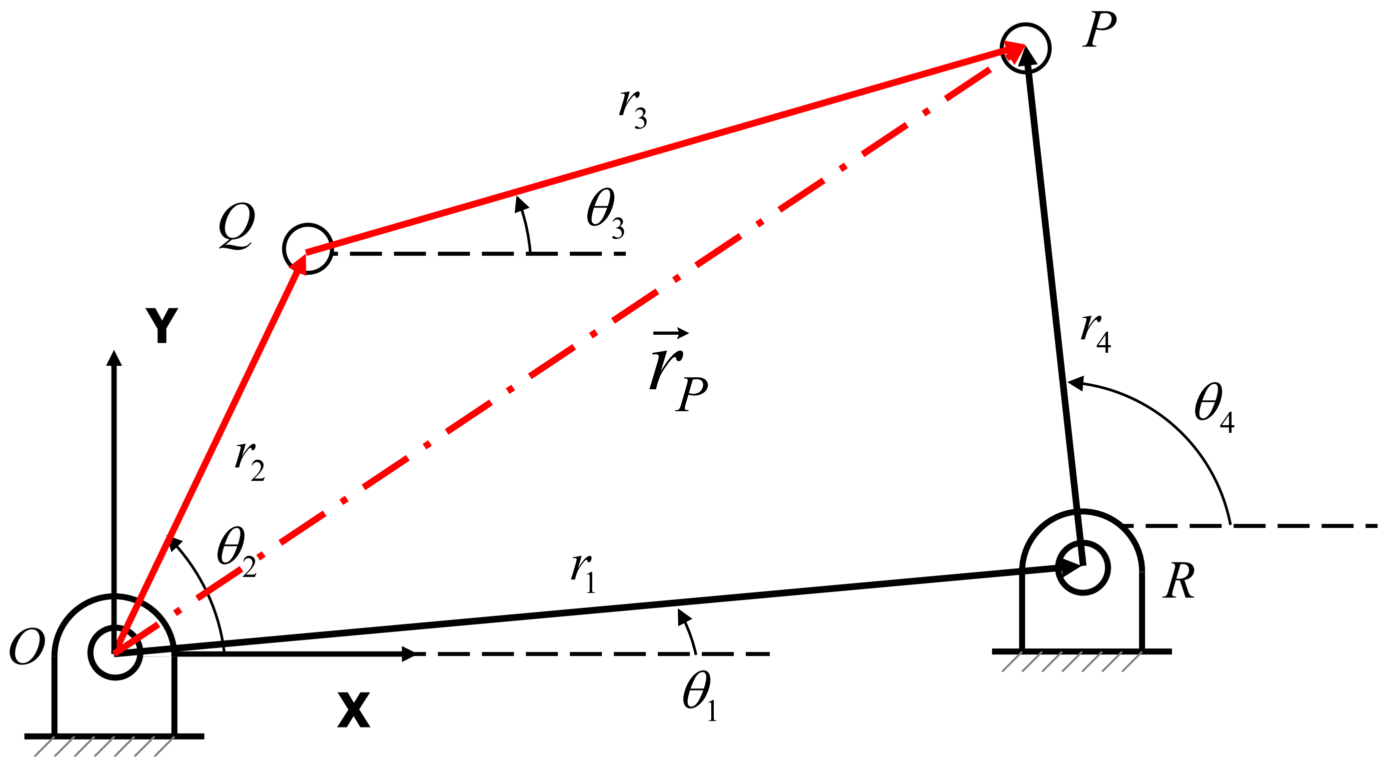
ME 5751: **Kinematics and mechanisms overview**

Lecture 2

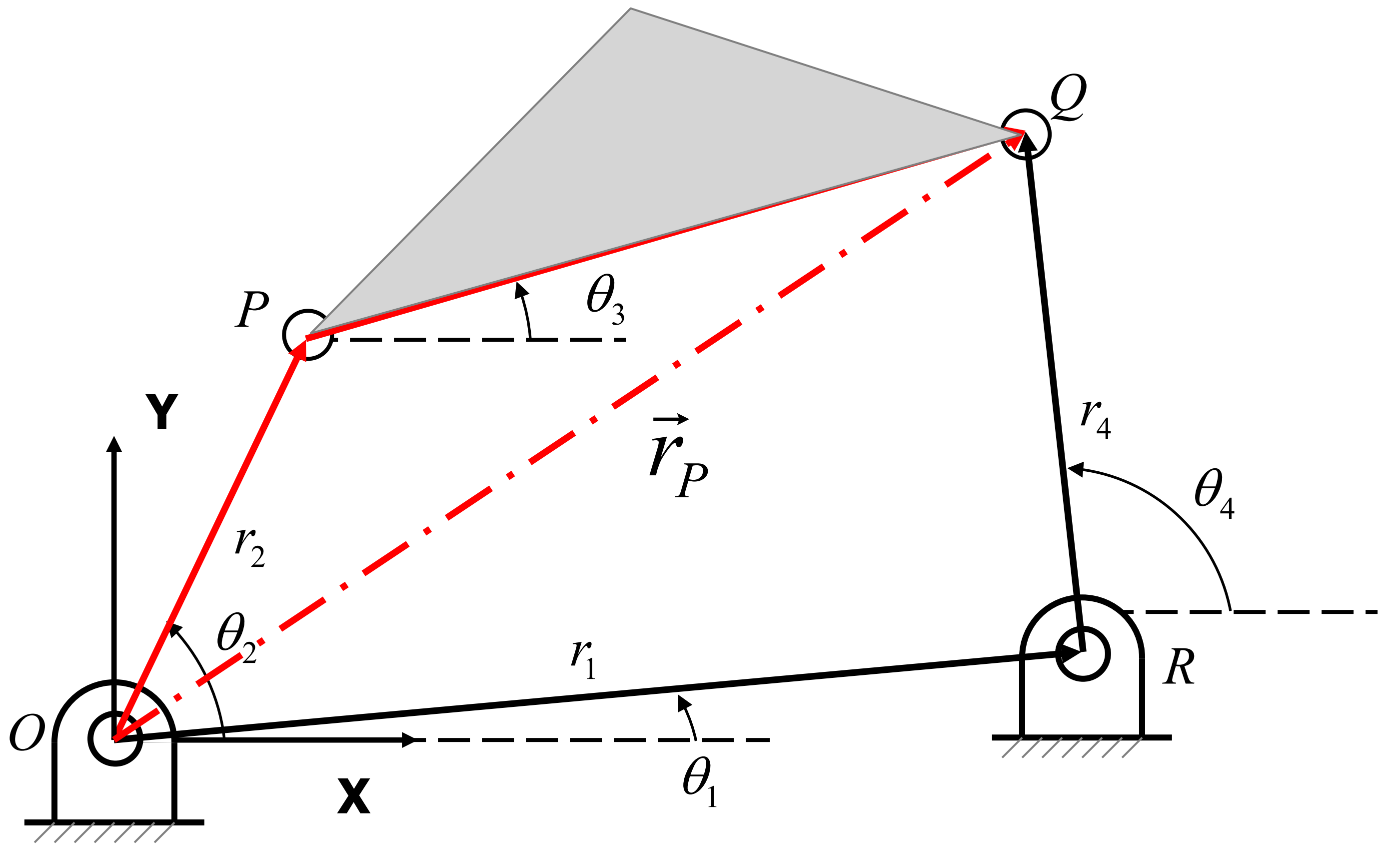
Vector Loop Approach, Positional Analysis of Planar Loop Linkages

Graphical Approach for Linkage Analysis

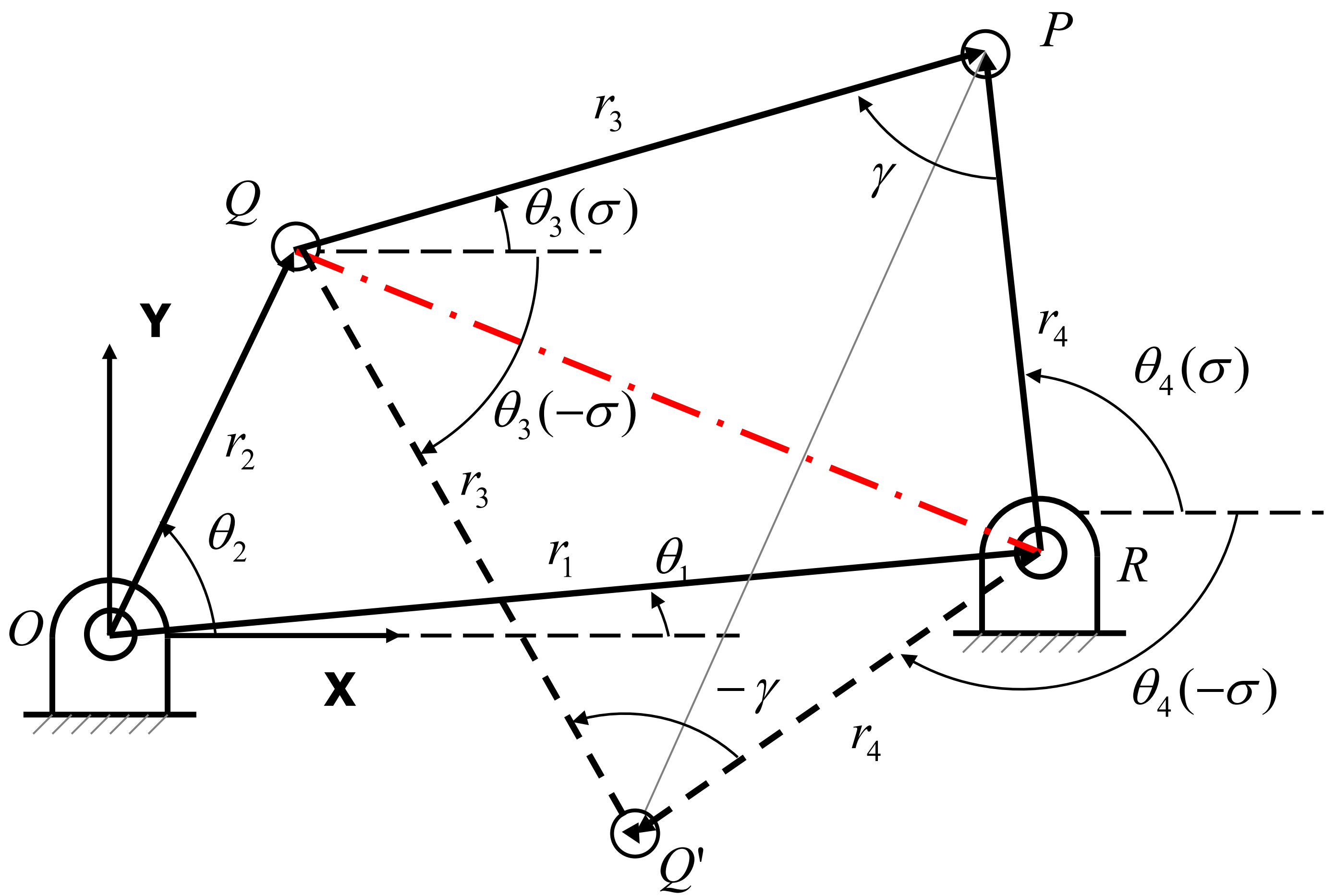
- Draw the linkage to scale (manually or using computer programs, e.g. Solidworks
- Measure the kinematic parameters: angles, distance etc.



Graphical Approach for Linkage Analysis



Graphical Approach for Linkage Analysis



Vector Loop Method for Kinematic analysis of planar linkage

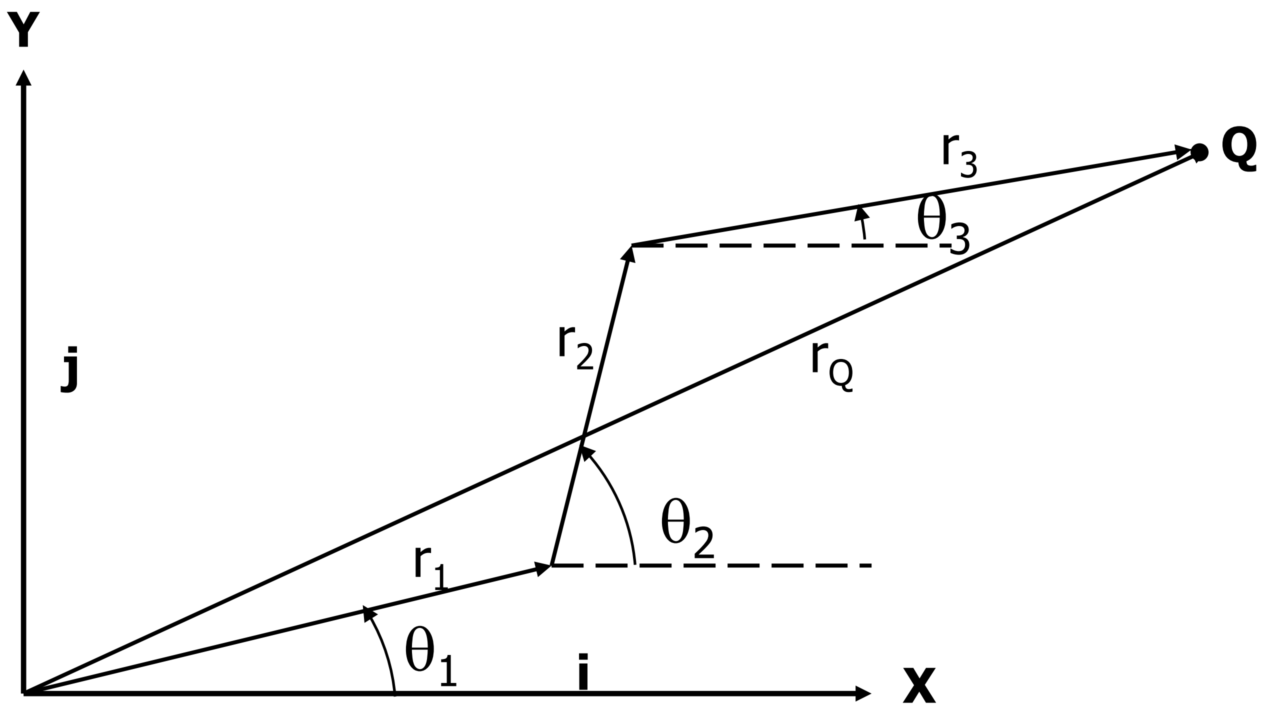
Why Analytical Approach?

- Easy for computer programming
  - Motion control: robotics, machine control
  - Design optimization
- Differentiate these expressions to determine the velocity and acceleration of the mechanism
- How: Uses geometric constraints introduced by mechanism closure
  - Vector loop equations
  - i.e. there are 2 different but equivalent paths connecting points on the same vector loop

Representing Position with Vectors

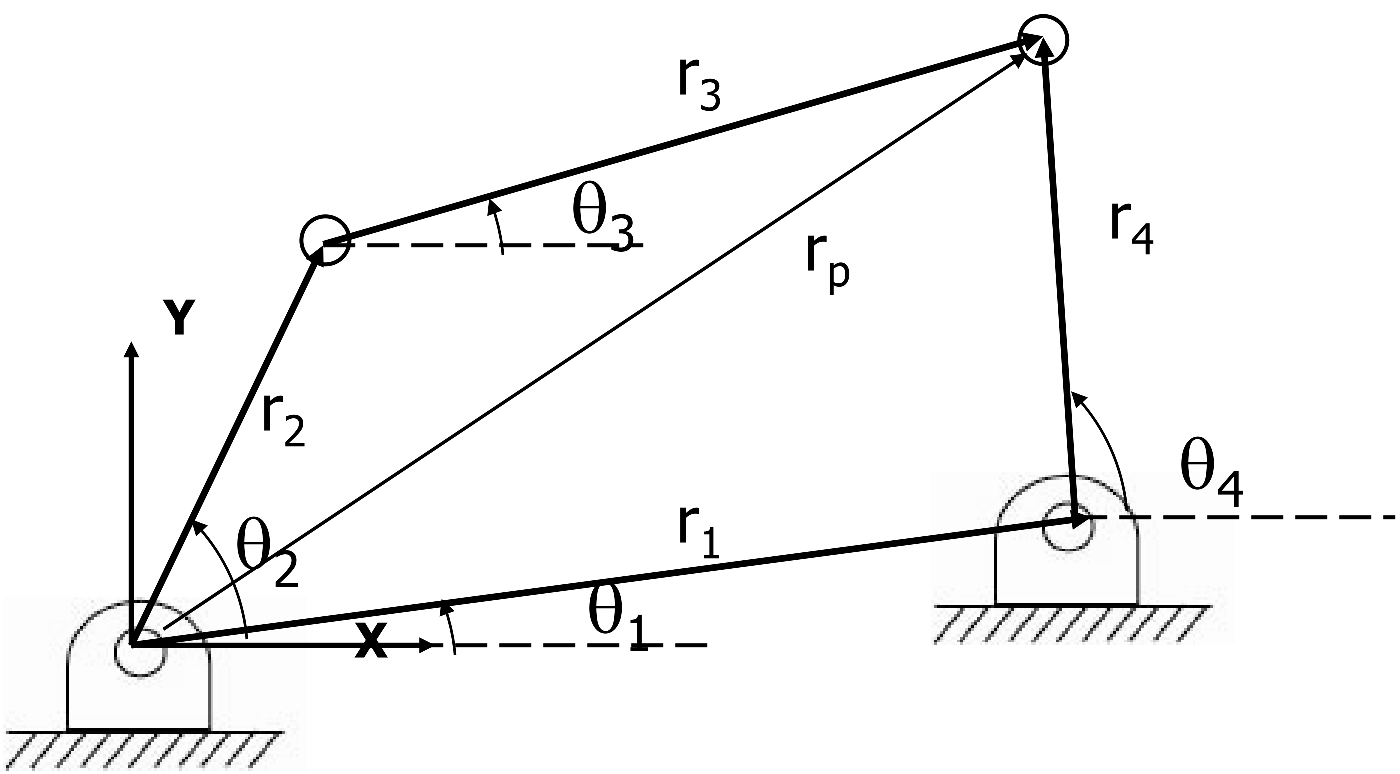
- Vectors have magnitude and direction
- Link lengths r_k, and angles θ_k

$$\vec{r}_Q = \vec{r}_1 + \vec{r}_2 + \vec{r}_3$$



The 4-Bar Linkage: Vector Loop

$$\vec{r}_p = \vec{r}_2 + \vec{r}_3 = \vec{r}_1 + \vec{r}_4 \longrightarrow \text{x and y components}$$



Closure Equations

- Position

$$\hat{i}: r_2 \cos\theta_2 + r_3 \cos\theta_3 = r_1 \cos\theta_1 + r_4 \cos\theta_4$$

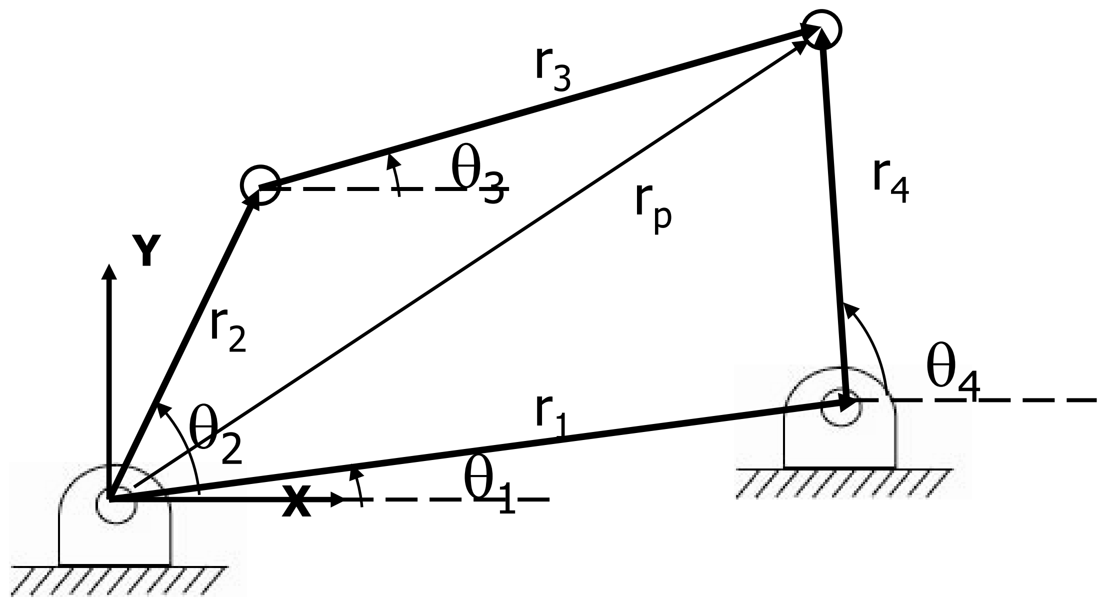
$$\hat{j}: r_2 \sin\theta_2 + r_3 \sin\theta_3 = r_1 \sin\theta_1 + r_4 \sin\theta_4$$

- Identify constants
  - r_1, r_2, r_3, r_4, θ_1
- Identify the driver and what is given
- Differentiate to get velocity and acceleration

Methods for Solving the Vector Loop Equations

$$\vec{r}_p = \vec{r}_2 + \vec{r}_3 = \vec{r}_1 + \vec{r}_4 \longrightarrow \begin{array}{l} \hat{i} : r_2 \cos\theta_2 + r_3 \cos\theta_3 = r_1 \cos\theta_1 + r_4 \cos\theta_4 \\ \hat{j} : r_2 \sin\theta_2 + r_3 \sin\theta_3 = r_1 \sin\theta_1 + r_4 \sin\theta_4 \end{array}$$

- Two equations two unknowns.
- Method 1: Analytical. Eliminate one variable and solve the other
- Method 2: Numerical. Newton's iteration method.
  - Matlab function: fsolve(f, x0).
  - Need an initial guess x0 to the solution



Method 1: Elimination Method

- θ_2 is known, θ_1 is constant, so eliminate θ_3 to solve for θ_4

- Rearrange to isolate θ_3

$$r_3 \cos\theta_3 = r_1 \cos\theta_1 + r_4 \cos\theta_4 - r_2 \cos\theta_2$$

$$r_3 \sin\theta_3 = r_1 \sin\theta_1 + r_4 \sin\theta_4 - r_2 \sin\theta_2$$

- Square and add

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

Want to Simplify This Expression

$$A\cos\theta_4 + B\sin\theta_4 + C = 0$$

where

$$A = 2r_1r_4\cos\theta_1 - 2r_2r_4\cos\theta_2$$

$$B = 2r_1r_4\sin\theta_1 - 2r_2r_4\sin\theta_2$$

$$C = r_1^2 + r_2^2 + r_4^2 - r_3^2 - 2r_1r_2\left(\cos\theta_1\cos\theta_2 + \sin\theta_1\sin\theta_2\right)$$

There are two solutions this equation

$$\theta_{\mathbf{4}}^{\pm} = 2 * \operatorname{atan}\left(t^{\pm}\right) = 2 * \operatorname{atan}\left(\frac{-B \pm \sqrt{A^2 + B^2 - C^2}}{C - A}\right)$$

See the solution process in the next slide

Math Review: Solve Trigonometric Equations

- To solve the following for angle θ_1 in terms of coefficients A, B and C

$$A cos\theta + B sin\theta + C = 0$$

- Let $\mathrm{t} = \tan\left(\frac{\theta}{2}\right)$, we have $\sin\theta = \frac{2t}{1+t^2}, \cos\theta = \frac{1-t^2}{1+t^2}$

- The original equation can be converted the following quadratic equation

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

- This equation has up to two real roots shown below

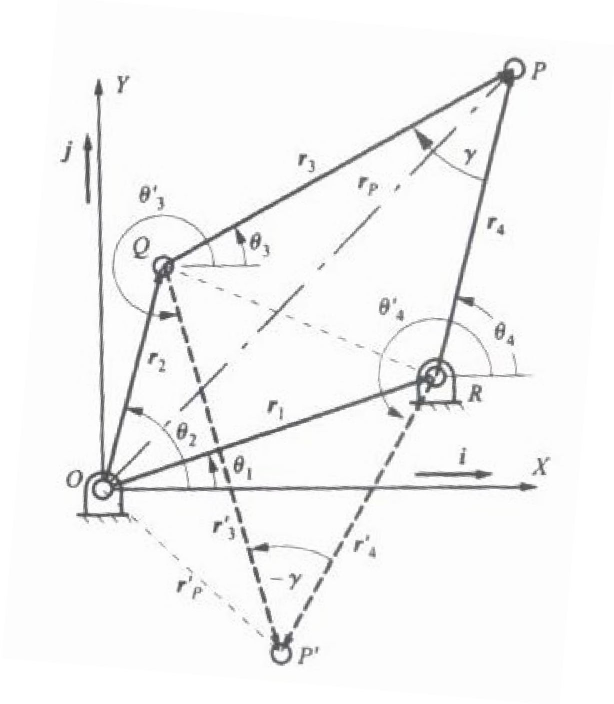
$$t^{\pm} = \frac{-B \pm \sqrt{A^2 + B^2 - C^2}}{C - A}$$

- For each root of t, we can obtain a solution of θ_2 as

$$\theta^{\pm} = 2 * \mathrm{atan}(t^{\pm}) = 2 * \mathrm{atan}\left(\frac{-B \pm \sqrt{A^2 + B^2 - C^2}}{C - A}\right)$$

2 Important Notes about That Angle

- The solutions correspond to 2 assembly modes associated with the + and –



- If the solution of t is complex
  - Mechanism can’t be assembled in the position specified- transition between assembly modes

Solution of the Coupler Angle θ_3

- Sub the solution of θ_4 into these equations

$$r_3 \cos\theta_3 = r_1 \cos\theta_1 + r_4 \cos\theta_4 - r_2 \cos\theta_2$$

$$r_3 \sin\theta_3 = r_1 \sin\theta_1 + r_4 \sin\theta_4 - r_2 \sin\theta_2$$

- With MATLAB

$$\theta_3 = \boldsymbol{atan}2(\boldsymbol{y}, \boldsymbol{x})$$
$$= \boldsymbol{atan}2(r_1 \sin\theta_1 + r_4 \sin\theta_4 - r_2 \sin\theta_2\,, r_1 \cos\theta_1$$
$$+ r_4 \cos\theta_4 - r_2 \cos\theta_2)$$

- Without MATLAB (use a calculator)

$$\theta'_3 = \boldsymbol{atan}\left[\frac{\boldsymbol{y}}{\boldsymbol{x}}\right] = \boldsymbol{atan}\left[\frac{r_1 \sin\theta_1 + r_4 \sin\theta_4 - r_2 \sin\theta_2}{r_1 \cos\theta_1 + r_4 \cos\theta_4 - r_2 \cos\theta_2}\right]$$

if $x>0$ $\theta_3 = \theta'_3$

if $x<0$ $\theta_3 = \theta'_3 + \pi$

When The Coupler Is The Driving Link

- θ_3 is known, θ_1 is constant, so eliminate θ_2 to solve for θ_4

- Rearrange to isolate θ_2

$$r_2 \cos\theta_2 = r_1 \cos\theta_1 + r_4 \cos\theta_4 - r_3 \cos\theta_3$$
$$r_2 \sin\theta_2 = r_1 \sin\theta_1 + r_4 \sin\theta_4 - r_3 \sin\theta_3$$

- Look similar to anything from before?

- Table 7.1 of your text
  - Link 2 is the driver: M=2, J=3
  - Link 3 is the driver: M=3, J=2

Table 7.1 Summary of 4-Bar

- Table 7.1 Summarize position equations for 4-bar linkages. The position of the driver link θ_M are given.

  - Use M=2, J=3 if link 2 is the input/driver link
  - Use M=3, J=2 if link 3 is the input/driver link

Position

$$A = 2r_1 r_4 \cos\theta_1 - 2r_M r_4 \cos\theta_M$$
$$B = 2r_1 r_4 \sin\theta_1 - 2r_M r_4 \sin\theta_M$$
$$C = r_1^2 + r_M^2 + r_4^2 - r_J^2 - 2r_1 r_M \cos(\theta_M - \theta_1)$$

$$\theta_4 = 2\tan^{-1}\left[\frac{-B + \sigma\sqrt{B^2 - C^2 + A^2}}{C - A}\right]; \sigma = \pm 1 \qquad -\pi < \theta_4 < \pi$$

$$\theta_J = \tan^{-1}\left[\frac{r_1 \sin\theta_1 + r_4 \sin\theta_4 - r_M \sin\theta_M}{r_1 \cos\theta_1 + r_4 \cos\theta_4 - r_M \cos\theta_M}\right]$$

Use the sign of $\sin\theta_3, \cos\theta_3$ to determine the quadrant in which the angle θ_3 lies

$$\vec{r}_Q = \vec{r}_2 = r_2\left(\cos\theta_2 \hat{i} + \sin\theta_2 \hat{j}\right)$$
$$\vec{r}_P = \vec{r}_2 + \vec{r}_3 = r_2\left(\cos\theta_2 \hat{i} + \sin\theta_2 \hat{j}\right) + r_3\left(\cos\theta_3 \hat{i} + \sin\theta_3 \hat{j}\right)$$
$$= \vec{r}_1 + \vec{r}_4 = r_1\left(\cos\theta_1 \hat{i} + \sin\theta_1 \hat{j}\right) + r_4\left(\cos\theta_4 \hat{i} + \sin\theta_4 \hat{j}\right)$$

Method 2: Numerical Method for Solving Equations

- Solve systems of algebraic equations

$$f(x) = \begin{cases} f_1(x_1, x_2, \cdots x_n) = 0 \\ f_2(x_1, x_2, \cdots x_n) = 0 \\ \vdots \\ f_n(x_1, x_2, \cdots x_n) = 0 \end{cases}$$

- Kinematic analysis is about solving systems of algebraic equations
  - Numerical solutions:
    - need the initial guess to the solution, local solver
    - Matlab function: fsolve()
    - Mathematica function: FindRoot[]
  - Symbolic solver: global solver, find all solut[ions without initial] guess
    - Matlab function: solve(eqn, vars)
    - Mathematica function: Solve[]

```
syms a b c x
eqn = a*x^2 + b*x + c == 0
```

eqn = $a\,x^2 + b\,x + c = 0$

```
S = solve(eqn)
```

S =

$$\begin{pmatrix} -\frac{b+\sqrt{b^2-4\,a\,c}}{2\,a} \\ -\frac{b-\sqrt{b^2-4\,a\,c}}{2\,a} \end{pmatrix}$$

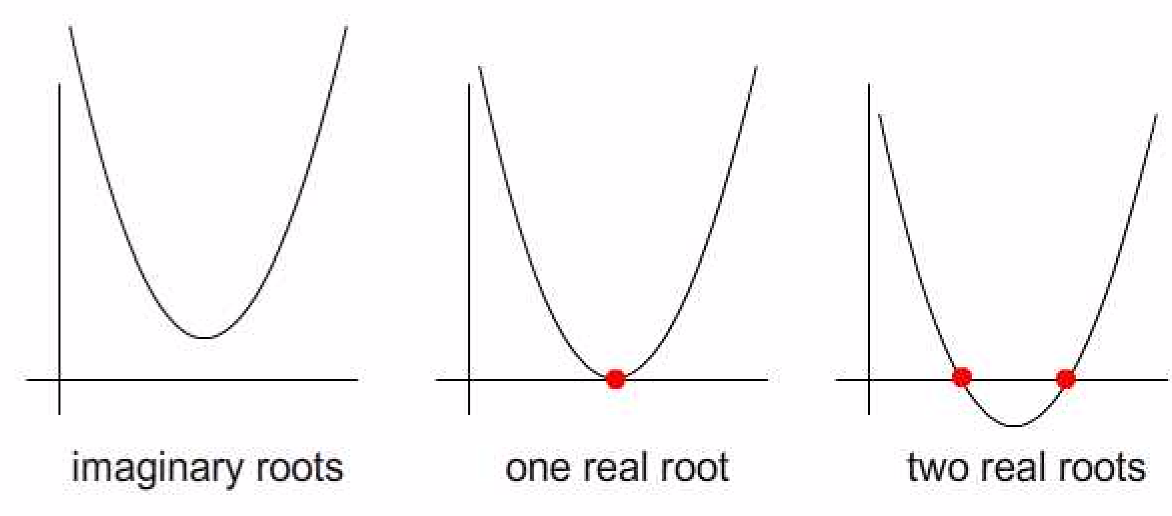
Roots of Quadratic Equations

A quadratic equation

$$ax^2 + bx + c = 0.$$

May have 0 (no), 1 (double) or 2 (distinct) real roots

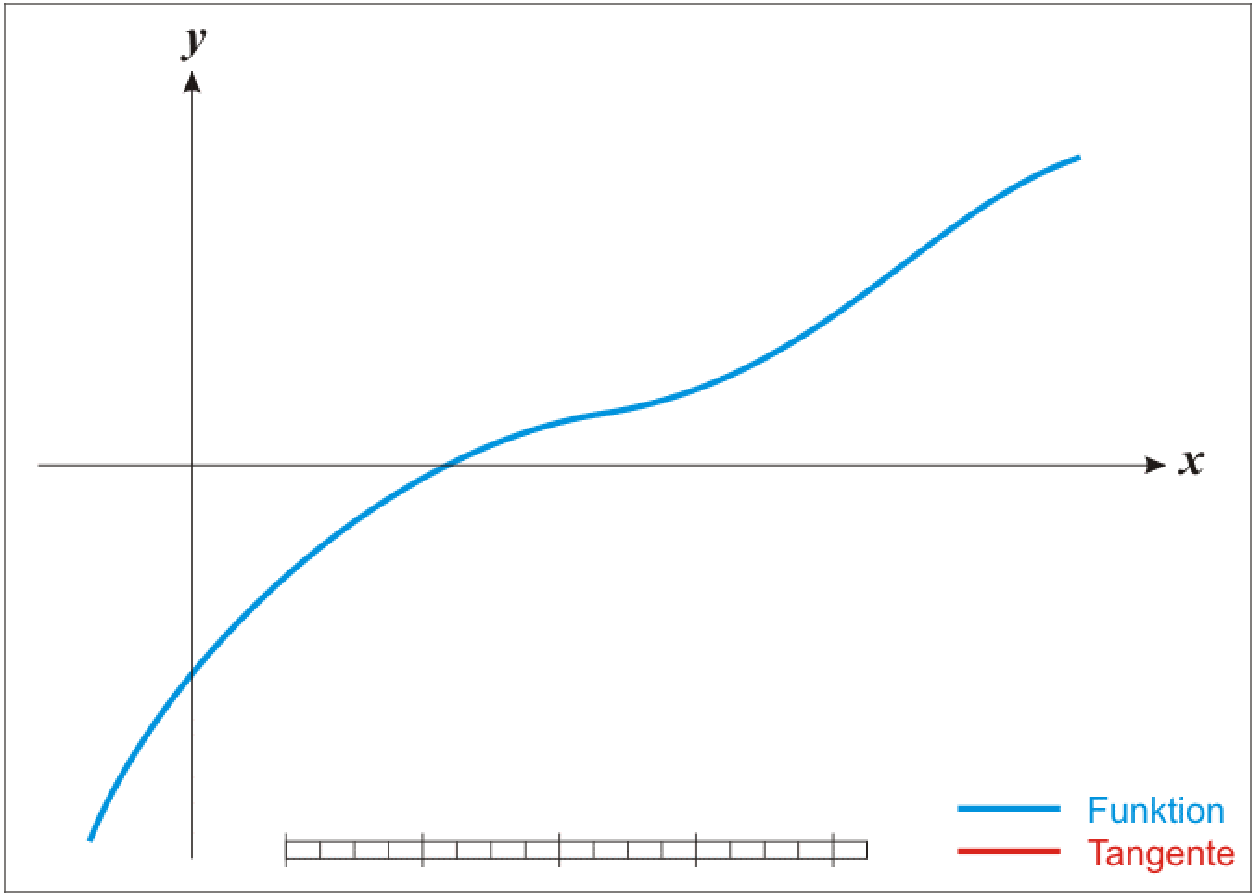
$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$



Newton’s Method for Solving Nonlinear Equations

- Newton’s method: need a good initial guess

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

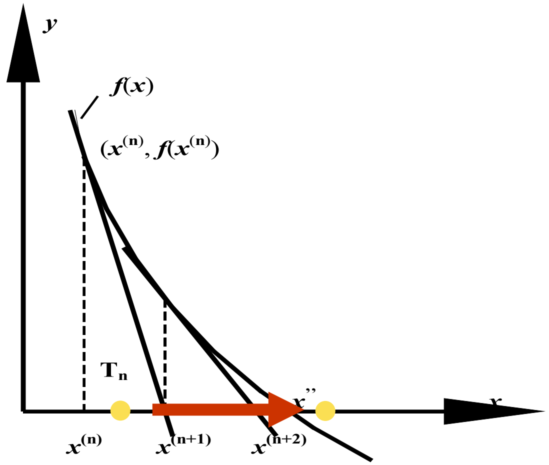


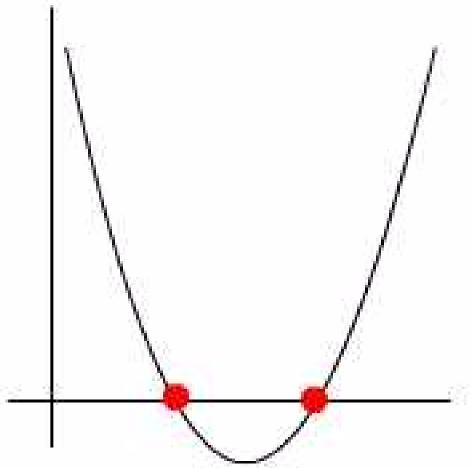
- Matlab function: fsolve(f, x0)
  - f: equation to be solved
  - x0 is the initial guess to the solution

Newton's Method for Solving 2 Equations

- One variable case

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$





Which root it converges to depends on choice of the initial guess

- Two variable case

$$\begin{cases} f_1(x,y) = 0 \\ f_2(x,y) = 0 \end{cases}$$

(1): Start with (x_0, y_0), compute the Jacobian

$$J = \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} \end{bmatrix}$$

(2) Solve the linea equation for $(\Delta x, \Delta y)$:

$$\begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} \end{bmatrix} \begin{Bmatrix} \Delta x \\ \Delta y \end{Bmatrix} = \begin{Bmatrix} \Delta f_1 \\ \Delta f_2 \end{Bmatrix}$$

(3): Update (x, y) as follows

$$(x_1, y_1) = (x_0, y_0) + (\Delta x, \Delta y)$$

(4): Continue to step (1) untill $(\Delta f_1, \Delta f_2)$ is small enough

Matlab Example

- Solve for two nonlinear equations
  - 2 unknowns x_1, x_2

$$e^{-e^{-(x_1+x_2)}} = x_2\left(1+x_1^2\right)$$
$$x_1\cos\left(x_2\right)+x_2\sin\left(x_1\right)=\tfrac{1}{2}.$$

- Convert to f(x)=0

$$e^{-e^{-(x_1+x_2)}} - x_2\left(1+x_1^2\right) = 0$$
$$x_1\cos\left(x_2\right)+x_2\sin\left(x_1\right)-\tfrac{1}{2}=0.$$

- Write a function that computes the left-hand side of these two equations.

```
function F = root2d(x)

F(1) = exp(-exp(-(x(1)+x(2)))) - x(2)*(1+x(1)^2);
F(2) = x(1)*cos(x(2)) + x(2)*sin(x(1)) - 0.5;
```

- Save this code as a file named root2d.m on your MATLAB path.
- Solve the system of equations starting at the point [0,0].

```
fun = @root2d;
x0 = [0,0];
x = fsolve(fun,x0)
```

Matlab output

```
x =

    0.3532    0.6061
```

Numerical Method for Solving Kinematic Analysis

- 2 Equations with 2 unknowns: θ_3 and θ_4

$$\hat{i} : r_2 \cos\theta_2 + r_3 \cos\theta_3 = r_1 \cos\theta_1 + r_4 \cos\theta_4$$

$$\hat{j} : r_2 \sin\theta_2 + r_3 \sin\theta_3 = r_1 \sin\theta_1 + r_4 \sin\theta_4$$

- Solve with Matlab

Main file

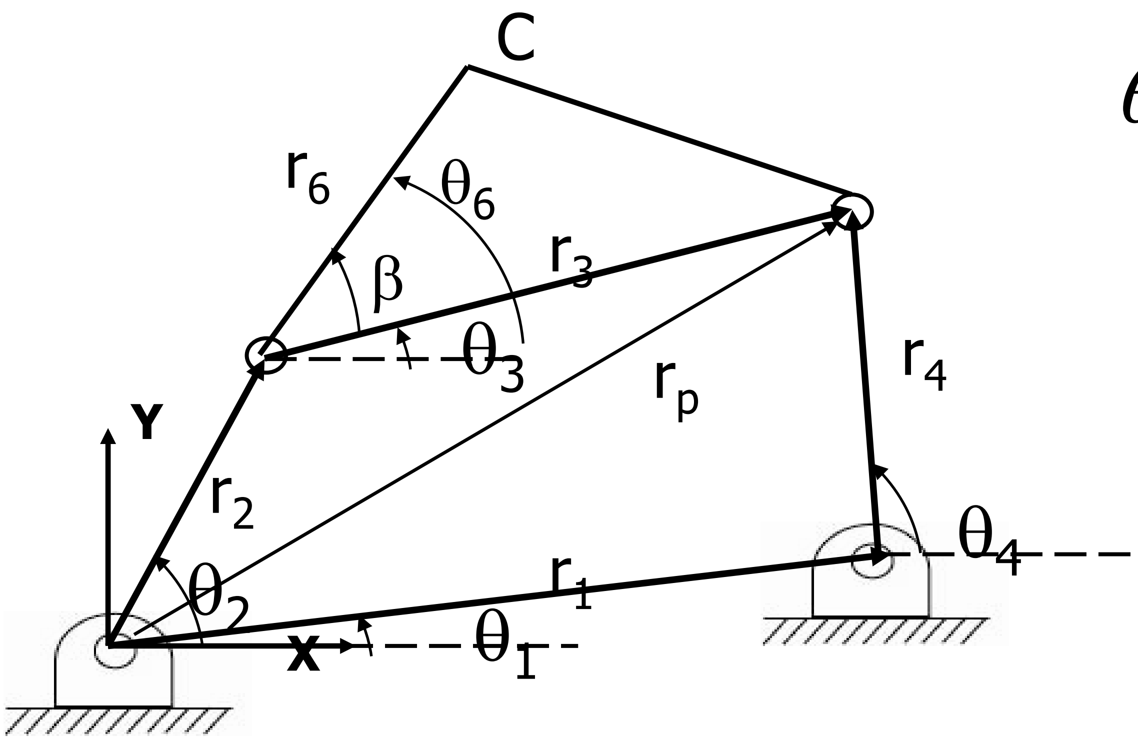
```
r1=1;
r2=2;
r3=3.5;
r4=4;
th1=0;
beta=atan(1/2); %coupler angle
r6=2/cos(beta); %length AE

th2=0;
x0=[-pi/3 -pi/3];

%th2=2*pi/3
%x0=[pi/6 pi/3]

f = @(x) vecloopeq(x,th1,th2,r1,r2,r3,r4);
solth3th4=fsolve(f,x0);
th3=solth3th4(1);
th4=solth3th4(2);
pA=r2*[cos(th2) sin(th2)]
pC=pA+2*[cos(th3) sin(th3)]+1*[cos(th3+pi/2) sin(th3+pi/2)]
```

vecloopeq.m file

```
function y = vecloopeq(x,th1,th2,r1,r2,r3,r4)
%x=[th3 th4]
f1=r2*cos(th2)+r3*cos(x(1))-r4*cos(x(2))-r1*cos(th1);
f2=r2*sin(th2)+r3*sin(x(1))-r4*sin(x(2))-r1*sin(th1)
y=[f1 f2];
end
```

Approach for Rigid Bodies

- Sometimes we are interested in the position, velocity, acceleration of a point in a rigid body that is not a part of the vector loop equations



$$\theta_6 = \theta_3 + \beta$$

Analysis of the Rigid Body

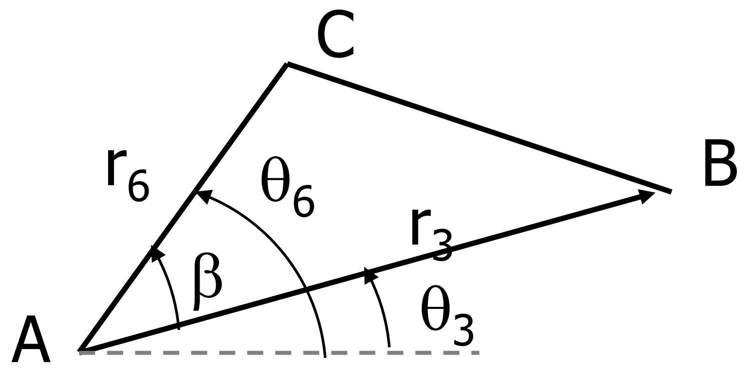
- Position

$$\vec{r}_c = \vec{r}_2 + r_6\left(\cos\theta_6\hat{i} + \sin\theta_6\hat{j}\right)$$

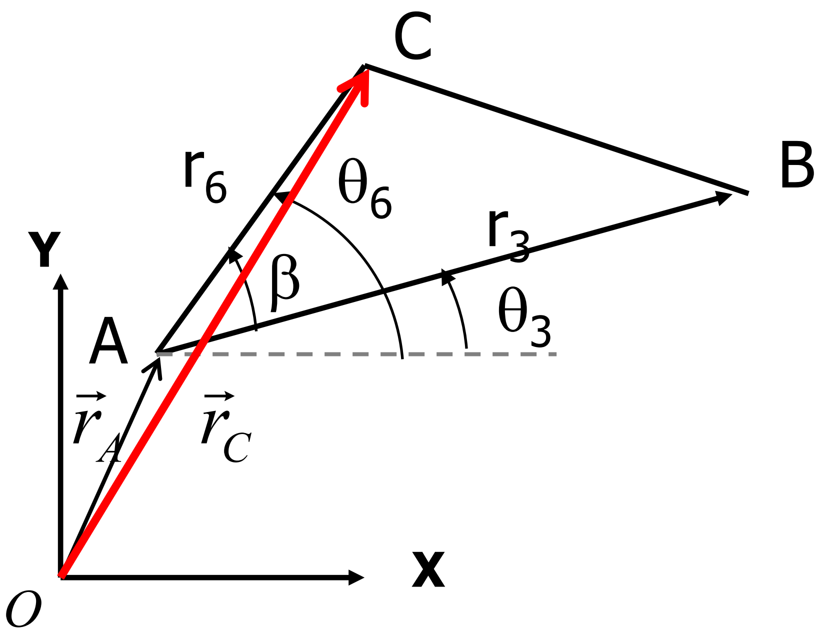
Finding θ_3 Directly

- If we know the position of 2 points on the rigid body (let's call them A and B)

$$\theta_3 = \tan^{-1}\left[\frac{r_{B_y} - r_{A_y}}{r_{B_x} - r_{A_x}}\right]$$



Finding θ_3 Directly



Example 7.1

- Example 7.1: For a given 4-bar linkage, r_1=1", r_2=2", r_3=3.5", r_4=4" and $\theta_1=0°$.
- When the driver angle $\theta_2 = 0, \pi/2, \pi, -\pi/2$, compute the values of $\theta4$, $\theta3$ and coordinates of point A and B.
- Roughly draw these two configurations. (Build a CAD model)
- What is the assembly mode σ gives the solution show in the Figure above?
- Compute coordinates of the coupler point C
- Full positional analysis using Matlab.

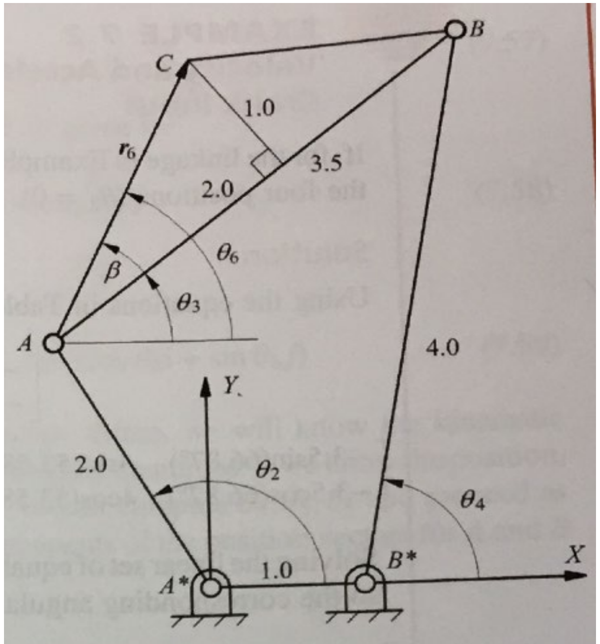


FIGURE 7.9
The linkage for Example 7.1, 7.2, and 7.3.

Example 7.1 Solution (1/2)

Solution

The solution procedure is to use the equations in Table 7.1. First compute A, B, C for each value of θ_2 and then select σ. Next compute θ_4 and then θ_3. The calculations for $\theta_2 = 0$ are as follows

$$A = 2r_1r_4\cos\theta_1 - 2r_2r_4\cos\theta_2 = 2(1)(4) - 2(2)(4) = -8$$

$$B = 2r_1r_4\sin\theta_1 - 2r_2r_4\sin\theta_2 = 0$$

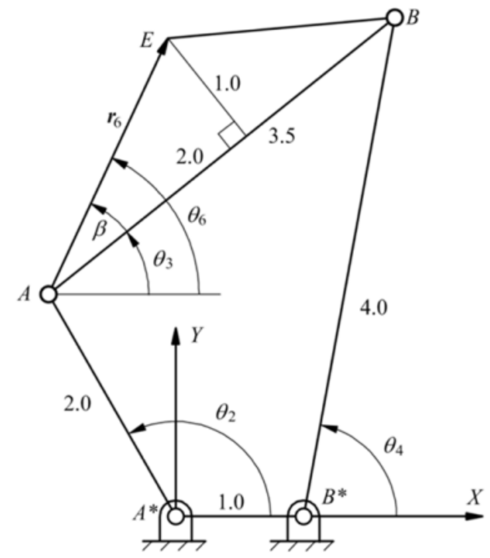
$$C = r_1^2 + r_2^2 + r_4^2 - r_3^2 - 2r_1r_2(\cos\theta_1\cos\theta_2 + \sin\theta_1\sin\theta_2) = 1^2 + 2^2 + 4^2 - 3.5^2 - 2(1)(2) = 4.75$$

$$\theta_4 = 2\tan^{-1}\left[\frac{-B + \sigma\sqrt{B^2 - C^2 + A^2}}{C - A}\right]$$

$$= 2\tan^{-1}\left[\frac{-0 + \sqrt{0^2 - 4.75^2 + (-8)^2}}{4.75 + 8}\right] = 2\tan^{-1}(0.5049) = 53.58^\circ$$

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

$$= \tan^{-1}\left[\frac{4\sin(53.58^\circ)}{1 + 4\cos(53.58^\circ) - 2}\right] = \tan^{-1}\left[\frac{3.2187}{1.3748}\right] = \tan^{-1}(2.3412) = 66.87^\circ$$

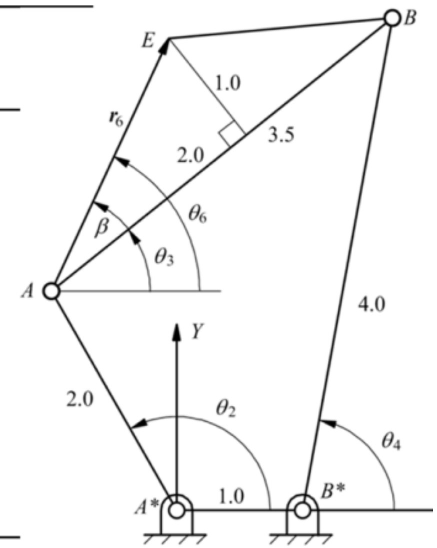


Example 7.1 Solution (2/2)

The remainder of the solution is summarized in Table 7.2.

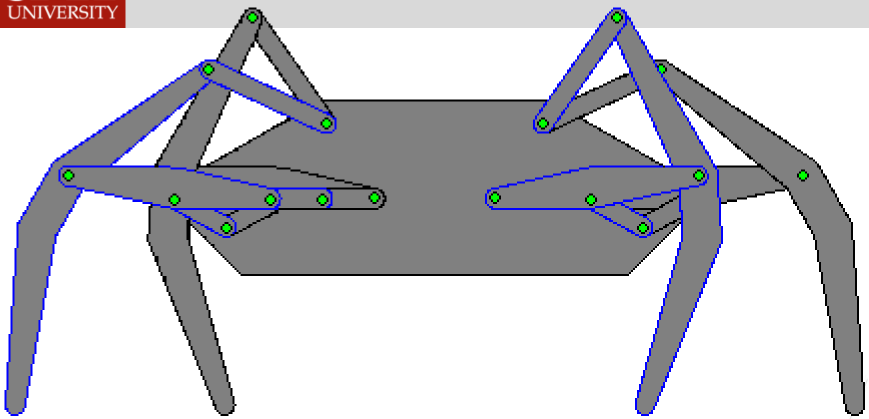
TABLE 7.2 Summary of results for Example 7.1

θ_2	σ	A	B	C	θ_4	θ_3
0	1	-8	0	4.75	53.58°	66.87°
	−1				-53.58°	-66.87°
π/2	1	87	-16	8.75	177.28°	-143.85°
	-1				55.85°	21.98°
π	1	24	0	12.75	-122.09°	-75.52°
	-1				122.09°	75.52°
-π/2	1	8	16	8.75	-55.85°	-21.98°
	-1				-177.28°	148.85°

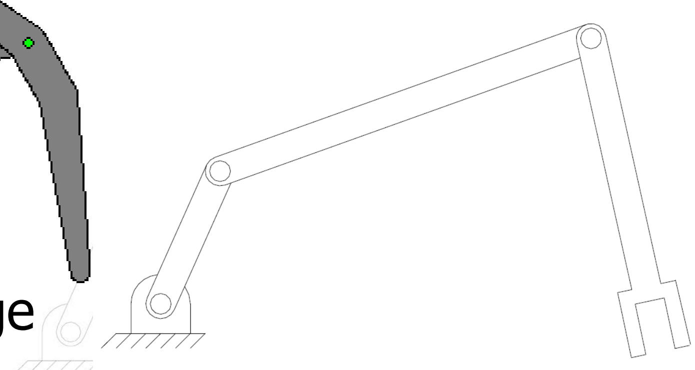


The arithmetic may also be checked by comparing $\gamma = \theta_4 - \theta_3$ for $\sigma = \pm 1$. One value should be minus the other if both values are in the range $-\pi < \gamma \le \pi$. It may be necessary to add or subtract 2π to either value to bring γ into the range $-\pi < \gamma \le \pi$.

More Linkages

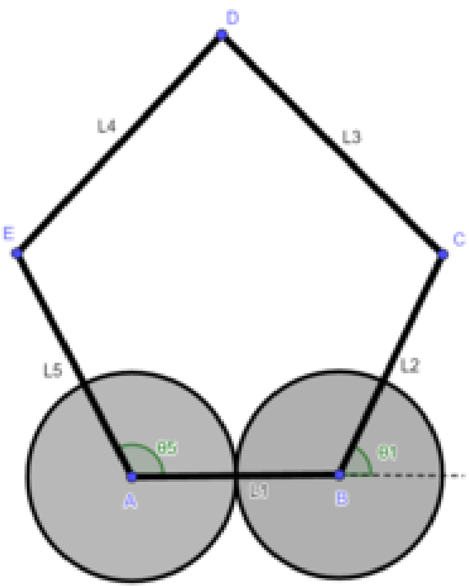


Klann 6-bar straight-line linkage

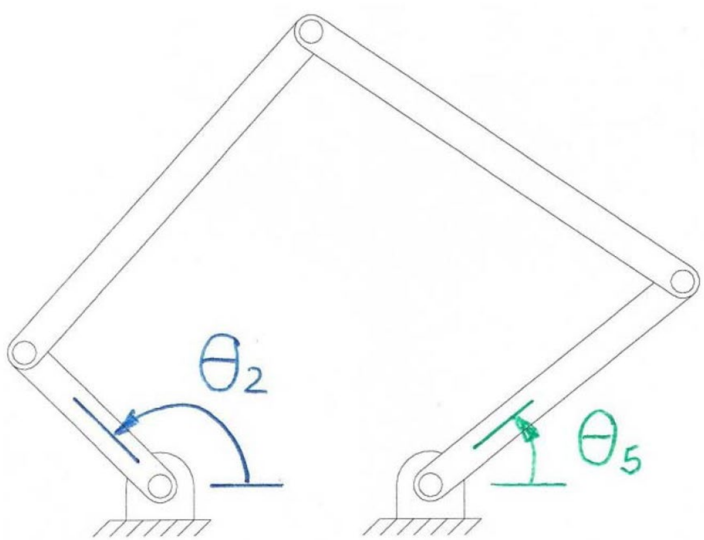


Planar 3-dof Robot

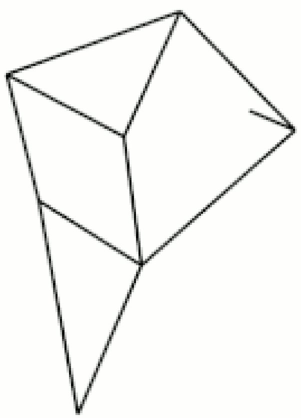
Adept 4-dof SCARA Robot



Geared five-bar linkage



2-dof 5-Bar Parallel Robot



Jensen linkage