AuE 8220: Robotic Mobility and Manipulation, Fall 2022 Đomework #1: Fourbar Position Analyses (Method 1, Method III and Numerical). Assigned on: September 6th 2022 Due: September 13th 2022, 1:00 PM

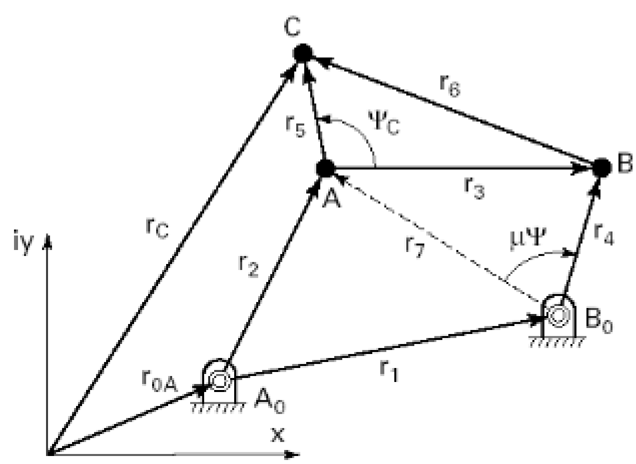


Figure 2: Nomenclature for the fourbar mechanism

		Ground Link	Input Link	Coupler	Follower	Coupler Point of Interest
Lengths	$r_{0A} = 0$	$r_1 = 4.0$	$r_2 = 2.0$	$r_3 = 3.0$	$r_4 = 6.0$	$r_5 = 4$
Angles	$\theta_0 = 0°$	$\theta_1 = 30°$	θ_2	θ_3	θ_4	Included Angle) $\psi_c = 30°$

Problem 1A – Method of Intersecting Circles

Following steps will help you to solve a given four-bar position analysis problem.

Step 1: Identify what type of mechanism your given problem is – crank-rocker, double rockers, etc.

Step 2: Try sketch the four-bar with the informations that are given to you.

Step 3: Know (approximately) where your solution(s) might be.

Step 4: Set-up the two Equations of circles.

Step 5: Find the two intersection points, the corresponding angles, determine the crossed/ uncrossed configurations.

Step 6: Check if your solution is reasonable.

Note: It is NOT unusual that you may need to repeat your calculation several times to get the correct answer if you use a calculator to perform the numerical calculation! Be patient.

Since $r_3 + r_1 < r_2 + r_4$, It is a double rocker mechanism

We now examine the development of the algebraic method for intersection of two circles:

Since $\vec{r}_1 = (r_1 \cos\theta_1, r_1 \sin\theta_1) = (D_x, D_y)$

and

$r_2 = (r_2 \cos\theta_2, r_2 \sin\theta_2) = (A_x, A_y)$

Are known, we can calculate the location of the points

This method of analysis involves finding the expressions for two circles, calculating the intersection of the two which then provide the solutions.

The equation of a circle of radius (r_3) centered about the point $A\ (A_x, A_y)$ may be written as:

$$\boxed{(B_x - A_x)^2 + (B_y - A_y)^2 = r_3^2} \quad (2)$$

where (B_x, B_y) is any point on the locus.

Similarly the equation of a circle of radius (r_4) centered at the point $D\ (D_x, D_y)$ may be written as:

$$\boxed{(B_x - D_x)^2 + (B_y - D_y)^2 = r_4^2} \quad (3)$$

Expanding equations 2 and 3, and then subtracting we get,

$$\cancel{B_x^2} + A_x^2 - (2A_x)B_x + \cancel{B_y^2} + A_y^2 - (2A_y)B_y = r_3^2$$
$$-\ \cancel{B_x^2} + D_x^2 - (2D_x)B_x + \cancel{B_y^2} + D_y^2 - (2D_y)B_y = r_4^2$$

$$(A_x^2 + A_y^2) - (D_x^2 + D_y^2) + 2(D_x - A_x)B_x + 2(D_y - A_y)B_y = r_3^2 - r_4^2 \quad — (4)$$

$$\Rightarrow \quad B_x = \frac{r_3^2 - r_4^2 + r_1^2 - r_2^2}{2(D_x - A_x)} + \frac{2(A_y - D_y)}{2(D_x - A_x)} B_y.$$

$$B_x = K_1 + K_2 B_y \quad — (5)$$

Substituting 5 in 3 we get,

Substituting (5) into (3) we get:

$$(K_1 + K_2 B_y - D_x)^2 + (B_y - D_y)^2 = r_4^2 \quad — (6)$$

If we let $K_1 - D_x = K_3$. Eqn. 6 may be written a

$$(K_2 B_y + K_3)^2 + (B_y - D_y)^2 = r_4^2 \quad — (7).$$

$$(K_2^2 + 1)B_y^2 + (2K_2K_3 - 2D_y)B_y + (K_3^2 + D_y^2 - r_4^2) = 0$$

$$\boxed{(P)B_y^2 + Q B_y + R = 0} \quad — (8).$$

$$(B_y)_{1,2} = \frac{-Q \pm \sqrt{Q^2 - 4PR}}{2P} \quad - (9)$$

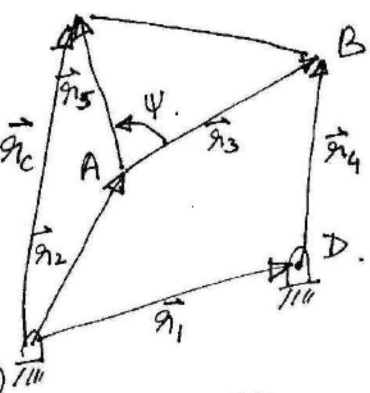
Substituting (9) back into (5) we get

$$\boxed{(B_x)_{1,2} = K_1 + K_2 (B_y)_{1,2}} \quad - (10)$$

To get feasible solutions,

$Q^2 - 4PR >= 0$

Selecting one of the branches corresponding to the (+) root of $\sqrt{Q^2 - 4PR}$ in Eqn. (9) we get.



$$\theta_{31} = \text{atan2}((By_1 - Ay), (Bx_1 - Ax))$$
$$\theta_{41} = \text{atan2}((By_1 - Dy), (Bx_1 - Dx)) \quad - (11)$$

$$\Rightarrow \theta_{51} = \theta_{31} + \Psi \quad - (12)$$

$$\vec{r}_c = \vec{r}_2 + \vec{r}_5 \quad - (13)$$
$$= (r_2 \cos\theta_2 \hat{i} + r_2 \sin\theta_2 \hat{j}) + (r_5 \cos\theta_5 \hat{i} + r_5 \sin\theta_5 \hat{j})$$
$$= \underbrace{\boxed{(A_x + r_5 \cos(\theta_5))}}_{C_x} \hat{i} + \underbrace{\boxed{(A_y + r_5 \sin\theta_5)}}_{C_y} \hat{j}$$

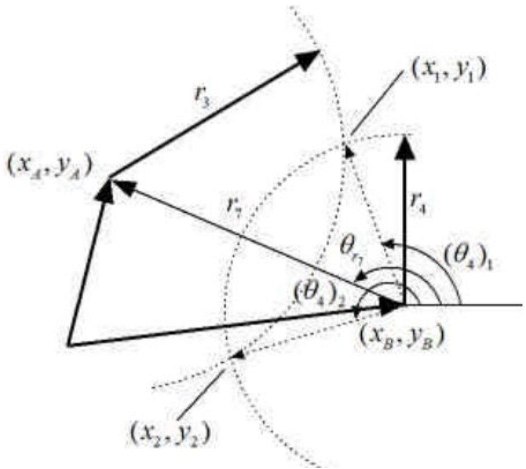


Figure B

Determining crossed / uncrossed configuration

At this point, we still not yet determine whether the two points $(x_i, y_i), i = 1, 2$ corresponding to crossed/ uncrossed configuration. To determine this, we need to use the vector r_7, which in this particular case, vector r_7 originate from the origin of r_4, and pointing towards the tip of r_2. The corresponding angle of vector r_7 is θ_{r_7}. Hence, if $(\theta_4)_1 > \theta_{r_7}$, the corresponding configuration is crossed configuration; if this is the case, then $(\theta_4)_2 < \theta_{r_7}$, the corresponding configuration is uncrossed configuration. See Figure B.

Reference Code:

```
clc
close all
clear all

% Given parameters (finding theta2, theta3):
r1=5;   t1=30*pi/180;
r2=4.5;
r3=3.0;
r4=2.5; t4=60*pi/180;

% Find the center & radius of the two circles:
xA=0; yA=0; rA=4.5;
xB=r1*cos(t1)+r4*cos(t4);
yB=r1*sin(t1)+r4*sin(t4);
rB=3;

% Find the two intersecting points using the function you have created in
% Homework #00 (given centers points, radius, find two intersecting points):
[xpts, ypts]= IntersectCircle(xA,yA,rA,xB,yB,rB);

% So, the first intersecting point is:
x_1=xpts(1,1);
y_1=ypts(1,1);
% The second intersecting point is:
x_2=xpts(1,2);
y_2=ypts(1,2);

% So, the two theta2 (t2_1, t2_2) can be found using:
t2_1 = atan2((y_1-yA),(x_1-xA));
t2_2 = atan2((y_2-yA),(x_2-xA));
% The two theta3 (t3_1, t3_2) can be found using:
t3_1 = atan2((yB-y_1),(xB-x_1));
t3_2 = atan2((yB-y_2),(xB-x_2));
```

Modify the values of r1, r2, r3, r4 as per the given problem and analyze graph plots.

```
% But which angles corresponding to crossed/uncrossed configuration?
% We need to construct the r7 vector (start from root of link4 (Bx,By)pointing to
% tip of link2 (Ax,Ay)) to check for crossed/ uncrossed configuration:
Ax=r2*cos(t2_1); Ay=r2*sin(t2_1);
Bx=r1*cos(t1);   By=r1*sin(t1);
t7_1=atan2(Ay-By,Ax-Bx);

% Make t3s, t2s and t7s all begins from horizontal x-axis (angle convention adopted
% in MAE412)Here again you see the usefulness of using atan2() function previously:
if (t2_1<0) t2_1 = t2_1 + 2*pi; end
if (t2_2<0) t2_2 = t2_2 + 2*pi; end
if (t3_1<0) t3_1 = t3_1 + 2*pi; end
if (t3_2<0) t3_2 = t3_2 + 2*pi; end
if (t7_1<0) t7_1 = t7_1 + 2*pi; end

% Now we can make the selection (For single points, this might appear redundant, but if
% we need to detemine a range of angle, for example, theta4 is from 0-360 deg, you can
% easily convert the following code into creating series of points:
if (t4 < t7_1) %then t2_1, t3_1 are uncrossed configs
   t2_uncrossed = t2_1;
   t3_uncrossed = t3_1
   t2_crossed = t2_2;
   t3_crossed = t3_2;
else           % else, t2_2, t3_2 are uncrossed configs
   t2_uncrossed = t2_2;
   t3_uncrossed = t3_2;
   t2_crossed = t2_1;
   t3_crossed = t3_1;
end

% Now that we have everything we need, we can start plotting our result!
fig=figure(1)
% For illustrative purpose, let us also create some points to plot the two
% circles that we used in this method (Method of intersecting circles):
t=0:0.1:2.05*pi;        %create theta values
circleAx=xA+rA*cos(t);
circleAy=yA+rA*sin(t);
circleBx=xB+rB*cos(t);
circleBy=yB+rB*sin(t);
plot(circleAx, circleAy,':b'); hold on %Plot the two circles.
plot(circleBx, circleBy,':b');

% Similarly, let also plot the two intersecting points:
plot(x_1,y_1,'*k','linewidth',2);
plot(x_2,y_2,'*r','linewidth',2);
```

```
% Now, let us create some data to plot the fourbar mechanism (Well, these lines of
% codes may not be necessary...but is easier to understand for everyone):

% For the uncrossed Configuration:
t2=t2_uncrossed;
t3=t3_uncrossed;

Link1x = [0 r1*cos(t1)];        % Data pts for Link 1
Link1y = [0 r1*sin(t1)];
Link2x = [0 r2*cos(t2)];        % Data pts for Link 2
Link2y = [0 r2*sin(t2)];
Link3x = [r2*cos(t2) r2*cos(t2)+r3*cos(t3)];    % Data pts for Link 3
Link3y = [r2*sin(t2) r2*sin(t2)+r3*sin(t3)];
Link4x = [r1*cos(t1) r1*cos(t1)+r4*cos(t4)];    % Data pts for Link 4
Link4y = [r1*sin(t1) r1*sin(t1)+r4*sin(t4)];
```

```
% Similarly, for the crossed Configuration (Lets not plot now):
% t2=t2_crossed;
% t3=t3_crossed;
%
% Link1xc = [0 r1*cos(t1)];
% Link1yc = [0 r1*sin(t1)];
% Link2xc = [0 r2*cos(t2)];
% Link2yc = [0 r2*sin(t2)];
% Link3xc = [r2*cos(t2) r2*cos(t2)+r3*cos(t3)];
% Link3yc = [r2*sin(t2) r2*sin(t2)+r3*sin(t3)];
% Link4xc = [r1*cos(t1) r1*cos(t1)+r4*cos(t4)];
% Link4yc = [r1*sin(t1) r1*sin(t1)+r4*sin(t4)];

% Plot the four bar (uncrossed configs)!
h1=plot(Link1x, Link1y,'k','LineWidth',4);
h2=plot(Link2x, Link2y,'r','LineWidth',4);
h3=plot(Link3x, Link3y,'b','LineWidth',4);
h4=plot(Link4x, Link4y,'g','LineWidth',4);
h=[h1,h2,h3,h4]; %create this array of handles, use for legend.
legend(h,'link 1','link 2','link 3','link 4',2);
% plot(Link1xc, Link1yc,'k','LineWidth',2);
% plot(Link2xc, Link2yc,'r','LineWidth',2);
% plot(Link3xc, Link3yc,':b','LineWidth',2);
% plot(Link4xc, Link4yc,':r','LineWidth',2);

% Let us also put some information on the plot:
text(5,-4,'\theta_2 = ','fontWeight','bold','fontSize',12);
text(5,-5,'\theta_3 = ','fontWeight','bold','fontSize',12);
text(6.5,-4,num2str(t2*180/pi),'fontWeight','bold','fontSize',10);
text(6.5,-5,num2str(t3*180/pi),'fontWeight','bold','fontSize',10);
title('Position Analysis using Method I: Intersecting of circles', ...
      'fontWeight','bold','fontSize',10);
xlabel('x position','fontWeight','bold','fontSize',12);
ylabel('y position','fontWeight','bold','fontSize',12);
grid on;
axis square
```

As you can see, once you have your derivation of the equation ready, programming it is fairly easy. Once you have this program written, you can use it to find θ_2, θ_3 for any other values given. But since the program does not consider considering if the given four-bar parameters are feasible, type of mechanism (crank-rocker, double rocker, etc), it has limited use.

You can now modify the program for achieving the required plots and vary the value of Θ_2 from 0-360. (Hint: Think of using for loop)

Problem 1B

The analytical solution procedure follows the same major steps as in the graphical solution. That is, a position analysis must first be performed, then a velocity analysis, and finally the acceleration analysis. The position analysis, for a closed-loop linkage, comprises the solution of the closure equations for the joint angles or link orientations. Once this solution is obtained, the velocity and acceleration states are quickly obtainable using the differentiated equations. It will be seen, however, that the position analysis, which is so easily performed graphically by construction of a drawing to scale, is a complex matter when performed analytically.

For all of the simple mechanisms that we will consider initially, the first step in solving the position equations is to identify the variable to be determined first. When the position equations involve two angles as unknowns, the solution procedure is to isolate the trigonometric function involving the angle to be eliminated on the left-hand side of the equation. In order to eliminate θ_3, first isolate it on one side of Eqs. (3.26) and (3.27) as follows:

$$r_3 \cos\theta_3 = r_1 \cos\theta_1 + r_4 \cos\theta_4 - r_2 \cos\theta_2 \tag{3.28}$$

$$r_3 \sin\theta_3 = r_1 \sin\theta_1 + r_4 \sin\theta_4 - r_2 \sin\theta_2 \tag{3.29}$$

Notice that the angle θ_1 is a known constant. Now square both sides of both equations, add, and simplify the result using the trigonometric identity $sin^2\,\theta + cos^2\,\theta = 1$. This gives

$$\begin{aligned} r_3^2 = r_1^2 + r_2^2 + r_4^2 + 2r_1r_4(\cos\theta_1\cos\theta_4 + \sin\theta_1\sin\theta_4) \\ -2r_1r_2(\cos\theta_1\cos\theta_2 + \sin\theta_1\sin\theta_2) - 2r_2r_4(\cos\theta_2\cos\theta_4 + \sin\theta_2\sin\theta_4) \end{aligned} \tag{3.30}$$

Equation (3.30) gives θ_4 in terms of the given angle θ_2 (and the constant angle θ_1) but not explicitly. To obtain an explicit expression, simplify Eq. (3.30) by combining the coefficients of $\cos\theta_4$ and $\sin\theta_4$ as follows:

$$A\cos\theta_4 + B\sin\theta_4 + C = 0 \tag{3.31}$$

where

$$\left.\begin{aligned} A &= 2r_1r_4\cos\theta_1 - 2r_2r_4\cos\theta_2 \\ B &= 2r_1r_4\sin\theta_1 - 2r_2r_4\sin\theta_2 \\ C &= r_1^2 + r_2^2 + r_4^2 - r_3^2 - 2r_1r_2(\cos\theta_1\cos\theta_2 + \sin\theta_1\sin\theta_2) \end{aligned}\right\} \tag{3.32}$$

To solve Eq. (3.31), use the standard trigonometric identities for half-angles given in the following:

$$\sin\theta_4 = \frac{2\tan(\theta_4/2)}{1 + \tan^2(\theta_4/2)} \tag{3.33}$$

$$\cos\theta_4 = \frac{1 - \tan^2(\theta_4/2)}{1 + \tan^2(\theta_4/2)} \tag{3.34}$$

After substitution and simplification, we get

$$(C - A)t^2 + 2Bt + (A + C) = 0$$

where

$$t = \tan\left(\frac{\theta_4}{2}\right)$$

Solving for t gives

$$t = \frac{-2B + \sigma\sqrt{4B^2 - 4(C - A)(C + A)}}{2(C - A)} = \frac{-B + \sigma\sqrt{B^2 - C^2 + A^2}}{C - A} \tag{3.35}$$

and

$$\theta_4 = 2\tan^{-1} t \tag{3.36}$$

where $\sigma = \pm 1$ is a sign variable identifying the assembly mode. Note that $\tan^{-1} t$ has a valid range of $-\pi/2 \leq \tan^{-1} t \leq \pi/2$. Therefore, θ_4 will have the range $-\pi \leq \theta_4 \leq \pi$. Unless the linkage is a Grashof type II linkage in one of the extreme positions of its motion range, there are two solutions for θ_4 corresponding to the two values of σ, and they are both valid. These correspond to two assembly modes or branches for the linkage. Once we pick the value for σ corresponding to the desired mode, the sign in an actual linkage stays the same for any value of θ_2.

Because of the square root in Eq. (3.35), the variable t can be complex $(A^2 + B^2) < C^2$. If this happens, the mechanism cannot be assembled in the position specified. The assembly configurations would then appear as shown in Fig. 3.6.

Equations (3.28) and (3.29) can now be solved for θ_3. Dividing Eq. (3.29) by Eq. (3.28) and solving for θ_3 gives

$$\theta_3 = \tan^{-1}\left[\frac{r_1 \sin\theta_1 + r_4 \sin\theta_4 - r_2 \sin\theta_2}{r_1 \cos\theta_1 + r_4 \cos\theta_4 - r_2 \cos\theta_2}\right] \tag{3.37}$$

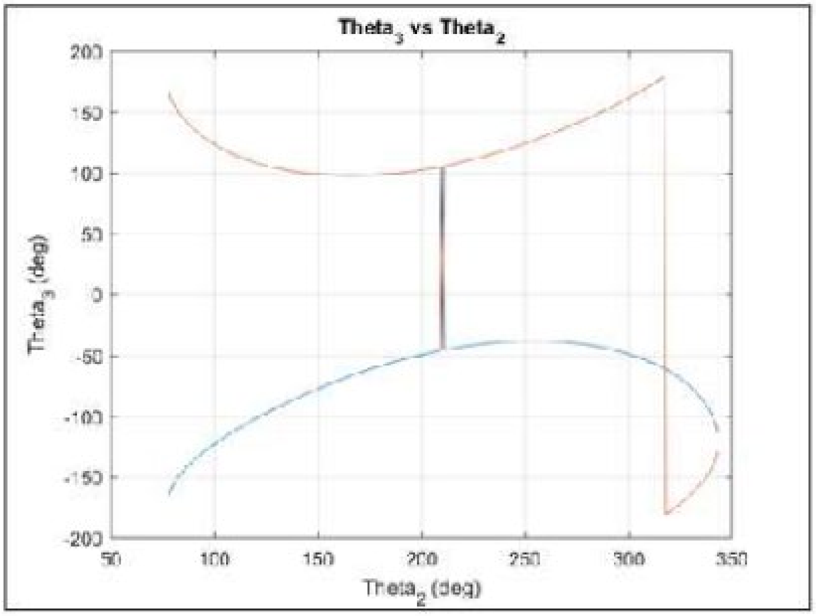
Note that in Eq. (3.37), it is essential that the sign of the numerator and denominator be maintained to determine the quadrant in which the angle θ_3 lies. This can be done directly by using the ATAN2 function. The form of this function is

$$\text{ATAN2}(\sin\theta_3, \cos\theta_3) = \tan^{-1}\left[\frac{\sin\theta_3}{\cos\theta_3}\right] \tag{3.38}$$

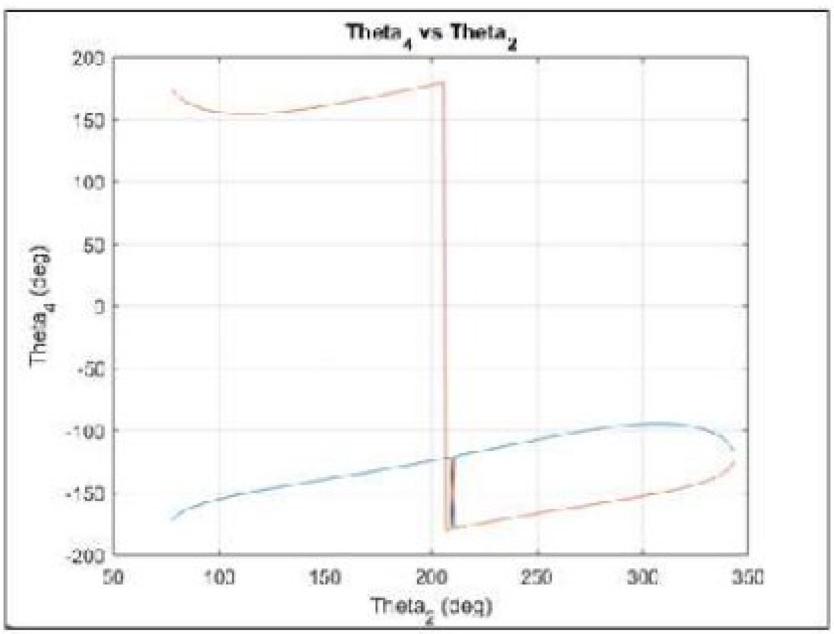
Equations (3.35)–(3.37) give a complete and consistent solution to the position problem. As indicated before, for any value of θ_2, there are typically two values of θ_3 and θ_4, given by substituting $\sigma = +1$ and -1, respectively, in Eq. (3.35). These two different solutions are shown in Fig. 3.7. The two solutions correspond to an assembly ambiguity that also appears in the graphical construction.

```
clear; clc;
r1 = 4;
r2 = 2;
r3 = 3;
r4 = 6;
theta_1 = 30; % First link angle
theta_2 = 0:1:360; %Second link angle
sigma = 1;
A = 2*r1*r4*cosd(theta_1) - 2*r2*r4*cosd(theta_2); %Coefficients
B = 2*r1*r4*sind(theta_1) - 2*r2*r4*sind(theta_2);
C = (r1^2 + r2^2 + r4^2 - r3^2).*ones(size(theta_2)) - (2*r1*r2).*(cosd(theta_1).*cosd(theta_2) + sind(theta_1).*sind(theta_2));
Sqrt_value = B.^2 - C.^2 + A.^2;
x = find(Sqrt_value<0);
theta_2NaN = theta_2(x);
theta_4 = 2.*(atan2d(-B + sigma.*(sqrt(B.^2 - C.^2 + A.^2)) , C - A));
```

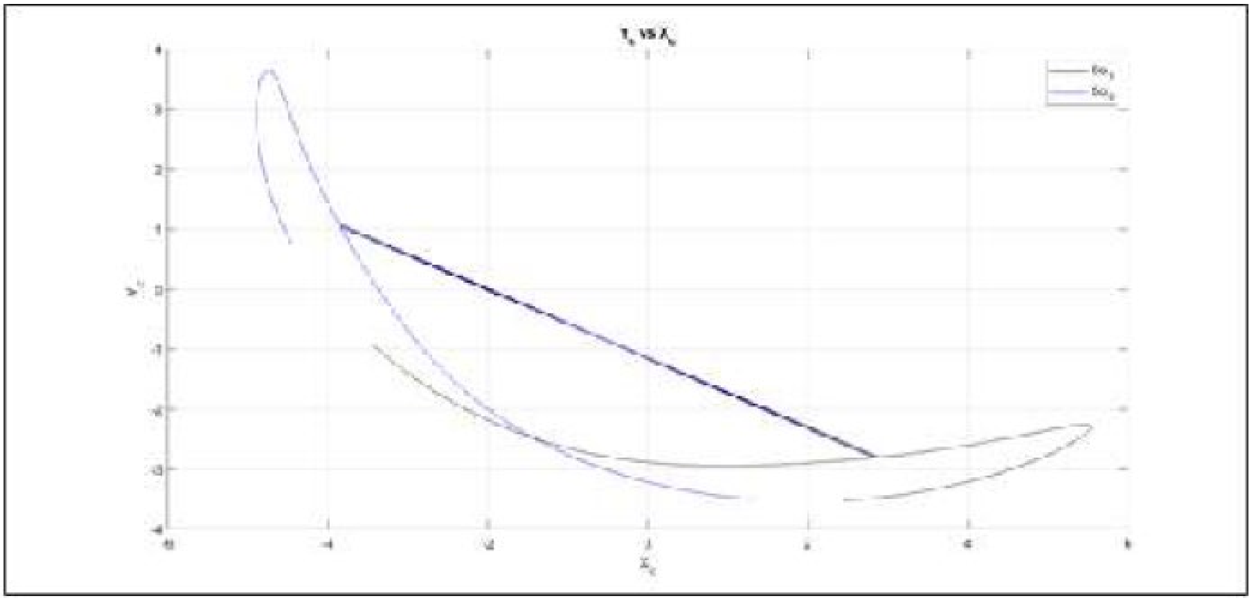
All solutions show both crossed and uncrossed configurations to be used as a reference.



Theta$_3$ vs Theta$_2$
200
150
100
50
0
-50
-100
-150
-200
50
100
150
200
250
300
350
Theta$_3$ (deg)
Theta$_2$ (deg)



Theta$_4$ vs Theta$_2$
200
150
100
50
0
-50
-100
-150
-200
50
100
150
200
250
300
350
Theta$_4$ (deg)
Theta$_2$ (deg)



Problem 1C

i) **Newton-Raphson Method**

Formulate the problem as:

$$\bar{F}(\Theta)=\begin{bmatrix}F_1\\F_2\end{bmatrix}=\begin{bmatrix}r_2\cos\theta_2+r_3\cos\theta_3-r_4\cos\theta_4-r_1\cos\theta_1\\r_2\sin\theta_2+r_3\sin\theta_3-r_4\sin\theta_4-r_1\sin\theta_1\end{bmatrix}=\underset{\sim}{0} \qquad (3)$$

$\Theta=\begin{bmatrix}\theta_3\\\theta_4\end{bmatrix}$, taking the Taylor series expansion about $\Theta=\Theta_k+\Delta\Theta$:

$$\bar{F}(\Theta_k+\Delta\Theta)=\bar{F}(\Theta_k)+\left[\frac{\partial\bar{F}}{\partial\Theta_k}\right]_{\Theta_k}\Delta\Theta+Higher\ Order\ Terms \qquad (4)$$

We want to find the roots (solutions of Θ) when is $\bar{F}=\underset{\sim}{0}$, hence let $\bar{F}(\Theta_k+\Delta\Theta)=\underset{\sim}{0}$. Ignoring the higher order terms, Eq(4) reduces to:

$$\begin{aligned}&\bar{F}(\Theta_k)+\left[\frac{\partial\bar{F}}{\partial\Theta_k}\right]_{\Theta_k}\Delta\Theta=\underset{\sim}{0}\\&\Rightarrow\Delta\Theta=-\left[\frac{\partial\bar{F}}{\partial\Theta_k}\right]_{\Theta_k}^{-1}\bar{F}(\Theta_k)\end{aligned} \qquad (5)$$

So, given an initial guess Θ_k, we can update Θ_k by $\Theta_{k+1}=\Theta_k+\Delta\Theta$.

1.3 Solution obtained when…

There are several ways to stop the iteration process:

(1) $\bar{F}(\Theta_k+\Delta\Theta)=\underset{\sim}{0}$: $\bar{F}(\Theta_k+\Delta\Theta)=\bar{F}(\Theta_{k+1})<Tolerance$
(2) $\Theta_{k+1}-\Theta_k=\Delta\Theta$: $abs(\Theta_{k+1}-\Theta_k)=abs(\Delta\Theta)<Tolerance$
(3) Number of iterations.

1.4 Example.

Let us use a four-bar problem as a example to illustrate the Newton-Raphson method:

$$\left[\frac{\partial\bar{F}}{\partial\Theta_k}\right]=\begin{bmatrix}\dfrac{\partial f_1}{\partial\theta_3}&\dfrac{\partial f_1}{\partial\theta_4}\\\dfrac{\partial f_2}{\partial\theta_3}&\dfrac{\partial f_2}{\partial\theta_4}\end{bmatrix}=\begin{bmatrix}-r_3\sin\theta_3&r_4\sin\theta_4\\r_3\cos\theta_3&-r_4\cos\theta_4\end{bmatrix} \qquad (6)$$

For checking crossed or uncrossed configurations, choose the initial theta_solutions that would represent the first crossed or uncrossed configurations and then apply numerical method from that point.
Note: Plots could vary based on solver configurations and the initial values of thetas chosen. Make sure you mention your initial conditions in the report.

Code – Newton Raphson

```
function [theta_uncrossed, theta_crossed] = Fourbar_Pos_NR_GivenT2(L, theta)
A = 2*L(1)*L(4)*cos(theta(1,:)) - 2*L(2)*L(4)*cos(theta(2,:));
B = 2*L(1)*L(4)*sin(theta(1,:)) - 2*L(2)*L(4)*sin(theta(2,:));
C = L(1)^2 + L(2)^2 + L(4)^2 - L(3)^2 - 2*L(1)*L(2)*(cos(theta(1,:)).*...
cos(theta(2,:)) + sin(theta(1,:)).*sin(theta(2,:)));
delta = A.^2 + B.^2 - C.^2;
theta2_filtered = [];
theta1_filtered = [];
k = 1;
for i = 1:length(theta(2,:))
if delta(i) >= 0
theta2_filtered(k) = theta(2, i);
theta1_filtered(k) = theta(1, i);
k = k+1;
else
k = k;
end
end
theta_uncrossed = zeros(2, length(theta2_filtered));
theta_crossed = zeros(2, length(theta2_filtered));
for i = 1:length(theta2_filtered)
theta3 = 0;
theta4 = -pi/2;
k = 0;
while (k < 20)
F1 = L(2)*cos(theta2_filtered(i)) + L(3)*cos(theta3) - L(4)*cos(theta4) ...
- L(1)*cos(theta1_filtered(i));
F2 = L(2)*sin(theta2_filtered(i)) + L(3)*sin(theta3) - L(4)*sin(theta4) ...
- L(1)*sin(theta1_filtered(i));
F_d = [-L(3)*sin(theta3), L(4)*sin(theta4); L(3)*cos(theta3), ...
-L(4)*cos(theta4)];
THETA = -inv(F_d)*[F1; F2];
theta3 = theta3 + THETA(1);
theta4 = theta4 + THETA(2);
k = k + 1;
```

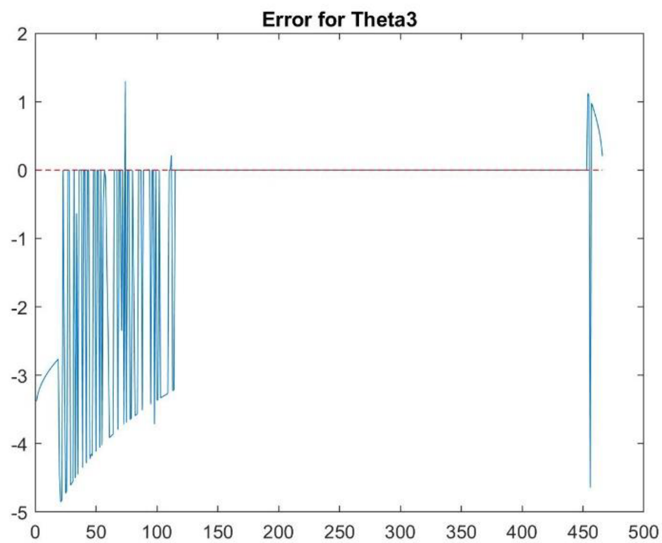
```
end
theta3 = atan2(sin(theta3), cos(theta3));
theta4 = atan2(sin(theta4), cos(theta4));
theta_uncrossed(:,i) = [theta3; theta4];
end
for i = 1:length(theta2_filtered)
theta3 = pi/2;
theta4 = pi;
k = 0;
while (k < 20)
F1 = L(2)*cos(theta2_filtered(i)) + L(3)*cos(theta3) - L(4)*cos(theta4) ...
- L(1)*cos(theta1_filtered(i));
F2 = L(2)*sin(theta2_filtered(i)) + L(3)*sin(theta3) - L(4)*sin(theta4) ...
- L(1)*sin(theta1_filtered(i));
F_d = [-L(3)*sin(theta3), L(4)*sin(theta4); L(3)*cos(theta3), ...
-L(4)*cos(theta4)];
THETA = -inv(F_d)*[F1; F2];
theta3 = theta3 + THETA(1);
theta4 = theta4 + THETA(2);
k = k + 1;
end
theta3 = atan2(sin(theta3), cos(theta3));
theta4 = atan2(sin(theta4), cos(theta4));
theta_crossed(:,i) = [theta3; theta4];
end
          end
```

Reference Code – Fsolve Method

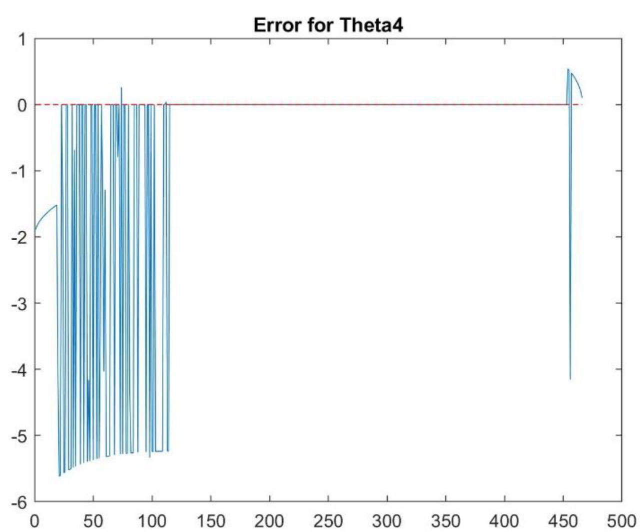
```
function [theta_uncrossed, theta_crossed] = Fourbar_Pos_FSOLVE_GivenT2(L,
theta)
global theta1_fi theta2_fi
A = 2*L(1)*L(4)*cos(theta(1,:)) - 2*L(2)*L(4)*cos(theta(2,:));
B = 2*L(1)*L(4)*sin(theta(1,:)) - 2*L(2)*L(4)*sin(theta(2,:));
C = L(1)^2 + L(2)^2 + L(4)^2 - L(3)^2 - 2*L(1)*L(2)*(cos(theta(1,:)).*...
cos(theta(2,:)) + sin(theta(1,:)).*sin(theta(2,:)));
delta = A.^2 + B.^2 - C.^2;
theta2_filtered = [];
theta1_filtered = [];
k = 1;
for i = 1:length(theta(2,:))
if delta(i) >= 0
theta2_filtered(k) = theta(2, i);
theta1_filtered(k) = theta(1, i);
k = k+1;
else
k = k;
end
end
theta_uncrossed = zeros(2, length(theta2_filtered));
theta_crossed = zeros(2, length(theta2_filtered));
for i = 1:length(theta2_filtered)
theta1_fi = theta1_filtered(i);
theta2_fi = theta2_filtered(i);
fun = @root2d;
theta0 = [0, -pi/2];
```

```
theta_s = fsolve(fun,theta0);
theta3 = atan2(sin(theta_s(1)), cos(theta_s(1)));
theta4 = atan2(sin(theta_s(2)), cos(theta_s(2)));
theta_uncrossed(:,i) = [theta3; theta4];
end
for i = 1:length(theta2_filtered)
theta1_fi = theta1_filtered(i);
theta2_fi = theta2_filtered(i);
fun = @root2d;
theta0 = [pi/2, pi];
theta_s = fsolve(fun,theta0);
theta3 = atan2(sin(theta_s(1)), cos(theta_s(1)));
theta4 = atan2(sin(theta_s(2)), cos(theta_s(2)));
theta_crossed(:,i) = [theta3; theta4];
end
```

```
end
function F = root2d(theta)
global L theta2_fi theta1_fi
F(1) = L(2)*cos(theta2_fi) + L(3)*cos(theta(1)) - L(4)*cos(theta(2)) ...
- L(1)*cos(theta1_fi);
F(2) = L(2)*sin(theta2_fi) + L(3)*sin(theta(1)) - L(4)*sin(theta(2)) ...
- L(1)*sin(theta1_fi);
```



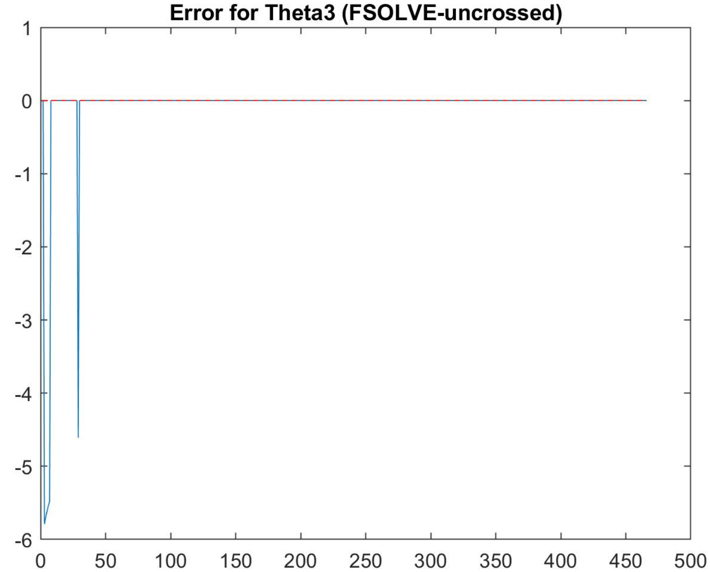
Newton-Raphson Method



Newton-Raphson Method

The spikes are due to discontinuity points that can cause the solver to diverge.

Error for Theta3 (FSOLVE-uncrossed)



Error for Theta4 (FSOLVE-uncrossed)