Mechanical Linkages

The four-Bar Mechanism

Melody Zhao and Zakariye Muse

Overview

- Introduction to “Watt curves”: “four-bar” linkage
- What are some types of linkages?
- Application: four-bar linkage in daily life
- How to determine the maximal workspace, given the fixed ends of the arms
- What curves could be reached by changes of lengths of segments

A Linkage mechanism

Kinematic chains---rigid bodies(called *links*) connected by *joints*.

Rotary joint: a rotation around a given axis

Prismatic joint: a translation along one given axis

Base link---link fixed to the ground

End-effector---whose position is the output of the mechanism

The number of parameters required to define the position of end-effector is called *degrees of freedom*(DOF)

The planar mechanisms: DOF is at most 3, two translations and one rotation

Figure 1 shows examples of open- and closed-loop planar kinematics chains with only rotary joints.

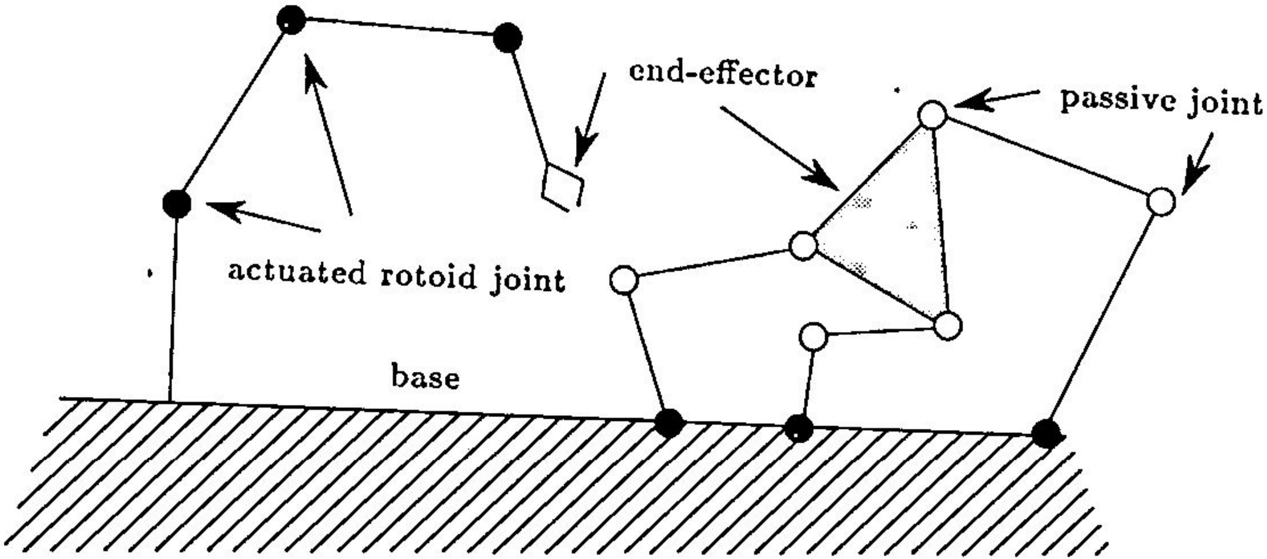


Figure 1: An open-loop planar kinematic chain left, and a closed-loop kinematic chain right

Four-bar mechanism

A closed-loop mechanism: four links with four joints

Cranks: A link that rotates through a full circle relative to another link

Rocker: Any link that does not rotate through a full circle

Workspace: the limited region which can be reached by the end-effector owing to the constraints of the joints

Fixed rod: The rod usually connected to the ground

Coupler rod: The opposite rod on the fixed rod

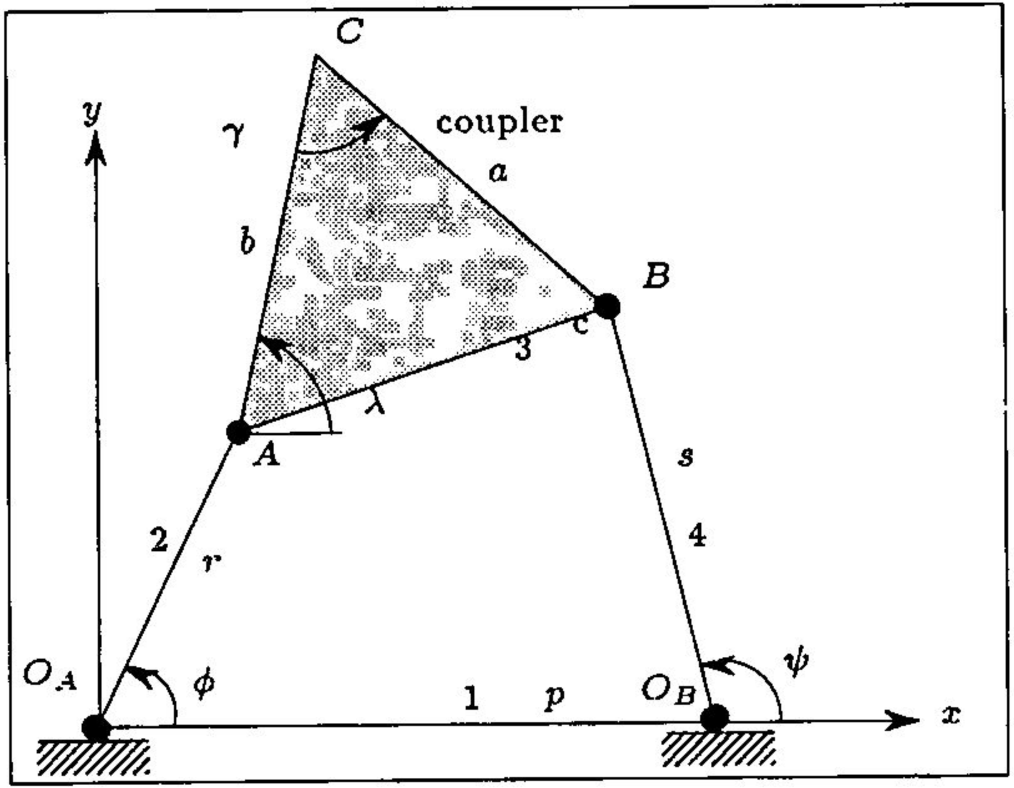


Figure 2: The four-bar mechanism

3-RPR parallel manipulator

A planar robot linkage consisting of three “arms” each fixed at one end with a revolute joint; each containing a prismatic joint, and each attached via a second revolute joint to one common end-effector

Algebraic geometry problems in mechanism theory

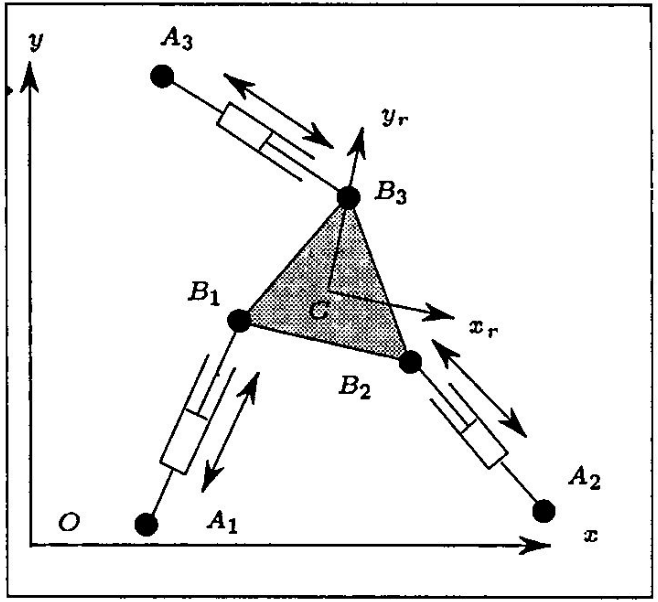
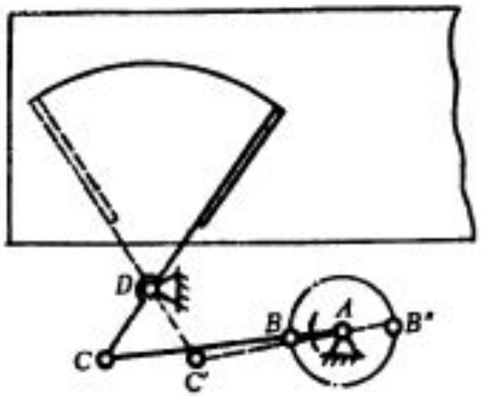


Figure 3: The 3-RPR parallel manipulator:
The end-effector is the gray triangle

Applications in Daily life



Car wiper

Crane

The Big Question

How can we determine the maximal workspace of the robot: that is, set of all points in the coupler curve?

What are we trying to achieve in this project?

- Find an equation for the coupler curve
- Experiment with different lengths of the links and workout their animations to answer the following questions:

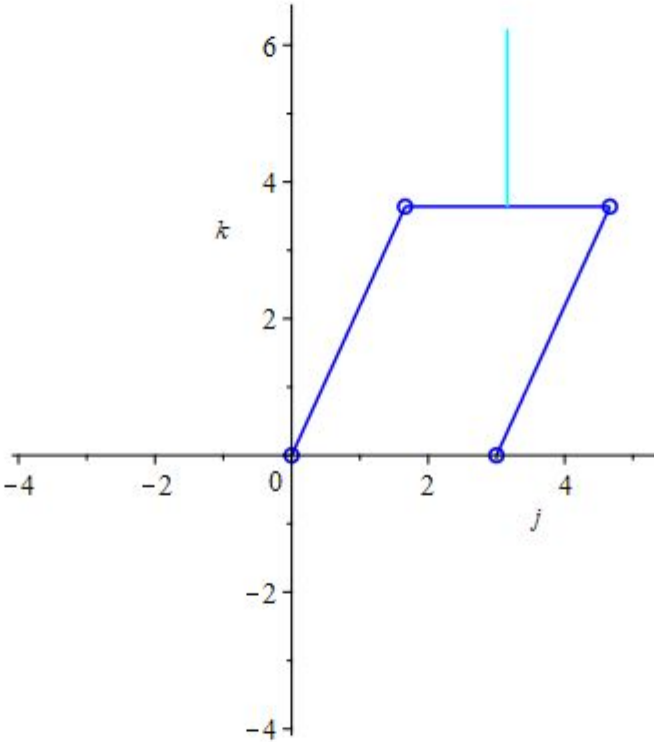
1. Is “hand” reaching all points of the curve?
2. When in terms of the lengths of the links is “hand” hitting the whole curve?

Designing the Mechanism on Maple

Lengths: a, b, c, d

Pts:(0,0)(u,v),(m,n)

(d,0) and (j,k)



> $with(Groebner):$

> $Id := (a,b,c,d) \rightarrow \left[u^2+v^2 - a^2, (m-u)^2+(n-v)^2 - b^2, (m-d)^2+n^2 - c^2, j - \left(\frac{(u+m)}{2} + \frac{s}{2}\cdot(v-n)\right), k - \left(\frac{(v+n)}{2} + \frac{s}{2}\cdot(m-u)\right), s^2-3\right];$

$$Id := (a,b,c,d) \mapsto \left[u^2+v^2-a^2, (m-u)^2+(n-v)^2-b^2, (m-d)^2+n^2-c^2, j - \frac{u}{2} - \frac{m}{2} - \frac{s\,(v-n)}{2}, k - \frac{v}{2} - \frac{n}{2} - \frac{s\,(m-u)}{2}, s^2-3\right] \quad \textbf{(1)}$$

Eliminating Variables and Left Variables

> $leftvars := [j, k, a, b, c, d, s]; elimvars := [m, n, u, v];$

$$leftvars := [j, k, a, b, c, d, s]$$

$$elimvars := [m, n, u, v] \tag{2}$$

Taking The Groebner Basis

$> \quad Ggen := Basis(Id(a, b, c, d), lexdeg(elimvars, leftvars)):$

$> \quad collect(Ggen[2], \{j, k\});$

$$-j^6 + 3\,d\,j^5 + (d\,k\,s + a^2 + b^2 + c^2 - 4\,d^2 - 3\,k^2)\,j^4 + (-2\,d^2\,s\,k - a^2\,d - 2\,b^2\,d - 3\,c^2\,d + 3\,d^3 + 6\,d\,k^2)\,j^3 + (-3\,k^4 + 2\,d\,s\,k^3 + (2\,a^2 + 2\,b^2 + 2\,c^2 - 6\,d^2)\,k^2 + (-a^2\,d\,s - c^2\,d\,s + d^3\,s)\,k - a^4 + a^2\,b^2 + a^2\,c^2 - a^2\,d^2 - b^4 + b^2\,c^2 + 2\,b^2\,d^2 - c^4 + 2\,c^2\,d^2 - d^4)\,j^2 + (3\,d\,k^4 - 2\,d^2\,s\,k^3 + (-a^2\,d - 2\,b^2\,d - 3\,c^2\,d + 3\,d^3)\,k^2 + 2\,a^2\,d^2\,s\,k + 2\,a^4\,d - 3\,a^2\,b^2\,d - a^2\,c^2\,d + a^2\,d^3 + b^4\,d + b^2\,c^2\,d - b^2\,d^3)\,j - k^6 + d\,s\,k^5 + (a^2 + b^2 + c^2 - 2\,d^2)\,k^4 + (-a^2\,d\,s - c^2\,d\,s + d^3\,s)\,k^3 + (-a^4 + a^2\,b^2 + a^2\,c^2 + a^2\,d^2 - b^4 + b^2\,c^2 - 2\,b^2\,d^2 - c^4 + 2\,c^2\,d^2 - d^4)\,k^2 + (-a^2\,b^2\,d\,s + a^2\,c^2\,d\,s - a^2\,d^3\,s + b^4\,d\,s - b^2\,c^2\,d\,s + b^2\,d^3\,s)\,k - a^4\,d^2 + 2\,a^2\,b^2\,d^2 - b^4\,d^2 \tag{3}$$

What Does The Equation Mean?

- A polynomial in j and k with coefficients a,b,c,d and s
- A general equation for the curve of points that will be followed by j, k as the mechanism goes through all of its motion.
- This wasn’t possible a decade ago.
- A coupler curve of degree 6.

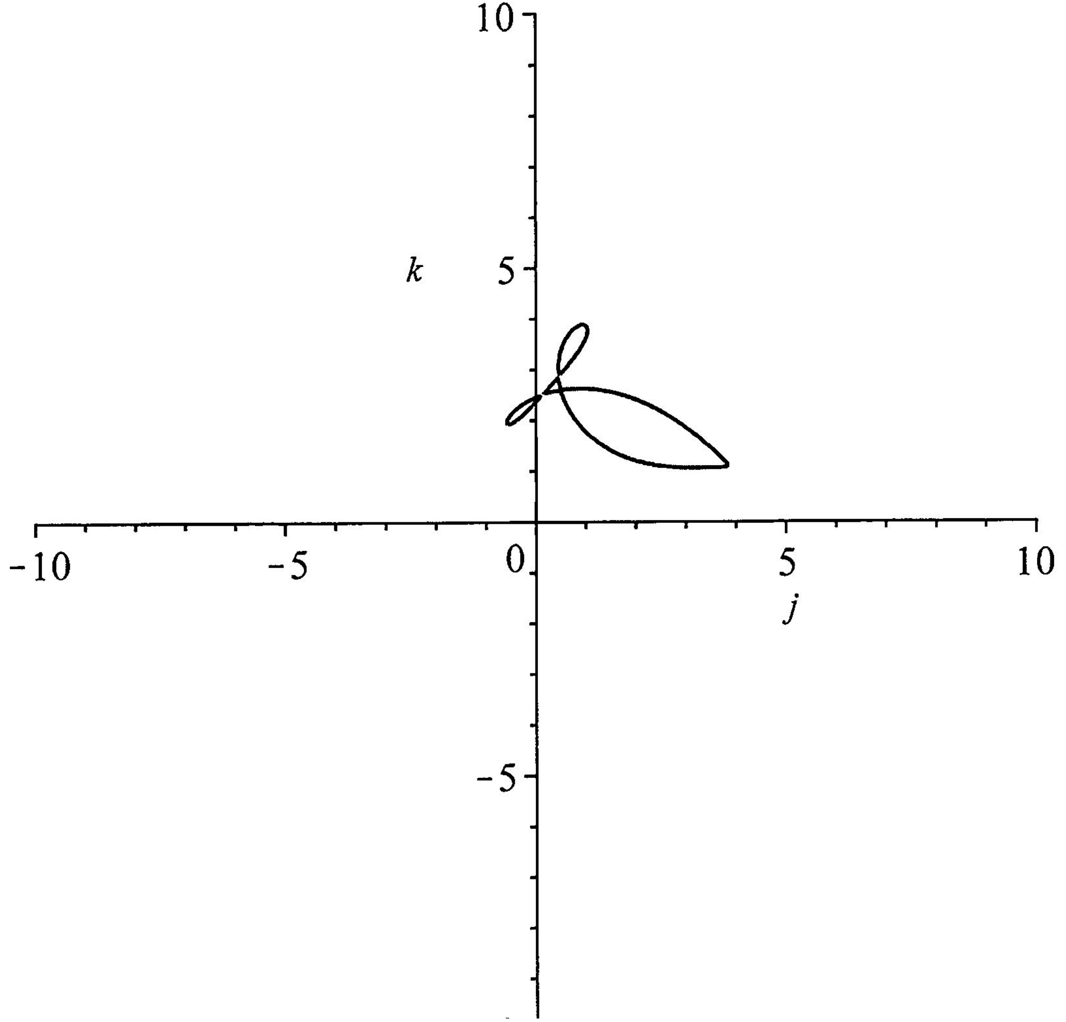
Grashof's Condition

- A test applied when analysing kinematic chains
- For the shortest link to make a full rotation with respect to a neighbouring link, the following condition must hold:

$$S + L \leq P + Q$$

Experimenting with different lengths

```
> with(plots) :
> IP := implicitplot(subs( {a = 1, b = 3, c = 2, d = 4 , s = sqrt(3) }, Ggen[2]), j =- 10 ..10, k =- 10
      ..10, grid = [100, 100], view = [ -10 ..10, -10 ..10]);
```



10
5
k
-5
-10
-5
0
5
10
j

Parameterization

- Find point (u,v) as function of theta
- Find point (m,n) as a function of theta
- Find point (j,k) as the mechanism goes through all of its motion.

```
> uparam := 1·cos(theta) : vparam := 1·sin(theta) :
> CrankI := [u^2 + v^2 - 1, (m - u)^2 + (n - v)^2 - 9, (m - 4)^2 + n^2 - 4];
```

$$CrankI := \left[u^2 + v^2 - 1,\ (m - u)^2 + (n - v)^2 - 9,\ (m - 4)^2 + n^2 - 4\right] \tag{5}$$

```
> CrankB := Basis(CrankI, plex(m, n, u, v));
```

$$CrankB := \left[u^2 + v^2 - 1,\ 64\, n^2 v^2 - 64\, n\, v^3 + 40\, n\, u\, v - 96\, u\, v^2 + 225\, n^2 - 140\, n\, v - 76\, v^2 - 400\, u - 400,\ 8\, n^2 u - 8\, n\, u\, v - 17\, n^2 + 12\, n\, v + 12\, v^2 + 16\, u + 16,\ m\, v^2 - n\, u\, v - 4\, n\, v + 15\, m - 10\, u - 40,\ m\, u + n\, v - 4\, m + 10,\ -8\, n^2 v + 8\, n\, v^2 + 15\, m\, n - 10\, m\, v - 10\, n\, u + 12\, u\, v - 40\, n + 32\, v,\ m^2 + n^2 - 8\, m + 12\right] \tag{6}$$

```
> nops(CrankB);
```

$$7 \tag{7}$$

```
> drivingcrank := animate([t·uparam, t·vparam, t = 0 ..1], theta = 0 ..2·Pi, frames = 100, color
      = blue) :
> followingcrank := animate([(1 − t)·3 + t·mparam, t·nparam, t = 0 ..1], theta = 0 ..2·Pi, frames
      = 100, color = blue) :
> connectingcrank := animate([(1 − t)·uparam + t·mparam, (1 − t)·vparam + t·nparam, t = 0
      ..1], theta = 0 ..2·Pi, frames = 100, color = blue) :
> joints := animate({[0.1·cos(phi), 0.1·sin(phi), phi = 0 ..2·Pi], [0.1·cos(phi) + uparam, 0.1
      ·sin(phi) + vparam, phi = 0 ..2·Pi], [0.1·cos(phi) + mparam, 0.1·sin(phi) + nparam, phi
      = 0 ..2·Pi], [0.1·cos(phi) + 3, 0.1·sin(phi), phi = 0 ..2·Pi]}, theta = 0 ..2·Pi, frames = 100,
      color = blue) :
```

$$> eff := animate\left(\left[\frac{(uparam + mparam)}{2} + t\cdot\frac{\mathrm{sqrt}(3)}{2}\cdot(vparam - nparam),\ \frac{(vparam + nparam)}{2} + \frac{t\cdot\mathrm{sqrt}(3)}{2}\cdot(mparam - uparam),\ t = 0\,..1\right],\ \mathrm{theta} = 0\,..2\cdot\mathrm{Pi},\ frames = 100,\ color = cyan\right) :$$

```
> display(IP, drivingcrank, followingcrank, connectingcrank, joints, eff, scaling = constrained);
```

```
> drivingcrank := animate([t·uparam, t·vparam, t = 0 ..1], theta = 0 ..2·Pi, frames = 100, color
      = blue) :
> followingcrank2 := animate([(1 − t)·3 + t·mparam2, t·nparam2, t = 0 ..1], theta = 0 ..2·Pi,
      frames = 100, color = blue) :
> connectingcrank2 := animate([(1 − t)·uparam + t·mparam2, (1 − t)·vparam + t·nparam2, t
      = 0 ..1], theta = 0 ..2·Pi, frames = 100, color = blue) :
> joints2 := animate({[0.1·cos(phi), 0.1·sin(phi), phi = 0 ..2·Pi], [0.1·cos(phi) + uparam, 0.1
      ·sin(phi) + vparam, phi = 0 ..2·Pi], [0.1·cos(phi) + mparam2, 0.1·sin(phi) + nparam2, phi
      = 0 ..2·Pi], [0.1·cos(phi) + 3, 0.1·sin(phi), phi = 0 ..2·Pi]}, theta = 0 ..2·Pi, frames = 100,
      color = blue) :
```

> $eff2 := animate\left(\left[\frac{(uparam + mparam2)}{2} + t\cdot\frac{\mathrm{sqrt}(3)}{2}\cdot(vparam - nparam2),\ \frac{(vparam + nparam2)}{2} + \frac{t\cdot\mathrm{sqrt}(3)}{2}\cdot(mparam2 - uparam),\ t = 0 ..1\right], \mathrm{theta} = 0 ..2\cdot\mathrm{Pi},\ frames = 100,\ color = cyan\right) :$

```
> display(IP, drivingcrank, followingcrank2, connectingcrank2, joints2, eff2, scaling = constrained,
      view = [-6 ..6, -6 ..6]);
```

Animation

Shown in MAPLE

Another Example with Different Combinations

a= 4

b=3

c=3

d=4

Animation

Shown in MAPLE

Bibliography

Cox, David A., et al. *Ideals, Varieties, and Algorithms An Introduction to Computational Algebraic Geometry and Commutative Algebra* by David A Cox, John Little, Donal O’Shea. Springer International Publishing, 2015.

González-Vega, Laureano., and Tomás. Recio. *Algorithms in Algebraic Geometry and Applications* Edited by Laureano González-Vega, Tomás Recio. Birkhäuser Basel, 1996.