta_3 = 180$

$\boxed{\theta_3 = \alpha - 180}$



$\boxed{\theta_2 = \beta + \gamma}$

elbow down

$$\alpha = \cos^{-1}\left(\frac{-(x'^2+y'^2) + l_2^2 + l_3^2}{2l_2l_3}\right)$$

$\alpha + \theta_3 = 180$ $\boxed{\theta_3 = 180 - \alpha}$

$$\beta = \cos^{-1}\left(\overbrace{\frac{x'^2+y'^2+l_2^2-l_3^2}{2l_2\sqrt{x'^2+y'^2}}}^{k}\right)$$

$\beta + \theta_2 = \gamma$ where $\gamma = \tan^{-1}\left(\frac{y'}{x'}\right)$

$\boxed{\theta_2 = \gamma - \beta}$

OR

$\cos\beta = k$ $\quad \sin\beta = \sqrt{1-\cos^2\beta}$

$$\tan\beta = \frac{\sin\beta}{\cos\beta}$$

$$\beta = \tan^{-1}\left(\frac{\sqrt{1-k^2}}{k}\right)$$

Task 5.2	Inverse Kinematics MATLAB function (10 points)

Say there are N possible solutions to the IK problem of our manipulator, in general. Write a MATLAB function `findJointAngles(x,y,z,phi)`, which accepts the position and orientation of end-effector as arguments and returns an $N \times 4$ matrix containing all the IK solutions. Row i of this matrix corresponds to solution i, and column j of the matrix contains the values for θ_j.

```
function solutions = findJointAngles(x,y,z,phi)
theta_1_1 = atan2(y,x);
theta_1_2 = atan2(y,x)-pi;
d_1 = 142;
a_4 = 95;
a_3 = 108;
a_2 = 110;

r = sqrt(x^2 + y^2);
s = z - d_1;
x_new= r - a_4*cos(phi);
y_new= s - a_4*sin(phi);

alpha = acos((-(x_new^2 + y_new^2) + a_2^2 + a_3^2)/(2*a_2*a_3));
k=(a_2^2 + y_new^2 + x_new^2 - a_3^2)/ (2 * a_2 * sqrt(y_new^2 + x_new^2));%the
term in derivation
beta = atan2(sqrt(1 - (k)^2),k);
gamma = atan2(y_new,x_new);

theta_2_1 = gamma - beta;
theta_2_2 = gamma + beta;
theta_3_1 = pi - alpha;
theta_3_2 = - pi + alpha;
theta_4_1 = phi - theta_3_1 - theta_2_1;
theta_4_2 = phi - theta_3_2 - theta_2_2;

solutions = [theta_1_1 theta_2_1 theta_3_1 theta_4_1;
    theta_1_1 theta_2_2 theta_3_2 theta_4_2;
   theta_1_2 pi-theta_2_1 -theta_3_1 -theta_4_1;
   theta_1_2 pi-theta_2_2 -theta_3_2 -theta_4_2];

end
```

Task 5.3 Optimal Solution (30 points)

Write a MATLAB function `findOptimalSolution(x,y,z,phi)`, which accepts the desired position and orientation as arguments and returns a vector `[theta1,theta2,theta3,theta4]` corresponding to the optimal and realizable inverse kinematics solution. Optimal solution is the IK solution closest to the current configuration of the robot, i.e. minimize $b_1|\Delta\theta_1| + b_2|\Delta\theta_2| + b_3|\Delta\theta_3| + b_4|\Delta\theta_4|$. You can choose $b_i = 1$. (See below)

```
%%Task 5.3
solutions=findJointAngles(180,100,50,5) %% first we find the solution by
% providing our own end effector frame co-ordinates through joint angles
% functions written in Task 5.2. This will give our 4 solutions through
% the formula that we have computed
x=arb.getpos()%to get actual solution from the robotics manipulator itself.
%This tells us the vector [theta1 theta2 theta3 theta4] in radians
sum=0; % sum variable that calculates the product
temp=[];% 1x4 vector that stores the sum of solutions
```

```
for i=1:4
    sum=abs(solutions(i,1)-x(1))+ abs(solutions(i,2)-x(2)) + abs(solutions(i,3)-x(3)) + abs(solutions(i,4)-x(4))
    temp(i)=sum;
    sum=0;
end
solution=find(min(temp)) % find the Best solution
arb.setpos([solutions(solution,:) 0],[50,50,50,50,50]) % the best solution
%is then transmitted to get the end effector orientation co-ordinates
%now we measure the difference between the end effector orientation that we
%fed in and the actual one that we measured to get the error
```

Task 5.4	IK Exploration (20 points)

(a) Select five points (x, y, z, ϕ) in the workspace of the robot and execute the optimal solution for each point, as determined by your `findOptimalSolution` function. Measure and note down the achieved point in each case.

(b) Determine the accuracy of your system and compare it to the value obtained in the previous lab. Comment on any possible statistically significant differences.

(c) Is there any (x, y, z, ϕ) in the workspace for which all possible solutions are realizable? Justify.

First (x,y,z,Φ) are selected then a corresponding solution of 4 by 4 matrix is extracted from which the optimal solution is extracted. Once the optimal solution is extracted then the ar.setpos([vector of angles],[vector of speed for each joint]) is given.

The following MATLAB code has 4 different points and the achieved points as measured ones. Then their accuracy is determined.

```
clc;clear;
real=[50,50,20,0; 10,30,20,0 ; 10 30 5 0 ; 20 30 5 0 ;20 70 5 0];
measured=[30,60,28,0 ; 3,40,15,0; 11 26 2 0; 15 23 1 0 ; 10 84 3 0];
accuracy=[0 0 0 0];
for i=1:4
 accuracy(i)=sqrt((real(i,1)-measured(i,1))^2 + (real(i,2)-measured(i,2))^2
+(real(i,3)-measured(i,3))^2 + (real(i,4)-measured(i,4))^2)
end
```

Output:

```
  23.7487   13.1909    5.0990    9.4868
```

(c) No there is no workspace where all possible solution are realizable