Position Analysis

Nguyen Quy Khoi

HCMUT-OISP - Faculty of Mechanical Engineering

20 February, 2020

Outline

1. Introduction
   - Absolute Cartesian Method
2. Examples
   - Example: Slider-crank Mechanism
   - Example: Four-bar linkage Mechanism
   - Example: Inverted slider-crank Mechanism
   - Example: R-RTR-RTR mechanism

Definition

- Use a fixed reference frame which has x, y, z-axes (as opposed to r, θ, z-axes or ρ, θ, z-axes).
- In a planar kinematic chain, the position of every link is defined by:
  - Joint positions
  - Center of gravity g
  - Angle θ with respect to x-axis

For planar mechanisms, the following relations are considered:

$$(x_A - x_B)^2 + (y_A - y_B)^2 = AB^2 = l_{AB}^2$$

$$k = \tan\theta = \frac{y_B - y_A}{x_B - x_A}$$

$$\Rightarrow y = ax + b$$

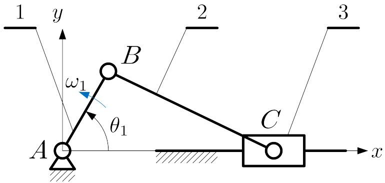
where:

- $A(x_A, y_A)$, $B(x_B, y_B)$ are joint coordinates
- l_{AB} is the length of link AB
- θ is the angle of link AB with respect to x-axis
- a, b are coefficients (a is slope of link AB, b is intercept)

Solving these equations often yields 2 position coordinates. Depending on positions of other links, choose the correct solution.

Example 1: Slider-crank Mechanism

Given	$l_{AB} = l_1 = 0.5m$ $l_{BC} = l_2 = 1m$ $\theta_1 = 60°$
Find	x_C



Solution 1

Position of joint B: $\vec{\boldsymbol{r}}_{\boldsymbol{B}} = x_B\vec{\mathbf{i}} + y_B\vec{\mathbf{j}} = l_1\cos\theta_1\vec{\mathbf{i}} + l_1\sin\theta_1\vec{\mathbf{j}}$
Position of joint C: $\vec{\boldsymbol{r}}_{\boldsymbol{C}} = x_C\vec{\mathbf{i}}$

$$\Rightarrow (x_B - x_C)^2 + y_B^2 = l_1^2$$

Solving the system of equations yields x_{C_1} and x_{C_2}. Notice that in the mechanism, $x_C > x_B$ is the condition to obtain correct solution.

Solution 2

Let $\vec{r}_B = l_1 \cos\theta_1 \vec{\mathbf{i}} + l_1 \sin\theta_1 \vec{\mathbf{j}}$

We can determine position of C by using direct solution:

$$\vec{r}_C = \left(l_1 \cos\theta_1 + l_2 \left(\arcsin \frac{l_1 \sin\theta_1}{l_2} \right) \right) \vec{\mathbf{i}}$$

MATLAB R2019a code

```
function [rB, rC] = pos(phi1)
global AB BC
syms x

rB = AB*[cos(phi1), sin(phi1), 0];
rC = [AB*cos(phi1)+BC*asin(AB*sin(phi1)/BC), 0, 0];
end
```

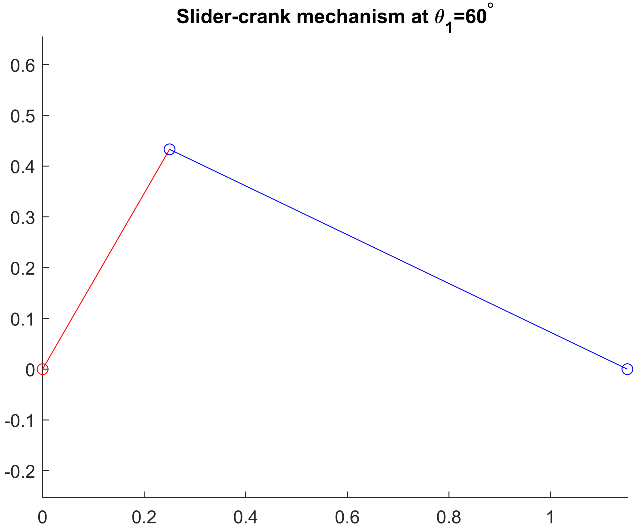
MATLAB R2019a code

```
function [rB, rC] = pos(phi1)
global AB BC
syms x

rB = AB*[cos(phi1), sin(phi1), 0]; rC = [x, 0, 0];

% find rC
eq = (rB(1)-rC(1))^2 + rB(2)^2 == BC^2;
sol = solve(eq, x);
rC1 = subs(rC, sol(1)); rC2 = subs(rC, sol(2));
if rC1(1) > rB(1), rC = double(rC1);
else, rC = double(rC2); end
end
```

Plotting using MATLAB R2019a

```
global AB BC omg1 alp1
syms theta(t)

AB = .5; BC = 1; omg1 = [0,0,1]; alp1 = [0,0,-1];

hold on
[rB, rC] = pos(pi/3);
plot([0, rB(1)], [0,rB(2)], 'r-o');
plot([rB(1), rC(1)], [rB(2), rC(2)], 'b-o');
axis equal
```

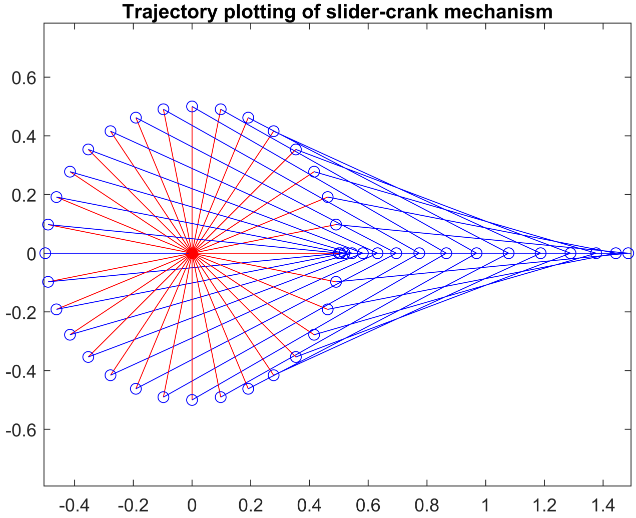
Output figure



Trajectory plotting using MATLAB R2019a

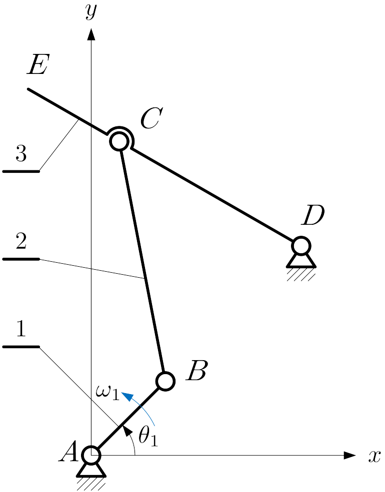
```
global AB BC omg1 alp1
AB = .5; BC = 1; omg1 = [0,0,1]; alp1 = [0,0,-1];

for phi=0:2*pi/32:2*pi
    hold on, axis equal
    [rB, rC] = pos(phi);
    plot([0, rB(1)], [0,rB(2)], 'r-o');
    plot([rB(1), rC(1)], [rB(2), rC(2)], 'b-o');
end
```

Output figure



Example 2: Four-bar linkage Mechanism

Given	$l_{AB} = l_1 = 0.15m$
	$l_{BC} = l_2 = 0.35m$
	$l_{CD} = l_3 = 0.3m$
	$l_{CE} = l_4 = 0.15m$
	$x_D = 0.3m$
	$y_D = 0.3m$
	$\theta_1 = 45°$
Find	$\vec{r}_B$, $\vec{r}_C$, $\vec{r}_E$



Solution

Position of joint B: $\vec{\boldsymbol{r}}_{\boldsymbol{B}} = x_B\vec{\mathbf{i}} + y_B\vec{\mathbf{j}} = l_1\cos\theta_1\vec{\mathbf{i}} + l_1\sin\theta_1\vec{\mathbf{j}}$
Position of joint C: $\vec{\boldsymbol{r}}_{\boldsymbol{C}} = x_C\vec{\mathbf{i}} + y_C\vec{\mathbf{j}} = 0.1\vec{\mathbf{j}}$
Position of joint D: $\vec{\boldsymbol{r}}_{\boldsymbol{D}} = x_D\vec{\mathbf{i}} + y_D\vec{\mathbf{j}}$
Position of joint E: $\vec{\boldsymbol{r}}_{\boldsymbol{E}} = x_E\vec{\mathbf{i}} + y_E\vec{\mathbf{j}}$

$$\Rightarrow \begin{cases} (x_B - x_C)^2 + (y_B - y_C)^2 = l_1^2 \\ (x_D - x_C)^2 + (y_D - y_C)^2 = l_3^2 \end{cases}$$

Solving the system of equations yields x_{C_1} and x_{C_2}. Notice that in the mechanism, $0 > x_C$ is the condition to obtain correct solution.

$$\Rightarrow \begin{cases} (x_C - x_E)^2 + (y_C - y_E)^2 = l_4^2 \\ \dfrac{y_C - y_E}{x_C - x_E} = \dfrac{y_D - y_E}{x_D - x_E} \end{cases}$$

Solving the system of equations yields x_{E_1} and x_{E_2}. Notice that in the mechanism, $x_C > x_E$ is the condition to obtain correct solution.

MATLAB R2019a code

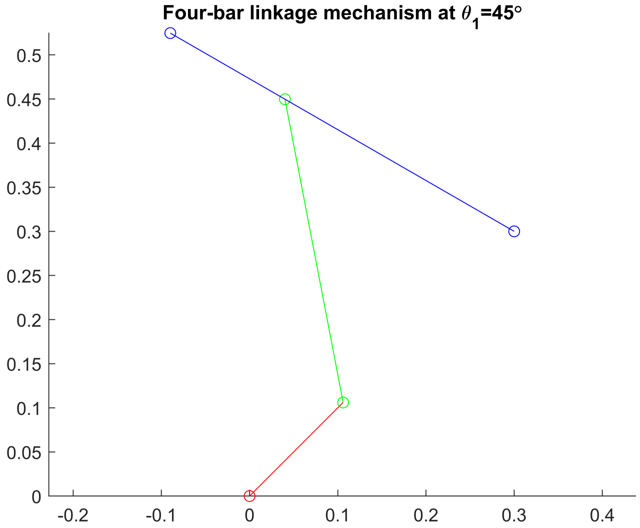
```
function [rB, rC, rE] = pos(phi1)
global AB BC CD CE rD
syms x y

rB = AB*[cos(phi1), sin(phi1), 0];

% find rC
rC = [x, y, 0];
eq1 = (x-rB(1))^2 + (y-rB(2))^2 == BC^2;
eq2 = (x-rD(1))^2 + (y-rD(2))^2 == CD^2;
sol = solve(eq1, eq2, x, y);
if 0>sol.x(1)
    rC = double(subs(rC, [x, y], [sol.x(1), sol.y(1)]));
else
    rC = double(subs(rC, [x, y], [sol.x(2), sol.y(2)])); end
```

```
% find rE
rE = [x, y, 0];
eq1 = (rD(2)-rE(2))/(rD(1)-rE(1)) == (rD(2)-rC(2))/(rD(1)-rC
    (1));
eq2 = (rC(1)-rE(1))^2 + (rC(2)-rE(2))^2 == CE^2;
sol = solve(eq1, eq2, x, y);
if rC(1)>double(sol.x(1))
    rE = double(subs(rE, [x, y], [sol.x(1), sol.y(1)]));
else
    rE = double(subs(rE, [x, y], [sol.x(2), sol.y(2)])); end
end
```

Plotting using MATLAB R2019a

```
global AB BC CD CE rD omg1 alp1
AB = .15; BC = .35; CD = .3; CE = .15;
rD = [.3 .3 0]; omg1 = [0, 0, 2*pi]; alp1 = [0,0,0];

hold on
[rB, rC, rE] = pos(pi/4);
plot([0, rB(1)], [0,rB(2)], 'r-o')
plot([rB(1), rC(1)], [rB(2), rC(2)], 'g-o')
plot([rD(1), rE(1)], [rD(2), rE(2)], 'b-o')
axis equal
```

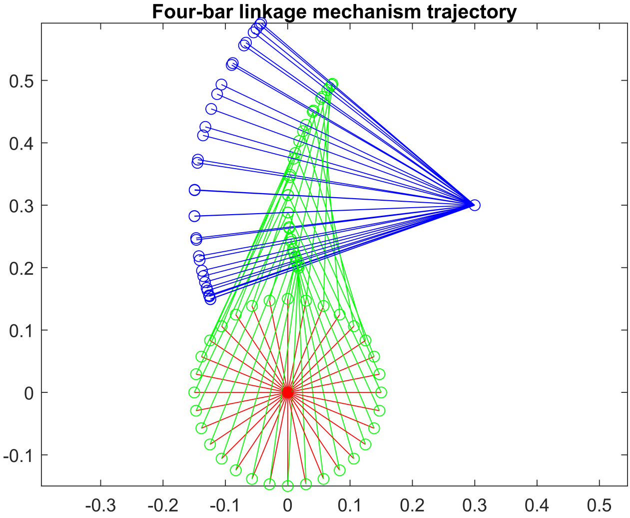
Output figure



Trajectory plotting using MATLAB R2019a

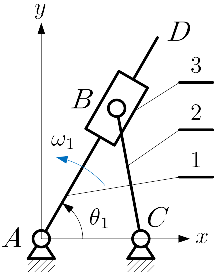
```
global AB BC CD CE rD omg1 alp1
AB = .15; BC = .35; CD = .3; CE = .15;
rD = [.3 .3 0]; omg1 = [0, 0, 2*pi]; alp1 = [0,0,0];

for phi=0:2*pi/32:2*pi
    hold on, axis equal
    [rB, rC, rE] = pos(phi);
    plot([0, rB(1)], [0, rB(2)], 'r-o');
    plot([rB(1), rC(1)], [rB(2), rC(2)], 'g-o');
    plot([rD(1), rE(1)], [rD(2), rE(2)], 'b-o');
end
```

Output figure



Example 3: Inverted slider-crank Mechanism

Given	$l_{AD} = l_1 = 0.35m$ $l_{BC} = l_2 = 0.20m$ $l_{AC} = 0.15m$ $\theta_1 = 60°$
Find	$\vec{r}_B$, $\vec{r}_D$



Solution

Position of joint B: $\vec{r}_B = x_B\vec{\mathbf{i}} + y_B\vec{\mathbf{j}} = l_{AB}\cos\theta_1\vec{\mathbf{i}} + l_{AB}\sin\theta_1\vec{\mathbf{j}}$
Position of joint C: $\vec{r}_C = x_C\vec{\mathbf{i}} + y_C\vec{\mathbf{j}} = 0.15\vec{\mathbf{i}}$
Position of joint D: $\vec{r}_D = x_D\vec{\mathbf{i}} + y_D\vec{\mathbf{j}} = l_1\cos\theta_1\vec{\mathbf{i}} + l_1\sin\theta_1\vec{\mathbf{j}}$

Solution

Position of joint B: $\vec{\boldsymbol{r}}_{\boldsymbol{B}} = x_B\vec{\mathbf{i}} + y_B\vec{\mathbf{j}} = l_{AB}\cos\theta_1\vec{\mathbf{i}} + l_{AB}\sin\theta_1\vec{\mathbf{j}}$
Position of joint C: $\vec{\boldsymbol{r}}_{\boldsymbol{C}} = x_C\vec{\mathbf{i}} + y_C\vec{\mathbf{j}} = 0.15\vec{\mathbf{i}}$
Position of joint D: $\vec{\boldsymbol{r}}_{\boldsymbol{D}} = x_D\vec{\mathbf{i}} + y_D\vec{\mathbf{j}} = l_1\cos\theta_1\vec{\mathbf{i}} + l_1\sin\theta_1\vec{\mathbf{j}}$

$$\Rightarrow (x_B - x_C)^2 + y_B^2 = l_2^2$$

$$\text{or } (l_{AB}\cos 60^\circ - 0.15)^2 + l_{AB}\sin 60^\circ = l_2^2$$

Solving the system of equations yields $l_{AB} > 0$ and $l_{AB} < 0$. Then, choose $l_{AB} > 0$ and substitute the result into $\vec{\boldsymbol{r}}_{\boldsymbol{B}}$

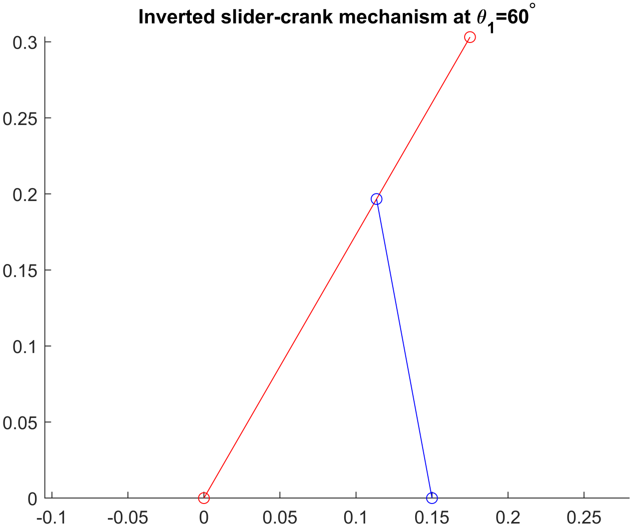
MATLAB R2019a code

```
function [rB, rD] = pos(phi1)
global AD BC rC
syms l_AB

rB = l_AB*[cos(phi1), sin(phi1), 0];
rD = AD*[cos(phi1), sin(phi1), 0];

% find rB
eq = (rB(1)-rC(1))^2 + (rB(2)-rC(2))^2 == BC^2;
sol = solve(eq, l_AB);
if sol(1)>0, rB = double(subs(rB, sol(1)));
else, rB = double(subs(rB, sol(2))); end
end
```

Plotting using MATLAB R2019a

```
global AD BC AC rC
AD = 0.35; BC = 0.20; AC = 0.15; rC = [AC,0,0];
phi = pi/3; omg1 = [0,0,30*pi/30]; alp1 = [0,0,0];

hold on, axis equal
[rB, rD] = pos(pi/3);
plot([0, rD(1)], [0,rD(2)], 'r-o')
plot([rB(1), rC(1)], [rB(2), rC(2)], 'b-o');
```

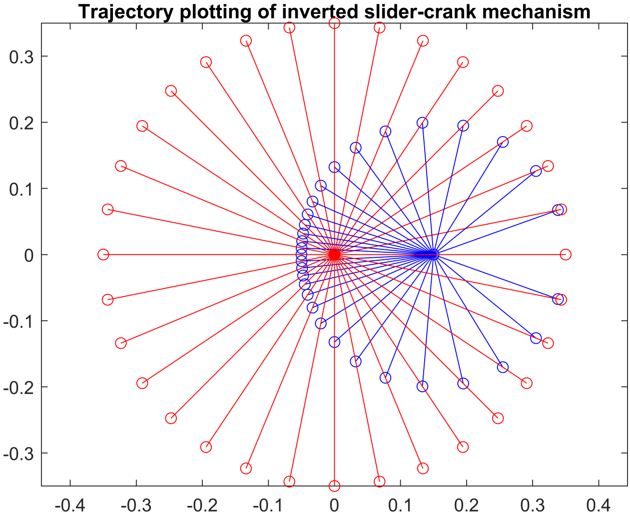
Output figure



Trajectory plotting using MATLAB R2019a

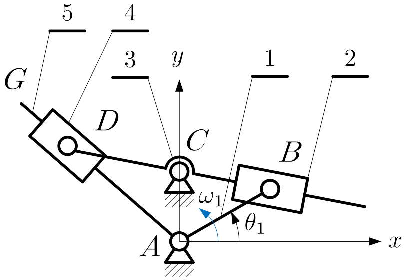
```
global AD BC AC rC
AD = 0.35; BC = 0.20; AC = 0.15; rC = [AC,0,0];
phi = pi/3; omg1 = [0,0,30*pi/30]; alp1 = [0,0,0];

for phi=0:2*pi/32:2*pi
    hold on, axis equal
    [rB, rD] = pos(phi);
    plot([rC(1), rB(1)], [rC(2), rB(2)], 'b-o');
    plot([0, rD(1)], [0, rD(2)], 'r-o');
end
```

Output figure



Example 4: R-RTR-RTR mechanism

Given	$l_{AB} = l_1 = 0.15m$ $l_{AC} = l_2 = 0.1m$ $l_{CD} = l_3 = 0.15m$ $l_{DF} = l_4 = 0.4m$ $l_{AG} = l_5 = 0.3m$ $\theta_1 = 30^\circ$
Find	$\vec{\boldsymbol{r}}_{\boldsymbol{B}}$, $\vec{\boldsymbol{r}}_{\boldsymbol{D}}$, $\vec{\boldsymbol{r}}_{\boldsymbol{G}}$



Solution

Position of joint B: $\vec{\boldsymbol{r}}_{\boldsymbol{B}} = x_B\vec{\mathbf{i}} + y_B\vec{\mathbf{j}} = l_1\cos\theta_1\vec{\mathbf{i}} + l_1\sin\theta_1\vec{\mathbf{j}}$
Position of joint C: $\vec{\boldsymbol{r}}_{\boldsymbol{C}} = x_C\vec{\mathbf{i}} + y_C\vec{\mathbf{j}} = 0.1\vec{\mathbf{j}}$
Position of joint D: $\vec{\boldsymbol{r}}_{\boldsymbol{D}} = x_D\vec{\mathbf{i}} + y_D\vec{\mathbf{j}}$

Position of joint E: $\vec{r}_E = x_E\vec{\mathbf{i}} + y_E\vec{\mathbf{j}}$
Position of joint G: $\vec{r}_G = x_G\vec{\mathbf{i}} + y_G\vec{\mathbf{j}}$

$$\Rightarrow \begin{cases} x_D^2 + (y_D - x_C)^2 = l_3^2 \\ \dfrac{y_D - y_C}{x_D - x_C} = \dfrac{y_D - y_B}{x_D - x_B} \end{cases}$$

Solving the system of equations yields x_{D_1} and x_{D_2}. Since $\theta_1 \in [0, 90)$, the condition is $x_D \leq x_C$

$$\Rightarrow \begin{cases} \theta_2 = \arctan \dfrac{y_B - y_C}{x_B - x_C} \\ \theta_3 = \theta_2 \\ \theta_4 = \arctan \dfrac{y_D}{x_D} \\ \theta_5 = \theta_4 \end{cases}$$

For other an arbitrary angle θ_1, finding the right condition is tricky (*4 conditions* corresponding to 4 quadrants of 1 full rotation).

1^{st} quadrant	2^{nd} quadrant	3^{rd} quadrant	4^{th} quadrant
$x_D \leq x_C = 0$	$x_D \geq x_C = 0$	$x_D \geq x_C = 0$	$x_D \leq x_C = 0$

Table 1: 4 conditions to find x_D from $[0, 2\pi]$

For this mechanism, observe that for any θ_1:

- $x_D x_B < 0$
- $\vec{\boldsymbol{r}}_{\boldsymbol{DC}}$ and $\vec{\boldsymbol{r}}_{\boldsymbol{DF}}$ have the same angle
- $\vec{\boldsymbol{r}}_{\boldsymbol{D}}$ and $\vec{\boldsymbol{r}}_{\boldsymbol{G}}$ have the same angle

Using these properties, we can obtain direct solution of position analysis for the mechanism:

- Find $\vec{\boldsymbol{r}}_{\boldsymbol{D}}$

$$\begin{cases} \langle \vec{\boldsymbol{Ox}}, \vec{\boldsymbol{r}}_{\boldsymbol{CD}} \rangle = \arctan \dfrac{y_B - y_C}{x_B - x_C} + \pi \\ \vec{\boldsymbol{r}}_{\boldsymbol{D}} = \vec{\boldsymbol{r}}_{\boldsymbol{C}} + sign(x_B) l_3 \left(\cos \langle \vec{\boldsymbol{Ox}}, \vec{\boldsymbol{r}}_{\boldsymbol{CD}} \rangle \vec{\mathbf{i}} + \sin \langle \vec{\boldsymbol{Ox}}, \vec{\boldsymbol{r}}_{\boldsymbol{CD}} \rangle \vec{\mathbf{j}} \right) \end{cases}$$

- Find $\vec{r}_F$

$$\begin{cases} \langle \vec{Ox}, \vec{r}_{FC} \rangle = \arctan \dfrac{y_C - y_D}{x_C - x_D} \\ \vec{r}_F = \vec{r}_C + sign(x_B)(l_4 - l_3)\left(\cos \langle \vec{Ox}, \vec{r}_{FC} \rangle \vec{\mathbf{i}} + \sin \langle \vec{Ox}, \vec{r}_{FC} \rangle \vec{\mathbf{j}}\right) \end{cases}$$

- Find $\vec{r}_G$

$$\begin{cases} \langle \vec{Ox}, \vec{r}_G \rangle = \arctan \dfrac{y_D}{x_D} \\ \vec{r}_G = -sign(x_B) l_5 \left(\cos \langle \vec{Ox}, \vec{r}_G \rangle \vec{\mathbf{i}} + \sin \langle \vec{Ox}, \vec{r}_G \rangle \vec{\mathbf{j}}\right) \end{cases}$$

Note that $\arctan \theta \in (-\frac{\pi}{2}, \frac{\pi}{2})$, which leads to the presence of $sign(x_B)$.

MATLAB R2019a code

```
function [rB, rD, rF, rG] = pos(phi1)

global AB DC DF AG rC
syms x y

rB = AB*[cos(phi1), sin(phi1), 0];

% find rD
rD = [x, y, 0];
eq1 = (x-rC(1))^2 + (y - rC(2))^2 == DC^2;
eq2 = (y-rC(2))/(x-rC(1)) == (y-rB(2))/(x-rB(1));
sol = solve(eq1, eq2, x, y);
```

MATLAB R2019a code

```
rD1 = double(subs(rD, [sol.x(1), sol.y(1)]));
rD2 = double(subs(rD, [sol.x(2), sol.y(2)]));
if rD1(1) <= rC(1), rD = rD1; else, rD = rD2; end

% find rF, rG
rFC_angle = atan((rB(2)-rC(2))/(rB(1)-rC(1)));
rG_angle = atan(rD(2)/rD(1)) + pi;
rF = double(rD + DF*[cos(rFC_angle), sin(rFC_angle), 0]);
rG = double(AG*[cos(rG_angle), sin(rG_angle), 0]);

end
```

MATLAB R2019a code (direct solution)

```
function [rB, rD, rF, rG] = pos(phi1)

global AB DC DF AG rC

rB = AB*[cos(phi1), sin(phi1), 0];
pm = sign(rB(1));

% find rD
CB_angle = atan((rB(2)-rC(2)) / (rB(1)-rC(1)))+pi;
rD = double(rC + pm*DC*[cos(CB_angle), sin(CB_angle), 0]);
```

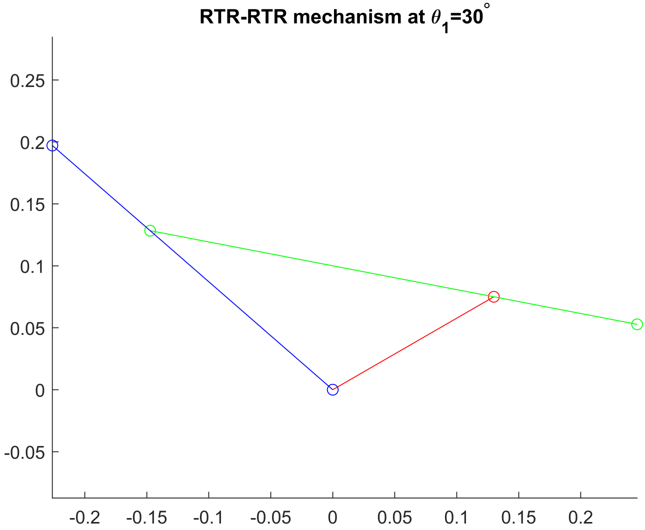
MATLAB R2019a code (direct solution)

```
% find rF
BC_angle = atan((rC(2)-rD(2)) / (rC(1)-rD(1)));
rF = double(rC + pm*(DF-DC)*[cos(BC_angle), sin(BC_angle),
    0]);

% find rG
D_angle = atan(rD(2) / rD(1));
rG = double(-pm*AG*[cos(D_angle), sin(D_angle), 0]);
end
```

Plotting using MATLAB R2019a

```
global AB AC DC DF AG rA rC omg1 alp1
AB = .15; AC = .1; DC = .15; DF = .4; AG = .3;
rA = [0,0,0]; rC = [0,AC,0];
omg1 = [0,0,5/3*pi]; alp1 = [0,0,0];

hold on, axis equal
[rB, rD, rF, rG] = pos(pi/6);
plot([rA(1), rB(1)], [rA(2), rB(2)], 'r-o');
plot([rD(1), rF(1)], [rD(2), rF(2)], 'g-o');
plot([rA(1), rG(1)], [rA(2), rG(2)], 'b-o');
```

Output figure



Trajectory plotting using MATLAB R2019a

```
global AB AC DC DF AG rA rC omg1 alp1
AB = .15; AC = .1; DC = .15; DF = .4; AG = .3;
rA = [0,0,0]; rC = [0,AC,0];
omg1 = [0,0,5/3*pi]; alp1 = [0,0,0];

for phi=0:2*pi/32:2*pi
    hold on, axis equal
    [rB, rD, rF, rG] = pos(phi);
    plot([rA(1), rB(1)], [rA(2), rB(2)], 'r-o');
    plot([rD(1), rF(1)], [rD(2), rF(2)], 'g-o');
    plot([rA(1), rG(1)], [rA(2), rG(2)], 'b-o');
end
```

Output figure