Velocity - Acceleration Analysis

Nguyen Quy Khoi

HCMUT-OISP - Faculty of Mechanical Engineering

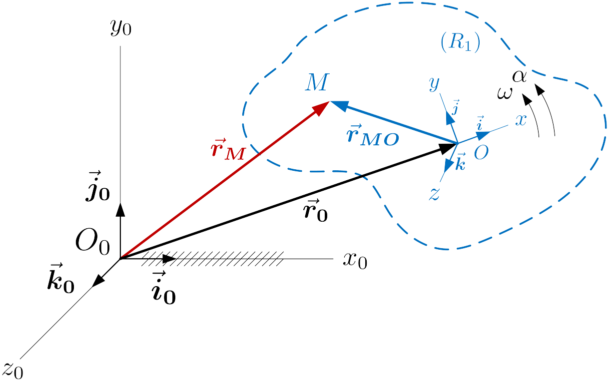
20 February, 2020

Outline

1. Introduction
   - Velocity Field for a Rigid Body
   - Acceleration Field for a Rigid Body
   - Motion of a Point that moves relative to a Rigid Body
   - Independent Contour Method

2. Examples
   - Example: Slider-crank Mechanism
   - Example: Four-bar linkage Mechanism
   - Example: Inverted slider-crank Mechanism

Introduction



Let M be a point in rigid body (R_1). Then $\vec{r}_M$ is the position vector of M relative to fixed reference frame $(O_0x_0y_0z_0)$.

$$\vec{r}_M = \vec{r}_O + \vec{r}_{MO} = \vec{r}_O + x\vec{\mathbf{i}} + y\vec{\mathbf{j}} + z\vec{\mathbf{k}}$$

For Cartesian coordinates, the following relations are true:

$$\begin{cases} \vec{\mathbf{i}} \cdot \vec{\mathbf{i}} = \vec{\mathbf{j}} \cdot \vec{\mathbf{j}} = \vec{\mathbf{k}} \cdot \vec{\mathbf{k}} = 1 \\ \vec{\mathbf{i}} \cdot \vec{\mathbf{j}} = \vec{\mathbf{j}} \cdot \vec{\mathbf{k}} = \vec{\mathbf{k}} \cdot \vec{\mathbf{i}} = 0 \end{cases}$$

Velocity vector $\vec{\boldsymbol{v}}_M = \dot{\vec{\boldsymbol{r}}}_M$ with respect to time t:

$$\vec{\boldsymbol{v}}_M = \vec{\boldsymbol{v}}_O + \dot{x}\vec{\mathbf{i}} + \dot{y}\vec{\mathbf{j}} + \dot{z}\vec{\mathbf{k}} + x\frac{d\vec{\mathbf{i}}}{dt} + y\frac{d\vec{\mathbf{j}}}{dt} + z\frac{d\vec{\mathbf{k}}}{dt}$$

For a rigid body, $\dot{x} = \dot{y} = \dot{z} = 0$

$$\Rightarrow \vec{\boldsymbol{v}}_M = \vec{\boldsymbol{v}}_O + x\frac{d\vec{\mathbf{i}}}{dt} + y\frac{d\vec{\mathbf{j}}}{dt} + z\frac{d\vec{\mathbf{k}}}{dt}$$

Let $\vec{\boldsymbol{\omega}} = \omega_x\vec{\mathbf{i}} + \omega_y\vec{\mathbf{j}} + \omega_z\vec{\mathbf{k}}$ be rotational vector of rigid body (R_1). Then

$$\vec{\boldsymbol{\omega}} = (\frac{d\vec{\mathbf{i}}}{dt}\cdot\vec{\mathbf{j}})\vec{\mathbf{i}} + (\frac{d\vec{\mathbf{j}}}{dt}\cdot\vec{\mathbf{k}})\vec{\mathbf{j}} + (\frac{d\vec{\mathbf{k}}}{dt}\cdot\vec{\mathbf{i}})\vec{\mathbf{k}}$$

$$= (-\vec{\mathbf{i}}\cdot\frac{d\vec{\mathbf{j}}}{dt})\vec{\mathbf{i}} + (-\vec{\mathbf{j}}\cdot\frac{d\vec{\mathbf{k}}}{dt})\vec{\mathbf{j}} + (-\vec{\mathbf{k}}\cdot\frac{d\vec{\mathbf{i}}}{dt})\vec{\mathbf{k}}$$

Combine with the fact that $\frac{d\vec{\mathbf{i}}}{dt}\cdot\vec{\mathbf{i}} = \frac{d\vec{\mathbf{j}}}{dt}\cdot\vec{\mathbf{j}} = \frac{d\vec{\mathbf{k}}}{dt}\cdot\vec{\mathbf{k}} = 0$

$$\frac{d\vec{\mathbf{i}}}{dt} = (\frac{d\vec{\mathbf{i}}}{dt}\cdot\vec{\mathbf{i}})\vec{\mathbf{i}} + (\frac{d\vec{\mathbf{i}}}{dt}\cdot\vec{\mathbf{j}})\vec{\mathbf{j}} + (\frac{d\vec{\mathbf{i}}}{dt}\cdot\vec{\mathbf{k}})\vec{\mathbf{k}} = 0\vec{\mathbf{i}} + \omega_z\vec{\mathbf{j}} - \omega_y\vec{\mathbf{k}} = \vec{\boldsymbol{\omega}}_\times\vec{\mathbf{i}}$$

$$\frac{d\vec{\mathbf{j}}}{dt} = (\frac{d\vec{\mathbf{j}}}{dt}\cdot\vec{\mathbf{i}})\vec{\mathbf{i}} + (\frac{d\vec{\mathbf{j}}}{dt}\cdot\vec{\mathbf{j}})\vec{\mathbf{j}} + (\frac{d\vec{\mathbf{j}}}{dt}\cdot\vec{\mathbf{k}})\vec{\mathbf{k}} = -\omega_z\vec{\mathbf{i}} + 0\vec{\mathbf{j}} + \omega_x\vec{\mathbf{k}} = \vec{\boldsymbol{\omega}}_\times\vec{\mathbf{j}}$$

$$\frac{d\vec{\mathbf{k}}}{dt} = (\frac{d\vec{\mathbf{k}}}{dt}\cdot\vec{\mathbf{i}})\vec{\mathbf{i}} + (\frac{d\vec{\mathbf{k}}}{dt}\cdot\vec{\mathbf{j}})\vec{\mathbf{j}} + (\frac{d\vec{\mathbf{k}}}{dt}\cdot\vec{\mathbf{k}})\vec{\mathbf{k}} = \omega_y\vec{\mathbf{i}} - \omega_x\vec{\mathbf{j}} + 0\vec{\mathbf{k}} = \vec{\boldsymbol{\omega}}_\times\vec{\mathbf{k}}$$

$$\Rightarrow \vec{\boldsymbol{v}} = \vec{\boldsymbol{v}}_{\boldsymbol{O}} + \vec{\boldsymbol{\omega}} \times (x\vec{\mathbf{i}} + y\vec{\mathbf{j}} + z\vec{\mathbf{k}})$$

Formula

For any point of a rigid body, its velocity equation is:

$$\vec{\boldsymbol{v}} = \vec{\boldsymbol{v}}_{\boldsymbol{O}} + \vec{\boldsymbol{\omega}} \times \vec{\boldsymbol{r}}$$

where $\vec{\boldsymbol{\omega}} \times \vec{\boldsymbol{r}}$ is the rotational motion of that point with respect to O

Example

In matrix form, velocity vector $\vec{\boldsymbol{v}}_{\boldsymbol{M}}$ of point M relative to fixed reference frame $(O_0x_0y_0z_0)$ can be written as:

$$\vec{\boldsymbol{v}}_{\boldsymbol{M}} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} = \begin{bmatrix} v_{Ox} + z\omega_y - y\omega_z \\ v_{Oy} + x\omega_z - z\omega_x \\ v_{Oz} + y\omega_x - x\omega_y \end{bmatrix} = \vec{\boldsymbol{v}}_{\boldsymbol{O}} + \vec{\boldsymbol{\omega}} \times \vec{\boldsymbol{r}}_{\boldsymbol{MO}}$$

Acceleration vector $\vec{a} = \ddot{\vec{r}}$ with respect to time t:

$$\vec{a} = \frac{d}{dt}(\vec{v}_O + \vec{\omega} \times \vec{r}) = \dot{\vec{r}}_M + \dot{\vec{\omega}} \times \vec{r} + \vec{\omega} \times \dot{\vec{r}} = \vec{a}_0 + \vec{\alpha} \times \vec{r} + \vec{\omega} \times (\vec{\omega} \times \vec{r})$$

where
$$\begin{aligned}\vec{\alpha} = \dot{\vec{\omega}} &= \dot{\omega}_x \vec{\mathbf{i}} + \dot{\omega}_y \vec{\mathbf{j}} + \dot{\omega}_z \vec{\mathbf{k}} + \omega_x \frac{d\vec{\mathbf{i}}}{dt} + \omega_y \frac{d\vec{\mathbf{j}}}{dt} + \omega_z \frac{d\vec{\mathbf{k}}}{dt} \\ &= \alpha_x \vec{\mathbf{i}} + \alpha_y \vec{\mathbf{j}} + \alpha_z \vec{\mathbf{k}} + \omega_x \vec{\omega} \times \vec{\mathbf{i}} + \omega_y \vec{\omega} \times \vec{\mathbf{j}} + \omega_z \vec{\omega} \times \vec{\mathbf{k}} \\ &= \alpha_x \vec{\mathbf{i}} + \alpha_y \vec{\mathbf{j}} + \alpha_z \vec{\mathbf{k}} + \vec{\omega} \times \vec{\omega} \\ &= \alpha_x \vec{\mathbf{i}} + \alpha_y \vec{\mathbf{j}} + \alpha_z \vec{\mathbf{k}}\end{aligned}$$

Formula

For any point of a rigid body, its acceleration equation is:

$$\vec{a} = \vec{a}_0 + \vec{\alpha} \times \vec{r} + \vec{\omega} \times (\vec{\omega} \times \vec{r})$$

where $\vec{\alpha} \times \vec{r}$ is tangential acceleration
$\vec{\omega} \times (\vec{\omega} \times \vec{r})$ is centripetal acceleration

In matrix form, acceleration vector $\vec{\boldsymbol{a}}_{\boldsymbol{M}}$ of point M relative to fixed reference frame $(O_0x_0y_0z_0)$ can be written as:

$$\vec{\boldsymbol{a}}_{\boldsymbol{M}} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} a_{Ox} + (z\alpha_y - y\alpha_z) + \omega_y(y\omega_x - x\omega_y) + \omega_z(x\omega_x - x\omega_z) \\ a_{Oy} + (x\alpha_z - z\alpha_x) + \omega_z(z\omega_y - y\omega_z) + \omega_x(x\omega_y - y\omega_z) \\ a_{Oz} + (y\alpha_x - x\alpha_y) + \omega_x(x\omega_z - z\omega_x) + \omega_y(y\omega_z - z\omega_y) \end{bmatrix}$$

$$= \vec{\boldsymbol{a}}_{\boldsymbol{O}} + \vec{\boldsymbol{\alpha}} \times \vec{\boldsymbol{r}}_{\boldsymbol{MO}} + \vec{\boldsymbol{\omega}} \times (\vec{\boldsymbol{\omega}} \times \vec{\boldsymbol{r}}_{\boldsymbol{MO}}) = \vec{\boldsymbol{a}}_{\boldsymbol{O}} + \vec{\boldsymbol{a}}^{\boldsymbol{t}}_{\boldsymbol{MO}} + \vec{\boldsymbol{a}}^{\boldsymbol{n}}_{\boldsymbol{MO}}$$

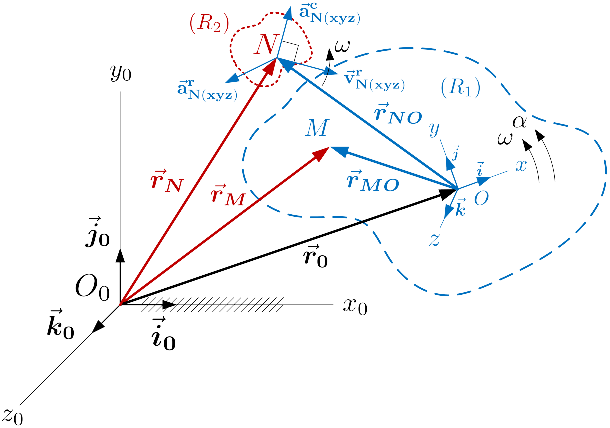
where

$$\vec{\boldsymbol{a}}^{\boldsymbol{t}}_{\boldsymbol{MO}} = \begin{cases} \perp \vec{\boldsymbol{r}}_{\boldsymbol{MO}} \circlearrowright \vec{\boldsymbol{\alpha}} \\ \alpha r_{MO} \end{cases}$$

$$\vec{\boldsymbol{a}}^{\boldsymbol{n}}_{\boldsymbol{MO}} = \begin{cases} \uparrow\uparrow \vec{\boldsymbol{r}}_{\boldsymbol{MO}} \\ \omega^2 r_{MO} = \frac{v^2_{MO}}{r_{MO}} \end{cases}$$

For planar motions: $\vec{\boldsymbol{\omega}} \times (\vec{\boldsymbol{\omega}} \times \vec{\boldsymbol{r}}_{\boldsymbol{MO}}) = -\vec{\boldsymbol{\omega}}^2 \vec{\boldsymbol{r}}_{\boldsymbol{MO}}$

Introduction



Position of point N in rigid body (R_2):

$$\vec{r}_N = \vec{r}_O + \vec{r}_{NO} = \vec{r}_O + x\vec{\mathbf{i}} + y\vec{\mathbf{j}} + z\vec{\mathbf{k}}$$

Velocity vector $\vec{v}_N$ of point N:

$$\begin{aligned}\vec{v}_N &= \vec{v}_O + [\dot{x}\vec{\mathbf{i}} + \dot{y}\vec{\mathbf{j}} + \dot{z}\vec{\mathbf{k}}] + [x\frac{d\vec{\mathbf{i}}}{dt} + y\frac{d\vec{\mathbf{j}}}{dt} + z\frac{d\vec{\mathbf{k}}}{dt}] \\ &= \vec{v}_O + \vec{v}^r_{N(xyz)} + \vec{\omega} \times \vec{r}\end{aligned}$$

where $\vec{v}^r_{N(xyz)}$ is the relative velocity of point N with respect to the reference frame of rigid body (R_1).

Acceleration vector $\vec{a}_N$ of point N:

$$\begin{aligned}\vec{a}_N &= \vec{a}_O + [\ddot{x}\vec{\mathbf{i}} + \ddot{y}\vec{\mathbf{j}} + \ddot{z}\vec{\mathbf{k}}] + 2\vec{\omega} \times \vec{v}^n_{N(xyz)} + \vec{\alpha} \times \vec{r} + \vec{\omega} \times (\vec{\omega} \times \vec{r}) \\ &= \vec{a}_O + \vec{a}^r_{N(xyz)} + \vec{a}^c_{N(xyz)} + \vec{\alpha} \times \vec{r} + \vec{\omega} \times (\vec{\omega} \times \vec{r})\end{aligned}$$

where:

$\vec{a}^r_{N(xyz)}$ is the relative acceleration of point N with respect to the reference frame of rigid body (R_1).

$\vec{a}^c_{N(xyz)}$ is referred to as *Coriolis acceleration*.

For planar motions: $\vec{a}_O = \vec{a}_O + \vec{a}^r_{N(xyz)} + \vec{a}^c_{N(xyz)} + \vec{\alpha} \times \vec{r} - \vec{\omega}^2\vec{r}$

Formulas

For any point N, its velocity and acceleration relative to a reference frame O of a rigid body are:

$$\begin{cases}\vec{v}_N = \vec{v}_O + \vec{v}^r_{NO} + \vec{\omega} \times \vec{r}_{NO} \\ \vec{a}_N = \vec{a}_O + \vec{a}^r_{NO} + 2\vec{\omega} \times \vec{v}^r_{NO} + \vec{\alpha} \times \vec{r}_{NO} + \vec{\omega} \times (\vec{\omega} \times \vec{r}_{NO})\end{cases}$$

Definition

Formula

For a closed mechanism, the number of independent contours N is calculated as follows:

$$N = c - n$$

where
c is the number of joints
n is the number of links, excluding ground

The number of independent contours depends on the number of joint connections to ground. Its graphical representation can be referred to as *connectivity table*, *structural diagram* or ***contour diagram***.

Velocity Analysis

Formulas

For a closed kinematic chain:

- $\sum_i \vec{\boldsymbol{\omega}}_{\boldsymbol{i-1,i}} = \vec{\mathbf{0}}$
- $\sum_i \vec{\boldsymbol{r}}_{\boldsymbol{A_i}} \times \vec{\boldsymbol{\omega}}_{\boldsymbol{i-1,i}} + \sum_i \vec{\boldsymbol{v}}_{\boldsymbol{A_{i-1,i}}} = \vec{\mathbf{0}}$

where:
$\vec{\boldsymbol{\omega}}_{\boldsymbol{i-1,i}}$ is the angular velocity of link i relative to link $i-1$
$\vec{\boldsymbol{r}}_{\boldsymbol{A_i}}$ is the position vector of joint A_i
$\vec{\boldsymbol{v}}_{\boldsymbol{A_{i-1,i}}}$ is the velocity vector of joint A_i on link i relative to joint A_i on link A_{i-1}

Acceleration Analysis

Formulas

For a closed kinematic chain:

- $$\sum_i \vec{\boldsymbol{\alpha}}_{\boldsymbol{i-1,i}} + \sum_i \vec{\boldsymbol{\omega}}_{\boldsymbol{i}} \times \vec{\boldsymbol{\omega}}_{\boldsymbol{i-1,i}} = \vec{\mathbf{0}}$$
- $$\sum_i \vec{\boldsymbol{r}}_{\boldsymbol{A_i}} \times (\vec{\boldsymbol{\alpha}}_{\boldsymbol{i-1,i}} + \vec{\boldsymbol{\omega}}_{\boldsymbol{i}} \times \vec{\boldsymbol{\omega}}_{\boldsymbol{i-1,i}}) + \sum_i \vec{\boldsymbol{a}}_{\boldsymbol{A_{i-1,i}}} + 2\sum_i \vec{\boldsymbol{\omega}}_{\boldsymbol{i-1}} \times \vec{\boldsymbol{v}}_{\boldsymbol{A_{i-1,i}}} + \sum_i \vec{\boldsymbol{\omega}}_{\boldsymbol{i}} \times (\vec{\boldsymbol{\omega}}_{\boldsymbol{i}} \times \vec{\boldsymbol{r}}_{\boldsymbol{A_{i+1}A_i}}) = \vec{\mathbf{0}}$$

where:
$\vec{\boldsymbol{\omega}}_{\boldsymbol{i}}$, $\vec{\boldsymbol{\omega}}_{\boldsymbol{i-1}}$ are the angular velocities of link i and $i-1$ relative to ground
$\vec{\boldsymbol{\alpha}}_{\boldsymbol{i-1,i}}$ is the angular acceleration of joint A_i relative to joint A_{i-1}
$\vec{\boldsymbol{a}}_{\boldsymbol{A_{i-1,i}}}$ is the acceleration vector of joint A_i on link i relative to joint A_i on link A_{i-1}

In planar motions, the above equations for acceleration analysis can be reduced to:

- $\sum_i \vec{\boldsymbol{\alpha}}_{\boldsymbol{i-1,i}} = \vec{\mathbf{0}}$
- $\sum_i \vec{\boldsymbol{r}}_{\boldsymbol{A_i}} \times \vec{\boldsymbol{\alpha}}_{\boldsymbol{i-1,i}} + \sum_i \vec{\boldsymbol{a}}_{\boldsymbol{A_{i-1,i}}} + 2\sum_i \vec{\boldsymbol{\omega}}_{\boldsymbol{i-1}} \times \vec{\boldsymbol{v}}_{\boldsymbol{A_{i-1,i}}} - \sum_i \vec{\boldsymbol{\omega}}_{\boldsymbol{i}}^2 \vec{\boldsymbol{r}}_{\boldsymbol{A_{i+1}A_i}}) = \vec{\mathbf{0}}$

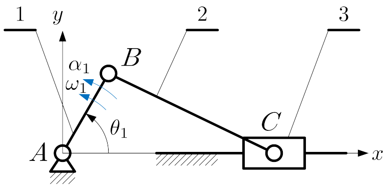
In general, the steps of this method are:

1. Perform position analysis.
2. Draw the contour diagram of the mechanism.
3. Use the formulas above to obtain the results.
4. Calculate $\vec{\boldsymbol{\omega}}_{\boldsymbol{i}}$ and $\vec{\boldsymbol{a}}_{\boldsymbol{i}}$ with respect to ground.
5. Compute relevant velocities and accelerations.

Normal Approach

Example 1: Slider-crank Mechanism

Given	$l_{AB} = l_1 = 0.5m$ $l_{BC} = l_2 = 1m$ $\theta_1 = 60^\circ$ $\vec{\boldsymbol{\omega}}_\mathbf{1} = 1\vec{\mathbf{k}}\ (rad/s)$ $\vec{\boldsymbol{\alpha}}_\mathbf{1} = -1\vec{\mathbf{k}}\ (rad/s^2)$
Find	$\vec{\boldsymbol{v}}_{\boldsymbol{B}}$, $\vec{\boldsymbol{v}}_{\boldsymbol{C}}$, $\vec{\boldsymbol{a}}_{\boldsymbol{B}}$, $\vec{\boldsymbol{a}}_{\boldsymbol{C}}$, $\vec{\boldsymbol{\omega}}_\mathbf{2}$, $\vec{\boldsymbol{\alpha}}_\mathbf{2}$



Solution

Using position analysis / MATLAB,

$\Rightarrow \vec{\boldsymbol{r}}_{\boldsymbol{B}} = 0.25\vec{\mathbf{i}} + 0.433\vec{\mathbf{j}}\ (m), \vec{\boldsymbol{r}}_{\boldsymbol{C}} = 1.1514\vec{\mathbf{i}}\ (m)$

$\vec{\boldsymbol{r}}_{\boldsymbol{CB}} = \vec{\boldsymbol{r}}_{\boldsymbol{C}} - \vec{\boldsymbol{r}}_{\boldsymbol{B}} = 0.9014\vec{\mathbf{i}} - 0.433\vec{\mathbf{j}}\ (m)$

- Find $\vec{\boldsymbol{v}}_{\boldsymbol{B}}$

$$\vec{\boldsymbol{v}}_{\boldsymbol{B}} = \vec{\boldsymbol{\omega}}_{\boldsymbol{1}} \times \vec{\boldsymbol{r}}_{\boldsymbol{B}} = -0.433\vec{\mathbf{i}} + 0.25\vec{\mathbf{j}}\ (m/s)$$

- Find $\vec{\boldsymbol{v}}_{\boldsymbol{C}}$, $\vec{\boldsymbol{\omega}}_{\boldsymbol{2}}$

$$\vec{\boldsymbol{v}}_{\boldsymbol{C}} = \vec{\boldsymbol{v}}_{\boldsymbol{B}} + \vec{\boldsymbol{v}}_{\boldsymbol{CB}} = \vec{\boldsymbol{v}}_{\boldsymbol{B}} + \vec{\boldsymbol{\omega}}_{\boldsymbol{2}} \times \vec{\boldsymbol{r}}_{\boldsymbol{CB}}$$

Projecting onto x, y-axes, we obtain the solution:

$$\Rightarrow \begin{cases} v_{Cx} = -0.5531\ (m/s) \\ \omega_{2z} = -0.2774\ (rad/s) \end{cases}$$

$$\Rightarrow \begin{cases} \vec{\boldsymbol{v}}_{\boldsymbol{C}} = -0.5531\vec{\mathbf{i}}\ (m/s) \\ \vec{\boldsymbol{\omega}}_{\boldsymbol{2}} = -0.2774\vec{\mathbf{k}}\ (rad/s) \end{cases}$$

- Find $\vec{a}_B$, $\vec{a}^n_{CB}$

$$\vec{a}_B = \vec{\alpha}_1 \times \vec{r}_B - \vec{\omega}_1^2 \vec{r}_B = 0.183\vec{\mathbf{i}} - 0.683\vec{\mathbf{j}}\ (m/s^2)$$
$$\vec{a}^n_{CB} = -\vec{\omega}_2^2 \vec{r}_{CB} = -0.0693\vec{\mathbf{i}} + 0.0333\vec{\mathbf{j}}\ (m/s^2)$$

- Find $\vec{a}_C$, $\vec{\alpha}_2$

$$\vec{a}_C = \vec{a}_B + \vec{a}_{CB} = \vec{a}_B + \vec{\alpha}_2 \times \vec{r}_{CB} - \vec{\omega}_2^2 \vec{r}_{CB}$$

Projecting onto x, y-axes, we obtain the solution:

$$\Rightarrow \begin{cases} a_{Cx} = 0.4258\ (m/s^2) \\ \alpha_{2z} = 0.7208\ (rad/s^2) \end{cases}$$
$$\Rightarrow \begin{cases} \vec{a}_C = 0.4258\vec{\mathbf{i}}\ (m/s^2) \\ \vec{\alpha}_2 = 0.7208\vec{\mathbf{k}}\ (rad/s^2) \end{cases}$$

MATLAB R2019a code

```
function [vB, vC, omg2] = vel(theta1)
global omg1, syms x y
% set variables
[rB, rC] = pos(theta1);
vB = cross(omg1, rB);
vC = [x,0,0]; omg2 = [0,0,y];
% find vC, omg2
eq = vB + cross(omg2, rC-rB) == vC;
sol = solve([eq(1),eq(2)], [x,y]);
% return the values
vC = double(subs(vC, sol.x));
omg2 = double(subs(omg2, sol.y));
end
```

MATLAB R2019a code

```
function [aB, aC, alp2] = acc(theta1)
global omg1 alp1, syms x y
% set variables
[rB, rC] = pos(theta1);
[~,~, omg2] = vel(theta1);
aB = cross(alp1, rB) - norm(omg1)^2*rB;
aC = [x,0,0]; alp2 = [0,0,y];
% find aC, alp2
eq = aB + cross(alp2, rC-rB) - omg2(3)^2*(rC-rB) == aC;
sol = solve(eq(1), eq(2), x, y);
% return the values
aC = double(subs(aC, sol.x));
alp2 = double(subs(alp2, sol.y));
end
```

Derivative Method

Let $\theta_1 = \theta(t)$ be a function of time t. We perform position analysis to find $\vec{\boldsymbol{r}}_{\boldsymbol{B}}(\theta(t))$ and $\vec{\boldsymbol{r}}_{\boldsymbol{C}}(\theta(t))$. Using MATLAB, we find out that:

$$\vec{\boldsymbol{r}}_{\boldsymbol{B}}(\theta(t)) = \frac{\cos\theta(t)}{2}\vec{\mathbf{i}} + \frac{\sin\theta(t)}{2}\vec{\mathbf{j}}$$

$$\vec{\boldsymbol{r}}_{\boldsymbol{C}}(\theta(t)) = \left(\frac{cos\theta(t)}{2} + \frac{\sqrt{2-\sin\theta(t)}\sqrt{\sin\theta(\theta(t))+2)}}{2}\right)\vec{\mathbf{i}}$$

Using the results above to obtain the angle $\theta_2(\theta(t))$ of link 2:

$$\theta_2(\theta(t)) = \arctan\left(-\frac{\sin\theta(t)}{\sqrt{2-\sin\theta(t))}\sqrt{\sin\theta(t)+2}}\right)$$

Then, find velocities and accelerations by taking first and second derivative of $\vec{\boldsymbol{r}}_{\boldsymbol{B}}$, $\vec{\boldsymbol{r}}_{\boldsymbol{C}}$, θ_2, respectively:

$$\begin{cases} \vec{\boldsymbol{v}}_{\boldsymbol{B}}(\theta(t)) = \dot{\vec{\boldsymbol{r}}}_{\boldsymbol{B}} \\ \vec{\boldsymbol{v}}_{\boldsymbol{C}}(\theta(t)) = \dot{\vec{\boldsymbol{r}}}_{\boldsymbol{C}} \\ \vec{\boldsymbol{\omega}}_{\boldsymbol{2}}(\theta(t)) = \dot{\theta}_2 \vec{\mathbf{k}} \end{cases}, \begin{cases} \vec{\boldsymbol{a}}_{\boldsymbol{B}}(\theta(t)) = \ddot{\vec{\boldsymbol{r}}}_{\boldsymbol{B}} \\ \vec{\boldsymbol{a}}_{\boldsymbol{C}}(\theta(t)) = \ddot{\vec{\boldsymbol{r}}}_{\boldsymbol{B}} \\ \vec{\boldsymbol{\alpha}}_{\boldsymbol{2}}(\theta(t)) = \ddot{\theta}_2 \vec{\mathbf{k}} \end{cases}$$

Let $\theta(t) = 60^\circ$, we obtain the final results.

MATLAB R2019a code

```
function [rB, rC] = d_pos(theta1)
global AB BC, syms x theta(t)
% set variables
rB = AB*[cos(theta(t)), sin(theta(t)), 0];
rC = [x,0,0];
% find rC
eq = (rB(1)-x)^2 + rB(2)^2 == BC^2;
sol = solve(eq, x);
if subs(sol(1),theta,theta1) > subs(sol(2),theta,theta1)
    rC = subs(rC, sol(1));
else
    rC = subs(rC, sol(2));
end
end
```

MATLAB R2019a code

```
function [vB, vC, omg2] = d_vel(theta1)
[rB, rC] = d_pos(theta1); theta2 = atan((rC(2)-rB(2))/(rC(1)-
    rB(1)));
% find vB, vC, omg2
vB = diff(rB); vC = diff(rC); omg2 = [0,0,diff(theta2)];
end
function [aB, aC, alp2] = d_acc(theta1)
[vB, vC, omg2] = d_vel(theta1);
% find aB, aC, alp2
aB = diff(vB); aC = diff(vC); alp2 = diff(omg2);
end
```

Independent Contour Method

Using the solution above,

$\vec{r}_B = 0.25\vec{\mathbf{i}} + 0.433\vec{\mathbf{j}}\ (m), \vec{r}_C = 1.1514\vec{\mathbf{i}}\ (m)$
$\vec{r}_{CB} = 0.9014\vec{\mathbf{i}} - 0.433\vec{\mathbf{j}}\ (m)$, $\vec{a}_B = 0.183\vec{\mathbf{i}} - 0.683\vec{\mathbf{j}}\ (m/s^2)$

Applying velocity formulas to find $\vec{\omega}_{12} = \omega_{12z}\vec{\mathbf{k}}$, $\vec{\omega}_{23} = \omega_{23z}\vec{\mathbf{k}}$,
$\vec{v}_{23} = v_{23x}\vec{\mathbf{i}}\vec{\mathbf{k}}$:

$$\begin{cases} \vec{\omega}_{01} + \vec{\omega}_{12} + \vec{\omega}_{23} = \vec{0} \\ \vec{r}_B \times \vec{\omega}_{12} + \vec{r}_C \times \vec{\omega}_{23} + \vec{v}_{23} = \vec{0} \end{cases}$$

Solving the system of equations by projecting onto x, y-axes, we obtain:

$\vec{\omega}_{12} = -1.2774\vec{\mathbf{k}}\ (rad/s)$, $\vec{\omega}_{23} = 0.2774\vec{\mathbf{k}}\ (rad/s)$, $\vec{v}_{23} = 0.5531\vec{\mathbf{i}}\ (m/s)$

The results is consistent with the 2 previous methods.

Again, using acceleration formulas and notice that there exists $\vec{a}_{23}$ corresponding to slider 2:

$$\begin{cases} \vec{\alpha}_{01} + \vec{\alpha}_{12} + \vec{\alpha}_{23} = \vec{0} \\ \vec{r}_B \times \vec{\alpha}_{12} + \vec{r}_C \times \vec{\alpha}_{23} + \vec{a}_{23} - \vec{\omega}_1^2 \vec{r}_B - \vec{\omega}_2^2 \vec{r}_{CB} = \vec{0} \end{cases}$$

Solving the system of equations by projecting onto x, y-axes, we obtain:

$\vec{\alpha}_{12} = 1.7208\vec{\mathbf{k}}\ (rad/s^2)$, $\vec{\alpha}_{23} = -0.7208\vec{\mathbf{k}}\ (rad/s^2)$,
$\vec{a}_{23} = -0.4258\vec{\mathbf{i}}\ (m/s^2)$

The results is consistent with the 2 previous methods.

MATLAB R2019a code

```
function [vB, vC, omg2, v23] = c_vel(theta1)
global omg1, syms x y z
% set variables
[rB, rC] = pos(theta1); vB = cross(omg1, rB);
omg12 = [0,0,x]; omg23 = [0,0,y]; v23 = [z,0,0];
% find omg12, omg23, v23
eq1 = omg1 + omg12 + omg23 == 0;
eq2 = cross(rB, omg12) + cross(rC, omg23) + v23 == 0;
sol = solve([eq1(3),eq2(1),eq2(2)], [x,y,z]);
```

MATLAB R2019a code

```
% return the values
omg12 = double(subs(omg12, sol.x));
omg23 = double(subs(omg23, sol.y));
v23 = double(subs(v23, sol.z));
vC = -v23;
omg2 = omg1 + omg12;
end
```

MATLAB R2019a code

```
function [aB, aC, alp2] = c_acc(theta1)
global omg1 alp1, syms x y z
% set variables
[rB, rC] = pos(theta1);
[~,~,omg2,~] = c_vel(theta1);
aB = cross(alp1,rB) - omg1(3)^2*rB;
alp12 = [0,0,x]; alp23 = [0,0,y]; a23 = [z,0,0];
% find alp12, alp23, a23
eq1 = alp1 + alp12 + alp23 == 0;
eq2 = cross(rB,alp12) + cross(rC,alp23) + a23 - omg1(3)^2*rB
    - omg2(3)^2*(rC-rB) == 0;
sol = solve([eq1(3),eq2(1),eq2(2)], [x,y,z]);
```

MATLAB R2019a code

```
% return the values
alp12 = double(subs(alp12, sol.x));
alp23 = double(subs(alp23, sol.y));
a23 = double(subs(a23, sol.z));
aC = −a23;
alp2 = alp1 + alp12;
end
```

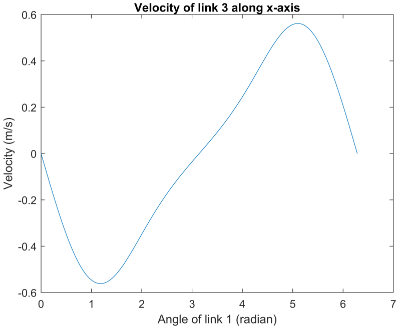
Velocity - Acceleration plotting

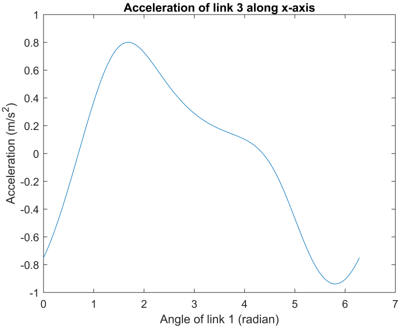
```
global AB BC omg1 alp1
AB = .5; BC = 1; omg1 = [0,0,1]; alp1 = [0,0,-1];

i = 0; theta1s = zeros(201, 1);
vCs = theta1s; aCs = theta1s;

for theta1=0:(2*pi/200):2*pi
    i=i+1; theta1s(i)=phi;
    [~,vC] = vel(theta1); vCs(i) = vC(1);
    [~,aC] = acc(theta1); aCs(i) = aC(1);
end
plot(theta1s, vCs)
plot(theta1s, aCs)
```

Output figures

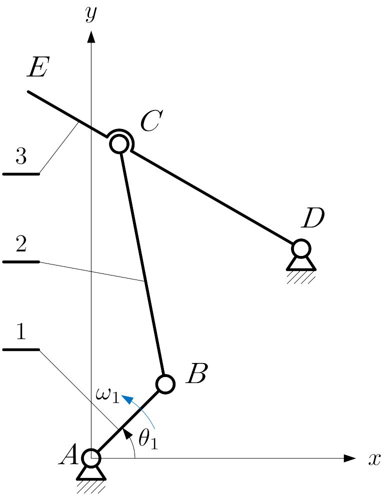




Normal Approach

Example 2: Four-bar linkage Mechanism

Given	$l_{AB} = l_1 = 0.15m$ $l_{BC} = l_2 = 0.35m$ $l_{CD} = l_3 = 0.3m$ $l_{CE} = l_4 = 0.15m$ $\vec{r}_D = 0.3\vec{\mathbf{i}} + 0.3\vec{\mathbf{j}}\ (m)$ $\theta_1 = 45°$ $\vec{\omega}_1 = 1\vec{\mathbf{k}}\ (rad/s)$ $\vec{\alpha}_1 = \vec{0}\ (rad/s^2)$
Find	$\vec{v}_B$, $\vec{v}_C$, $\vec{v}_E$, $\vec{\omega}_2$, $\vec{\omega}_3$, $\vec{a}_B$, $\vec{a}_C$, $\vec{a}_E$, $\vec{\alpha}_2$, $\vec{\alpha}_3$



Solution

Using position analysis / MATLAB,

$\vec{r}_B = 0.1061\vec{\mathbf{i}} + 0.1061\vec{\mathbf{j}}\ (m)$
$\vec{r}_C = 0.0401\vec{\mathbf{i}} + 0.4498\vec{\mathbf{j}}\ (m)$
$\vec{r}_E = -0.0899\vec{\mathbf{i}} + 0.5247\vec{\mathbf{j}}\ (m)$

$\vec{r}_{CD} = -0.2599\vec{\mathbf{i}} + 0.1498\vec{\mathbf{j}}\ (m)$
$\vec{r}_{ED} = -0.3899\vec{\mathbf{i}} + 0.2247\vec{\mathbf{j}}\ (m)$
$\vec{r}_{CB} = -0.066\vec{\mathbf{i}} + 0.3437\vec{\mathbf{j}}\ (m)$

- Find $\vec{v}_B$

  $$\vec{v}_B = \vec{\omega}_1 \times \vec{r}_B = -0.6664\vec{\mathbf{i}} + 0.6664\vec{\mathbf{j}}\ (m/s)$$

- Find $\vec{\omega}_2$, $\vec{\omega}_3$

  $$\vec{v}_C = \vec{v}_B + \vec{v}_{CB} \Leftrightarrow \vec{\omega}_3 \times \vec{r}_{CD} = \vec{v}_B + \vec{\omega}_2 \times \vec{r}_{CB}$$

  Projecting onto x, y-axes, we obtain the solution:

  $$\Rightarrow \begin{cases} \omega_{3z} = -3.4364\ (rad/s) \\ \omega_{2z} = -3.4364\ (rad/s) \end{cases}$$

  $$\Rightarrow \begin{cases} \vec{\omega}_3 = -3.4364\vec{\mathbf{k}}\ (rad/s) \\ \vec{\omega}_2 = -3.4364\vec{\mathbf{k}}\ (rad/s) \end{cases}$$

- Find $\vec{v}_C$, $\vec{v}_E$

  $$\vec{v}_C = \vec{\omega}_3 \times \vec{r}_{CD} = 0.5147\vec{\mathbf{i}} + 0.8932\vec{\mathbf{j}}\ (m/s),$$
  $$\vec{v}_E = \vec{\omega}_3 \times \vec{r}_{ED} = 0.7721\vec{\mathbf{i}} + 1.3398\vec{\mathbf{j}}\ (m/s)$$

- Find $\vec{a}_B$, $\vec{a}^n_{CB}$, $\vec{a}^n_{CD}$

$$\vec{a}_B = \vec{\alpha}_1 \times \vec{r}_B - \vec{\omega}_1^2 \vec{r}_B = -4.1873\vec{\mathbf{i}} - 4.1873\vec{\mathbf{j}}\ (m/s^2)$$
$$\vec{a}^n_{CB} = -\vec{\omega}_3^2 \vec{r}_{CB} = 0.7793\vec{\mathbf{i}} - 4.0589\vec{\mathbf{j}}\ (m/s^2)$$
$$\vec{a}^n_{CD} = -\vec{\omega}_3^2 \vec{r}_{CD} = 3.0695\vec{\mathbf{i}} - 1.7688\vec{\mathbf{j}}\ (m/s^2)$$

- Find $\vec{\alpha}_2 = \alpha_{2z}\vec{\mathbf{k}}$, $\vec{\alpha}_3 = \alpha_{3z}\vec{\mathbf{k}}$

$$\vec{a}_C = \vec{a}_B + \vec{a}_{CB} \Leftrightarrow \vec{a}^t_{CD} + \vec{a}^n_{CD} = \vec{a}_B + \vec{\alpha}_2 \times \vec{r}_{CB} + \vec{a}^n_{CB}$$

Projecting onto x, y-axes, we obtain the solution:

$$\Rightarrow \begin{cases} \alpha_{3z} = -8.9788\ (rad/s^2) \\ \alpha_{2z} = 22.6402\ (rad/s^2) \end{cases} \quad \Rightarrow \begin{cases} \vec{\alpha}_3 = -8.9788\vec{\mathbf{k}}\ (rad/s^2) \\ \vec{\alpha}_2 = 22.6402\vec{\mathbf{k}}\ (rad/s^2) \end{cases}$$

- Find $\vec{a}_C$, $\vec{a}_E$

$$\vec{a}_C = \vec{\alpha}_3 \times \vec{r}_{CD} - \vec{\omega}_3^2 \vec{r}_{CD} = -0.3218\vec{\mathbf{i}} - 7.6537\vec{\mathbf{j}}\ (m/s^2)$$
$$\vec{a}_E = \vec{\alpha}_3 \times \vec{r}_{ED} - \vec{\omega}_3^2 \vec{r}_{ED} = -0.4827\vec{\mathbf{i}} - 11.4805\vec{\mathbf{j}}\ (m/s^2)$$

MATLAB R2019a code

```
function [vB, vC, vE, omg2, omg3] = vel(theta1)
global omg1 rD, syms x y
% set variables
[rB, rC, rE] = pos(theta1);
omg2 = [0,0,x]; omg3 = [0,0,y];
vB = cross(omg1, rB);
vC = cross(omg3, rC-rD);
vE = cross(omg3, rE-rD);
```

MATLAB R2019a code

```
% find omg2, omg3
eq = vB + cross(omg2, rC-rB) == vC;
sol = solve([eq(1),eq(2)], [x,y]);
% return the values
omg2 = double(subs(omg2, sol.x));
omg3 = double(subs(omg3, sol.y));
vC = double(subs(vC, sol.y));
vE = double(subs(vE, sol.y));
end
```

MATLAB R2019a code

```
function [aB, aC, aE, alp2, alp3] = acc(theta1)
global omg1 rD alp1, syms x y
% set variables
[rB, rC, rE] = pos(theta1);
[~,~,~, omg2, omg3] = vel(theta1);
alp2 = [0,0,x]; alp3 = [0,0,y];
aB = cross(alp1, rB) - omg1(3)^2*rB;
aC = cross(alp3, rC-rD) - omg3(3)^2*(rC-rD);
aE = cross(alp3, rE-rD) - omg3(3)^2*(rE-rD);
```

MATLAB R2019a code

```
% find alp2, alp3
eq = aB + cross(alp2, rC-rB) - omg2(3)^2*(rC-rB) == aC;
sol = solve([eq(1),eq(2)], [x,y]);
% find aC, aE
aC = double(subs(aC, sol.y));
aE = double(subs(aE, sol.y));
alp2 = double(subs(alp2, sol.x));
alp3 = double(subs(alp3, sol.y));
end
```

Derivative Method

Let $\theta_1 = \theta(t)$ be a function of time t. We perform position analysis with MATLAB to find $\vec{\boldsymbol{r}}_{\boldsymbol{B}}(\theta(t))$, $\vec{\boldsymbol{r}}_{\boldsymbol{C}}(\theta(t))$ and $\vec{\boldsymbol{r}}_{\boldsymbol{E}}(\theta(t))$. Obtain the angle $\theta_2(\theta(t))$ of link 2 and $\theta_3(\theta(t))$ of link 3 using the results above.

Then, find velocities and accelerations by taking first and second derivative of $\vec{\boldsymbol{r}}_{\boldsymbol{B}}$, $\vec{\boldsymbol{r}}_{\boldsymbol{C}}$, $\vec{\boldsymbol{r}}_{\boldsymbol{E}}$, θ_2, θ_3 respectively:

$$\begin{cases} \vec{\boldsymbol{v}}_{\boldsymbol{B}}(\theta(t)) = \dot{\vec{\boldsymbol{r}}}_{\boldsymbol{B}} \\ \vec{\boldsymbol{v}}_{\boldsymbol{C}}(\theta(t)) = \dot{\vec{\boldsymbol{r}}}_{\boldsymbol{C}} \\ \vec{\boldsymbol{a}}_{\boldsymbol{E}}(\theta(t)) = \dot{\vec{\boldsymbol{r}}}_{\boldsymbol{E}} \\ \vec{\boldsymbol{\omega}}_{\boldsymbol{2}}(\theta(t)) = \dot{\theta}_2\vec{\mathbf{k}} \\ \vec{\boldsymbol{\omega}}_{\boldsymbol{3}}(\theta(t)) = \dot{\theta}_3\vec{\mathbf{k}} \end{cases}, \begin{cases} \vec{\boldsymbol{a}}_{\boldsymbol{B}}(\theta(t)) = \ddot{\vec{\boldsymbol{r}}}_{\boldsymbol{C}} \\ \vec{\boldsymbol{a}}_{\boldsymbol{C}}(\theta(t)) = \ddot{\vec{\boldsymbol{r}}}_{\boldsymbol{C}} \\ \vec{\boldsymbol{a}}_{\boldsymbol{E}}(\theta(t)) = \dot{\vec{\boldsymbol{r}}}_{\boldsymbol{E}} \\ \vec{\boldsymbol{\alpha}}_{\boldsymbol{2}}(\theta(t)) = \ddot{\theta}_2\vec{\mathbf{k}} \\ \vec{\boldsymbol{\alpha}}_{\boldsymbol{3}}(\theta(t)) = \ddot{\theta}_3\vec{\mathbf{k}} \end{cases}$$

Let $\theta(t) = 45^\circ$, we obtain the final results.

MATLAB R2019a code

```
function [rB, rC, rE] = d_pos(theta1)
global AB BC CD CE rD, syms x y theta(t)
% set variables
rB = AB*[cos(theta(t)), sin(theta(t)), 0];
rC = [x, y, 0]; rE = [x, y, 0];
% find rC
eq1 = (rB(1)-x)^2 + (rB(2)-y)^2 == BC^2;
eq2 = (x-rD(1))^2 + (y-rD(2))^2 == CD^2;
sol = solve([eq1,eq2], [x,y]);
if 0>subs(sol.x(1), theta, theta1)
    rC = subs(rC, [x,y], [sol.x(1),sol.y(1)]);
else
    rC = subs(rC, [x,y], [sol.x(2),sol.y(2)]);
end
```

MATLAB R2019a code

```
% find rE
eq1 = (rD(2)-y)/(rD(1)-x) == (rD(2)-rC(2))/(rD(1)-rC(1));
eq2 = (rC(1)-x)^2 + (rC(2)-y)^2 == CE^2;
sol = solve([eq1,eq2], [x,y]);
if subs(rC(1), theta, theta1) > subs(sol.x(1), theta, theta1)
    rE = subs(rE, [x, y], [sol.x(1), sol.y(1)]);
else
    rE = subs(rE, [x, y], [sol.x(2), sol.y(2)]);
end
end
```

MATLAB R2019a code

```
function [vB, vC, vE, omg2, omg3] = d_vel(theta1)
global rD
[rB, rC, rE] = d_pos(theta1);
theta2 = atan((rC(2)-rB(2))/(rC(1)-rB(1)));
theta3 = atan((rC(2)-rD(2))/(rC(1)-rD(1)));
% find vB, vC, vE, omg2, omg3
vB = diff(rB); vC = diff(rC); vE = diff(rE);
omg2 = [0,0,diff(theta2)]; omg3 = [0,0,diff(theta3)];
end
function [aB, aC, aE, alp2, alp3] = d_acc(theta1)
[vB, vC, vE, omg2, omg3] = d_vel(theta1);
% find aB, aC, aE, alp2, alp3
aB = diff(vB); aC = diff(vC); aE = diff(vE);
alp2 = diff(omg2); alp3 = diff(omg3);
end
```

Independent Contour Method

Using the solution above,

$\vec{r}_B = 0.1061\vec{\mathbf{i}} + 0.1061\vec{\mathbf{j}}\ (m)$
$\vec{r}_C = 0.0401\vec{\mathbf{i}} + 0.4498\vec{\mathbf{j}}\ (m)$
$\vec{r}_E = -0.0899\vec{\mathbf{i}} + 0.5247\vec{\mathbf{j}}\ (m)$

$\vec{r}_{ED} = -0.3899\vec{\mathbf{i}} + 0.2247\vec{\mathbf{j}}\ (m)$
$\vec{r}_{DC} = 0.2599\vec{\mathbf{i}} - 0.1498\vec{\mathbf{j}}\ (m)$
$\vec{r}_{CB} = -0.066\vec{\mathbf{i}} + 0.3437\vec{\mathbf{j}}\ (m)$

$\vec{a}_B = 0.183\vec{\mathbf{i}} - 0.683\vec{\mathbf{j}}\ (m/s^2)$

Applying velocity formulas to find $\vec{\boldsymbol{\omega}}_{\mathbf{12}} = \omega_{12z}\vec{\mathbf{k}}$, $\vec{\boldsymbol{\omega}}_{\mathbf{23}} = \omega_{23z}\vec{\mathbf{k}}$, $\vec{\boldsymbol{\omega}}_{\mathbf{30}} = \omega_{30z}\vec{\mathbf{k}}$:

$$\begin{cases} \vec{\boldsymbol{\omega}}_{\mathbf{01}} + \vec{\boldsymbol{\omega}}_{\mathbf{12}} + \vec{\boldsymbol{\omega}}_{\mathbf{23}} + \vec{\boldsymbol{\omega}}_{\mathbf{30}} = \vec{\mathbf{0}} \\ \vec{\boldsymbol{r}}_{\boldsymbol{B}} \times \vec{\boldsymbol{\omega}}_{\mathbf{12}} + \vec{\boldsymbol{r}}_{\boldsymbol{C}} \times \vec{\boldsymbol{\omega}}_{\mathbf{23}} + \vec{\boldsymbol{r}}_{\boldsymbol{D}} \times \vec{\boldsymbol{\omega}}_{\mathbf{30}} = \vec{\mathbf{0}} \end{cases}$$

Solving the system of equations by projecting onto x, y-axes, we obtain:

$\vec{\boldsymbol{\omega}}_{\mathbf{12}} = -9.7196\vec{\mathbf{k}}\ (rad/s)$, $\vec{\boldsymbol{\omega}}_{\mathbf{23}} = \vec{\mathbf{0}}\ (rad/s)$, $\vec{\boldsymbol{\omega}}_{\mathbf{30}} = 3.4364\vec{\mathbf{k}}\ (rad/s)$

The results is consistent with the 2 previous methods.

Again, using acceleration formulas to find $\vec{\boldsymbol{\alpha}}_{\mathbf{12}} = \alpha_{12z}\vec{\mathbf{k}}$, $\vec{\boldsymbol{\alpha}}_{\mathbf{23}} = \alpha_{23z}\vec{\mathbf{k}}$, $\vec{\boldsymbol{\alpha}}_{\mathbf{30}} = \alpha_{30z}\vec{\mathbf{k}}$:

$$\begin{cases} \vec{\boldsymbol{\alpha}}_{\mathbf{01}} + \vec{\boldsymbol{\alpha}}_{\mathbf{12}} + \vec{\boldsymbol{\alpha}}_{\mathbf{23}} + \vec{\boldsymbol{\alpha}}_{\mathbf{30}} = \vec{\mathbf{0}} \\ \vec{\boldsymbol{r}}_{\boldsymbol{B}} \times \vec{\boldsymbol{\alpha}}_{\mathbf{12}} + \vec{\boldsymbol{r}}_{\boldsymbol{C}} \times \vec{\boldsymbol{\alpha}}_{\mathbf{23}} + \vec{\boldsymbol{r}}_{\boldsymbol{D}} \times \vec{\boldsymbol{\alpha}}_{\mathbf{30}} - \vec{\boldsymbol{\omega}}_{\mathbf{1}}^{2}\vec{\boldsymbol{r}}_{\boldsymbol{B}} - \vec{\boldsymbol{\omega}}_{\mathbf{2}}^{2}\vec{\boldsymbol{r}}_{\boldsymbol{CB}} - \vec{\boldsymbol{\omega}}_{\mathbf{3}}^{2}\vec{\boldsymbol{r}}_{\boldsymbol{DC}} = \vec{\mathbf{0}} \end{cases}$$

Solving the system of equations by projecting onto x, y-axes, we obtain:

$\vec{\boldsymbol{\alpha}}_{\mathbf{12}} = -8.9788\vec{\mathbf{k}}\ (rad/s^2)$
$\vec{\boldsymbol{\alpha}}_{\mathbf{23}} = 31.619\vec{\mathbf{k}}\ (rad/s^2)$
$\vec{\boldsymbol{\alpha}}_{\mathbf{30}} = -22.6402\vec{\mathbf{k}}\ (rad/s^2)$

The results is consistent with the 2 previous methods.

MATLAB R2019a code

```
function [vB, vC, vE, omg2, omg3] = c_vel(theta1)
global rD omg1, syms x y z
% set variables
[rB, rC, rE] = pos(theta1); vB = cross(omg1, rB);
omg12 = [0,0,x]; omg23 = [0,0,y]; omg30 = [0,0,z];
% find omg12, omg23, omg30
eq1 = omg1 + omg12 + omg23 + omg30 == 0;
eq2 = cross(rB, omg12) + cross(rC, omg23) + cross(rD, omg30)
    == 0;
sol = solve([eq1(3),eq2(1),eq2(2)], [x,y,z]);
```

MATLAB R2019a code

```
% return the values
omg12 = double(subs(omg12, sol.x));
omg23 = double(subs(omg23, sol.y));
omg30 = double(subs(omg30, sol.z));
omg2 = omg1 + omg12; omg3 = -omg30;
vC = cross(omg3, rC-rD);
vE = cross(omg3, rE-rD);
end
```

MATLAB R2019a code

```
function [aB, aC, aE, alp2, alp3] = c_acc(theta1)
global omg1 alp1 rD, syms x y z
% set variables
[rB, rC, rE] = pos(theta1);
[~,~,~, omg2, omg3] = vel(theta1);
aB = cross(alp1, rB) - omg1(3)^2*rB;
alp12 = [0,0,x]; alp23 = [0,0,y]; alp30 = [0,0,z];
% find alp12, alp23, alp30
eq1 = alp1 + alp12 + alp23 + alp30 == 0;
eq2 = cross(rB, alp12) + cross(rC, alp23) + cross(rD, alp30)
   - omg1(3)^2*rB - omg2(3)^2*(rC-rB) - omg3(3)^2*(rD-rC) ==
    0;
sol = solve([eq1(3),eq2(1),eq2(2)], [x,y,z]);
```

MATLAB R2019a code

```
% return the values
alp12 = double(subs(alp12, sol.x));
alp23 = double(subs(alp23, sol.y));
alp30 = double(subs(alp30, sol.z));
alp2 = alp1 + alp12; alp3 = -alp30;
aC = cross(alp3, rC-rD) - omg3(3)^2*(rC-rD);
aE = cross(alp3, rE-rD) - omg3(3)^2*(rE-rD);
end
```

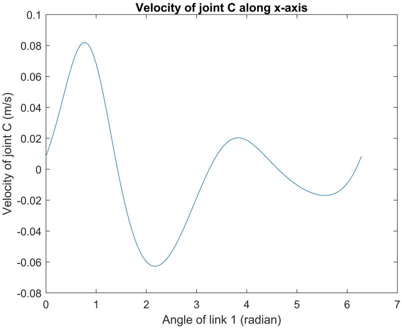
Velocity - Acceleration plotting

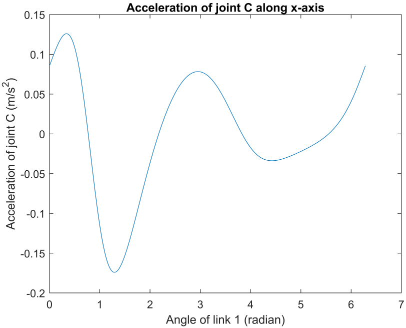
```
global AB BC CD CE rD omg1 alp1
AB = .15; BC = .35; CD = .3; CE = .15;
rD = [.3,.3,0]; omg1 = [0,0,1]; alp1 = [0,0,0];

i=0; theta1s = zeros(201,1);
aCs = theta1s; vCs = theta1s;

for theta1=0:(2*pi/200):2*pi
    i=i+1;
    [~,vC] = vel(theta1); vCs(i) = vC(1);
    [~,aC] = acc(theta1); aCs(i) = aC(1);
    theta1s(i)=theta1;
end
plot(theta1s, vCs)
plot(theta1s, aCs)
```

Output figures

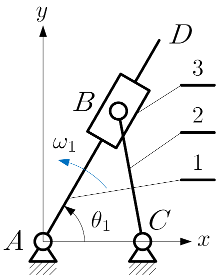




Normal Approach

Example 1: Inverted slider-crank Mechanism

Given	$l_{AC} = 0.15m$ $l_{BC} = 0.2m$ $l_{AD} = 0.35m$ $\theta_1 = 60°$ $\vec{\boldsymbol{\omega}}_{\mathbf{1}} = \pi\vec{\mathbf{k}}\ (rad/s)$ $\vec{\boldsymbol{\alpha}}_{\mathbf{1}} = \vec{\mathbf{0}}\ (rad/s^2)$
Find	$\vec{\boldsymbol{v}}_{\boldsymbol{B_3}}$, $\vec{\boldsymbol{v}}_{\boldsymbol{D}}$, $\vec{\boldsymbol{a}}_{\boldsymbol{B_3}}$, $\vec{\boldsymbol{a}}_{\boldsymbol{D}}$, $\vec{\boldsymbol{\omega}}_{\mathbf{2}}$, $\vec{\boldsymbol{\alpha}}_{\mathbf{2}}$



Solution

Using position analysis / MATLAB,

$\Rightarrow \vec{\boldsymbol{r}}_{\boldsymbol{B}} = 0.1135\vec{\mathbf{i}} + 0.1966\vec{\mathbf{j}}\ (m)$, $\vec{\boldsymbol{r}}_{\boldsymbol{C}} = 0.15\vec{\mathbf{i}}\ (m)$

$$\vec{\boldsymbol{r}}_{\boldsymbol{BC}} = \vec{\boldsymbol{r}}_{\boldsymbol{B}} - \vec{\boldsymbol{r}}_{\boldsymbol{C}} = -0.0365\vec{\mathbf{i}} + 0.1966\vec{\mathbf{j}}\ (m)$$

- Find $\vec{\boldsymbol{v}}_{\boldsymbol{B_1}}$

  $$\vec{\boldsymbol{v}}_{\boldsymbol{B_1}} = \vec{\boldsymbol{\omega}}_{\mathbf{1}} \times \vec{\boldsymbol{r}}_{\boldsymbol{B}} = -0.6178\vec{\mathbf{i}} + 0.3567\vec{\mathbf{j}}\ (m)$$

- Find $\vec{\boldsymbol{v}}_{\boldsymbol{B_2B_1}} = v_{Bx}\vec{\mathbf{i}} + v_{By}\vec{\mathbf{j}}$, $\vec{\boldsymbol{\omega}}_{\mathbf{3}} = \vec{\boldsymbol{\omega}}_{\mathbf{2}} = \omega_{3z}\vec{\mathbf{k}}$

  $$\begin{cases} \vec{\boldsymbol{v}}_{\boldsymbol{B_2}} = \vec{\boldsymbol{v}}_{\boldsymbol{B_1}} + \vec{\boldsymbol{v}}_{\boldsymbol{B_2B_1}} \Leftrightarrow \vec{\boldsymbol{\omega}}_{\mathbf{3}} \times \vec{\boldsymbol{r}}_{\boldsymbol{BC}} = \vec{\boldsymbol{v}}_{\boldsymbol{B_1}} + \vec{\boldsymbol{v}}_{\boldsymbol{B_2B_1}} \\ (\vec{\boldsymbol{r}}_{\boldsymbol{B}} \times \vec{\mathbf{i}}) \times \vec{\boldsymbol{r}}_{\boldsymbol{B}} \cdot \vec{\boldsymbol{v}}_{\boldsymbol{B_2}} = 0 \end{cases}$$

  Projecting onto x, y-axes, we obtain the solution:

  $$\Rightarrow \begin{cases} v_{Bx} = -0.3047\ (m/s) \\ v_{By} = -0.5277\ (m/s) \\ \omega_{3z} = 4.691\ (rad/s) \end{cases}$$

  $$\Rightarrow \begin{cases} \vec{\boldsymbol{v}}_{\boldsymbol{B_2B_1}} = -0.3047\vec{\mathbf{i}} - 0.5277\vec{\mathbf{j}}\ (m/s) \\ \vec{\boldsymbol{\omega}}_{\mathbf{3}} = \vec{\boldsymbol{\omega}}_{\mathbf{2}} = 4.691\vec{\mathbf{k}}\ (rad/s) \end{cases}$$

- Find $\vec{\boldsymbol{v}}_{\boldsymbol{B}} = \vec{\boldsymbol{v}}_{\boldsymbol{B_2}}$, $\vec{\boldsymbol{v}}_{\boldsymbol{D}}$

  $\vec{\boldsymbol{v}}_{\boldsymbol{B}} = \vec{\boldsymbol{v}}_{\boldsymbol{B_2}} = \vec{\boldsymbol{\omega}}_{\boldsymbol{3}} \times \vec{\boldsymbol{r}}_{\boldsymbol{BC}} = -0.9225\vec{\mathbf{i}} - 0.1711\vec{\mathbf{j}}\ (m/s)$
  $\vec{\boldsymbol{v}}_{\boldsymbol{D}} = \vec{\boldsymbol{\omega}}_{\boldsymbol{1}} \times \vec{\boldsymbol{r}}_{\boldsymbol{BC}} = -0.9522\vec{\mathbf{i}} + 0.5498\vec{\mathbf{j}}\ (m/s)$

- Find $\vec{\boldsymbol{a}}_{\boldsymbol{B_1}}$, $\vec{\boldsymbol{a}}^{\boldsymbol{n}}_{\boldsymbol{B_2}}$, $\vec{\boldsymbol{a}}^{\boldsymbol{c}}_{\boldsymbol{B_2B_1}}$

$$\vec{\boldsymbol{a}}_{\boldsymbol{B_1}} = \vec{\boldsymbol{\alpha}}_{\boldsymbol{1}} \times \vec{\boldsymbol{r}}_{\boldsymbol{B}} - \vec{\boldsymbol{\omega}}_{\boldsymbol{1}}^2 \vec{\boldsymbol{r}}_{\boldsymbol{B}} = -1.1205\vec{\mathbf{i}} - 1.9408\vec{\mathbf{j}}\ (m/s^2)$$
$$\vec{\boldsymbol{a}}^{\boldsymbol{n}}_{\boldsymbol{B_2}} = -\vec{\boldsymbol{\omega}}_{\boldsymbol{3}}^2 \vec{\boldsymbol{r}}_{\boldsymbol{BC}} = 0.8024\vec{\mathbf{i}} - 4.3274\vec{\mathbf{j}}\ (m/s^2)$$
$$\vec{\boldsymbol{a}}^{\boldsymbol{c}}_{\boldsymbol{B_2B_1}} = 2\vec{\boldsymbol{\omega}}_{\boldsymbol{1}} \times \vec{\boldsymbol{v}}_{\boldsymbol{B_2B_1}} = 3.3159\vec{\mathbf{i}} - 1.9144\vec{\mathbf{j}}\ (m/s^2)$$

- Find $\vec{\boldsymbol{a}}^{\boldsymbol{r}}_{\boldsymbol{B_2B_1}} = a_{Bx}\vec{\mathbf{i}} + a_{By}\vec{\mathbf{j}}$, $\vec{\boldsymbol{\alpha}}_{\boldsymbol{3}} = \vec{\boldsymbol{\alpha}}_{\boldsymbol{2}} = \alpha_{3z}\vec{\mathbf{k}}$

$$\begin{cases} \vec{\boldsymbol{a}}_{\boldsymbol{B_2}} = \vec{\boldsymbol{a}}_{\boldsymbol{B_1}} + \vec{\boldsymbol{a}}_{\boldsymbol{B_2B_1}} \\ \Leftrightarrow \vec{\boldsymbol{\alpha}}_{\boldsymbol{3}} \times \vec{\boldsymbol{r}}_{\boldsymbol{BC}} + \vec{\boldsymbol{a}}^{\boldsymbol{n}}_{\boldsymbol{B_2}} = \vec{\boldsymbol{a}}_{\boldsymbol{B_1}} + \vec{\boldsymbol{a}}^{\boldsymbol{c}}_{\boldsymbol{B_2B_1}} + \vec{\boldsymbol{a}}^{\boldsymbol{r}}_{\boldsymbol{B_2B_1}} \\ (\vec{\boldsymbol{r}}_{\boldsymbol{B}} \times \vec{\mathbf{i}}) \times \vec{\boldsymbol{r}}_{\boldsymbol{B}} \cdot \vec{\boldsymbol{a}}^{\boldsymbol{r}}_{\boldsymbol{B_2B_1}} = 0 \end{cases}$$

Projecting onto x, y-axes, we obtain the solution:

$$\Rightarrow \begin{cases} a_{Bx} = -0.1382 \ (m/s^2) \\ a_{By} = -0.2394 \ (m/s^2) \\ \alpha_{3z} = -6.3802 \ (rad/s^2) \end{cases}$$

$$\Rightarrow \begin{cases} \vec{\boldsymbol{a}}^{r}_{\boldsymbol{B_2B_1}} = -0.1382\vec{\mathbf{i}} - 0.2394\vec{\mathbf{j}} \ (m/s^2) \\ \vec{\boldsymbol{\alpha}}_{\boldsymbol{3}} = \vec{\boldsymbol{\alpha}}_{\boldsymbol{2}} = -6.3802\vec{\mathbf{k}} \ (rad/s^2) \end{cases}$$

- Find $\vec{\boldsymbol{a}}_{\boldsymbol{B}} = \vec{\boldsymbol{a}}_{\boldsymbol{B_2}}$, $\vec{\boldsymbol{a}}_{\boldsymbol{D}}$

  $\vec{\boldsymbol{a}}_{\boldsymbol{B}} = \vec{\boldsymbol{a}}_{\boldsymbol{B_2}} = \vec{\boldsymbol{\alpha}}_{\boldsymbol{3}} \times \vec{\boldsymbol{r}}_{\boldsymbol{BC}} - \vec{\boldsymbol{\omega}}^2_{\boldsymbol{3}}\vec{\boldsymbol{r}}_{\boldsymbol{BC}} = 2.0571\vec{\mathbf{i}} - 4.0947\vec{\mathbf{j}} \ (m/s^2)$
  $\vec{\boldsymbol{a}}_{\boldsymbol{D}} = \vec{\boldsymbol{\alpha}}_{\boldsymbol{1}} \times \vec{\boldsymbol{r}}_{\boldsymbol{D}} - \vec{\boldsymbol{\omega}}^2_{\boldsymbol{1}}\vec{\boldsymbol{r}}_{\boldsymbol{D}} = -1.7272\vec{\mathbf{i}} - 2.9916\vec{\mathbf{j}} \ (m/s^2)$

MATLAB R2019a code

```
function [vB, vD, omg3, vB2B1] = vel(theta1)
global rC omg1, syms x y z
% set variables
[rB, rD] = pos(theta1);
vB2B1 = [x,y,0]; omg3 = [0,0,z];
vB = cross(omg3, rB-rC); vD = cross(omg1, rD);
% find vB2B1, omg3=omg2
eq1 = cross(omg3, rB-rC) == vB2B1 + cross(omg1, rB);
eq2 = dot(cross(cross(rB,[1,0,0]),rB), vB2B1) == 0;
sol = solve([eq1(1), eq1(2), eq2], [x,y,z]);
% return the values
vB2B1 = double(subs(vB2B1, [x, y], [sol.x, sol.y]));
omg3 = double(subs(omg3, sol.z));
vB = double(subs(vB, sol.z));
end
```

MATLAB R2019a code

```
function [aB, aD, alp3] = acc(theta1)
global rC alp1 omg1, syms x y z
% set variables
[rB, rD] = pos(theta1); [~,~, omg3, vB2B1] = vel(theta1);
aB2B1 = [x,y,0]; alp3 = [0,0,z];
aB1 = cross(alp1, rB) - omg1(3)^2*rB;
aD = cross(alp1, rD) - omg1(3)^2*rD;
aB2 = cross(alp3, rB-rC) - omg3(3)^2*(rB-rC);
```

MATLAB R2019a code

```
% find aB2B1, alp3=alp2
eq1 = aB2 == aB1 + aB2B1 + 2*cross(omg1, vB2B1);
eq2 = dot(cross(cross(rB,[1,0,0]),rB), aB2B1) == 0;
sol = solve([eq1(1), eq1(2), eq2], [x,y,z]);
% return the values
aB2B1 = double(subs(aB2B1, [x,y], [sol.x, sol.y]));
aB = double(subs(aB2, sol.z));
alp3 = double(subs(alp3, sol.z));
end
```

Derivative Method

Let $\theta_1 = \theta(t)$ be a function of time t. We perform position analysis to find $\vec{\boldsymbol{r}}_{\boldsymbol{B}}(\theta(t))$ and $\vec{\boldsymbol{r}}_{\boldsymbol{D}}(\theta(t))$. Using MATLAB, we find out that:

$$\vec{\boldsymbol{r}}_{\boldsymbol{B}}(\theta(t)) = \frac{3\cos\theta(t) + \sqrt{9\cos\theta(t)^2 + 7}}{20}(\cos\theta(t)\vec{\mathbf{i}} + \sin\theta(t)\vec{\mathbf{j}})$$

$$\vec{\boldsymbol{r}}_{\boldsymbol{D}}(\theta(t)) = \frac{7}{20}\cos\theta(t)\vec{\mathbf{i}} + \frac{7}{20}\sin\theta(t)\vec{\mathbf{j}}$$

Using the results above to obtain the angle $\theta_3(\theta(t))$ of link 3:

$$\theta_3(\theta(t)) = \arctan\left(\cot\theta(t)\frac{3\cos\theta(t) + \sqrt{9\cos\theta(t)^2 + 7}}{\sqrt{9\cos\theta(t)^2 + 7} + 3\cos\theta(t)^2 - 3}\right)$$

Then, find velocities and accelerations by taking first and second derivative of $\vec{\boldsymbol{r}}_{\boldsymbol{B}}$, $\vec{\boldsymbol{r}}_{\boldsymbol{D}}$, θ_2, respectively:

$$\begin{cases} \vec{\boldsymbol{v}}_{\boldsymbol{B_2}}(\theta(t)) = \dot{\vec{\boldsymbol{r}}}_{\boldsymbol{B}} \\ \vec{\boldsymbol{v}}_{\boldsymbol{D}}(\theta(t)) = \dot{\vec{\boldsymbol{r}}}_{\boldsymbol{D}} \\ \vec{\boldsymbol{\omega}}_{\boldsymbol{3}}(\theta(t)) = \dot{\theta}_3 \vec{\mathbf{k}} \end{cases}, \begin{cases} \vec{\boldsymbol{a}}_{\boldsymbol{B_2}}(\theta(t)) = \ddot{\vec{\boldsymbol{r}}}_{\boldsymbol{B}} \\ \vec{\boldsymbol{a}}_{\boldsymbol{D}}(\theta(t)) = \ddot{\vec{\boldsymbol{r}}}_{\boldsymbol{D}} \\ \vec{\boldsymbol{\alpha}}_{\boldsymbol{3}}(\theta(t)) = \ddot{\theta}_3 \vec{\mathbf{k}} \end{cases}$$

Let $\theta(t) = 60^\circ$, we obtain the final results.

MATLAB R2019a code

```
function [rB, rD] = d_pos(theta1)
global AD BC rC, syms l_AB theta(t)
% set variables
rB = l_AB*[cos(theta(t)), sin(theta(t)), 0];
rD = AD*[cos(theta(t)), sin(theta(t)), 0];
% find rB
eq = (rB(1)-rC(1))^2 + (rB(2)-rC(2))^2 == BC^2;
sol = solve(eq, l_AB);
if double(subs(sol(1), theta, theta1))>0
    rB = subs(rB, l_AB, sol(1));
else
    rB = subs(rB, l_AB, sol(2)); end
end
```

MATLAB R2019a code

```
function [vB, vD, omg3] = d_vel(theta1)
global rC
[rB, rD] = d_pos(theta1);
theta3 = atan((rB(2)-rC(2))/(rB(1)-rC(1)));
% find vB, vD, omg3
vB = diff(rB); vD = diff(rD);
omg3 = [0,0,diff(theta3)];
end
function [aB, aD, alp3] = d_acc(theta1)
[vB, vD, omg3] = d_vel(theta1);
% find aB, aD, alp3
aB = diff(vB); aD = diff(vD); alp3 = diff(omg3);
end
```

Independent Contour Method

Using the solution above,

$\vec{\boldsymbol{r}}_{\boldsymbol{B}} = 0.1135\vec{\mathbf{i}} + 0.1966\vec{\mathbf{j}}\ (m)$, $\vec{\boldsymbol{r}}_{\boldsymbol{C}} = 0.15\vec{\mathbf{i}}\ (m)$
$\vec{\boldsymbol{r}}_{\boldsymbol{BC}} = -0.0365\vec{\mathbf{i}} + 0.1966\vec{\mathbf{j}}\ (m)$, $\vec{\boldsymbol{a}}_{\boldsymbol{B_1}} = -1.1205\vec{\mathbf{i}} - 1.9408\vec{\mathbf{j}}\ (m/s^2)$

Applying velocity formulas to find $\vec{\boldsymbol{v}}_{\mathbf{12}} = v_{12x}\vec{\mathbf{i}} + v_{12y}\vec{\mathbf{j}}$, $\vec{\boldsymbol{\omega}}_{\mathbf{23}} = \omega_{23z}\vec{\mathbf{k}}$, $\vec{\boldsymbol{\omega}}_{\mathbf{30}} = \omega_{30z}\vec{\mathbf{k}}$:

$$\begin{cases} \vec{\boldsymbol{\omega}}_{\mathbf{01}} + \vec{\boldsymbol{\omega}}_{\mathbf{23}} + \vec{\boldsymbol{\omega}}_{\mathbf{30}} = \vec{\mathbf{0}} \\ \vec{\boldsymbol{r}}_{\boldsymbol{B}} \times \vec{\boldsymbol{\omega}}_{\mathbf{23}} + \vec{\boldsymbol{r}}_{\boldsymbol{C}} \times \vec{\boldsymbol{\omega}}_{\mathbf{30}} + \vec{\boldsymbol{v}}_{\mathbf{12}} = \vec{\mathbf{0}} \\ (\vec{\boldsymbol{r}}_{\boldsymbol{B}} \times \vec{\mathbf{i}}) \times \vec{\boldsymbol{r}}_{\boldsymbol{B}} \cdot \vec{\boldsymbol{v}}_{\mathbf{12}} = 0 \end{cases}$$

Solving the system of equations by projecting onto x, y-axes, we obtain:

$\vec{\boldsymbol{\omega}}_{\mathbf{23}} = 1.5494\vec{\mathbf{k}}\ (rad/s)$, $\vec{\boldsymbol{\omega}}_{\mathbf{30}} = -4.691\vec{\mathbf{k}}\ (rad/s)$,
$\vec{\boldsymbol{v}}_{\mathbf{12}} = -0.3047\vec{\mathbf{i}} - 0.5277\vec{\mathbf{j}}\ (m/s)$

The results is consistent with the 2 previous methods.

Again, using acceleration formulas $\vec{\boldsymbol{a}}_{\mathbf{12}} = a_{12x}\vec{\mathbf{i}} + a_{12y}\vec{\mathbf{j}}$, $\vec{\boldsymbol{\alpha}}_{\mathbf{23}} = \alpha_{23z}\vec{\mathbf{k}}$, $\vec{\boldsymbol{\alpha}}_{\mathbf{30}} = \alpha_{30z}\vec{\mathbf{k}}$:

$$\begin{cases} \vec{\boldsymbol{\alpha}}_{\mathbf{01}} + \vec{\boldsymbol{\alpha}}_{\mathbf{23}} + \vec{\boldsymbol{\alpha}}_{\mathbf{30}} = \vec{\mathbf{0}} \\ \vec{\boldsymbol{r}}_{\boldsymbol{B}} \times \vec{\boldsymbol{\alpha}}_{\mathbf{23}} + \vec{\boldsymbol{r}}_{\boldsymbol{C}} \times \vec{\boldsymbol{\alpha}}_{\mathbf{30}} + \vec{\boldsymbol{a}}_{\mathbf{12}} - \vec{\boldsymbol{\omega}}_{\mathbf{1}}^2 \vec{\boldsymbol{r}}_{\boldsymbol{B}} - \vec{\boldsymbol{\omega}}_{\mathbf{3}}^2 \vec{\boldsymbol{r}}_{\boldsymbol{CB}} = \vec{\mathbf{0}} \\ (\vec{\boldsymbol{r}}_{\boldsymbol{B}} \times \vec{\mathbf{i}}) \times \vec{\boldsymbol{r}}_{\boldsymbol{B}} \cdot \vec{\boldsymbol{a}}_{\mathbf{12}} = 0 \end{cases}$$

Solving the system of equations by projecting onto x, y-axes, we obtain:

$\vec{\boldsymbol{\alpha}}_{\mathbf{23}} = -6.3802\vec{\mathbf{k}}\ (rad/s^2)$, $\vec{\boldsymbol{\alpha}}_{\mathbf{30}} = 6.3802\vec{\mathbf{k}}\ (rad/s^2)$,
$\vec{\boldsymbol{a}}_{\mathbf{12}} = -0.1382\vec{\mathbf{i}} - 0.2394\vec{\mathbf{j}}\ (m/s^2)$

The results is consistent with the 2 previous methods.

MATLAB R2019a code

```
function [vB, vD, omg3, v12] = c_vel(theta1)
global rC omg1, syms x y z t
% set variables
[rB, rD] = pos(theta1);
omg23 = [0,0,x]; omg30 = [0,0,y]; v12 = [z,t,0];
% find omg23, omg30, a12
eq1 = omg1 + omg23 + omg30 == 0;
eq2 = cross(rB, omg23) + cross(rC, omg30) + v12 == 0;
eq3 = dot(cross(cross(rB, [1,0,0]), rB), v12) == 0;
sol = solve([eq1(3), eq2(1), eq2(2), eq3], [x,y,z,t]);
```

MATLAB R2019a code

```
% return the results
omg23 = double(subs(omg23, x, sol.x));
omg30 = double(subs(omg30, y, sol.y));
v12 = double(subs(v12, [z,t], [sol.z,sol.t]));
omg3 = omg1 + omg23;
vB = cross(omg3, rB-rC);
vD = cross(omg1, rD);
end
```

MATLAB R2019a code

```
function [aB, aD, alp3] = c_acc(theta1)
global rC omg1 alp1, syms x y z t
% set variables
[rB, rD] = pos(theta1); [vB, vD, omg2, v12] = c_vel(theta1);
alp23 = [0,0,x]; alp30 = [0,0,y]; a12 = [z,t,0];
% find alp23, alp30, a12
eq1 = alp1 + alp23 + alp30 == 0;
eq2 = cross(rB, alp23) + cross(rC, alp30) + a12 + 2*cross(
    omg1, v12) - omg2(3)^2*(rC-rB) - omg1(3)^2*rB == 0;
eq3 = dot(cross(cross(rB, [1,0,0]), rB), a12) == 0;
sol = solve([eq1(3), eq2(1), eq2(2), eq3], [x,y,z,t]);
```

MATLAB R2019a code

```
% return the results
alp23 = double(subs(alp23, x, sol.x));
alp30 = double(subs(alp30, y, sol.y));
a12 = double(subs(a12, [z,t], [sol.z,sol.t]));
alp3 = alp1 + alp23;
aB = cross(alp3, rB-rC)-omg2(3)^2*(rB-rC);
aD = cross(alp1, rD)-omg1(3)^2*rD;
end
```

Velocity - Acceleration plotting

```
global AD BC AC rC
AD = 0.35; BC = 0.20; AC = 0.15; rC = [AC,0,0];
theta1 = pi/3; omg1 = [0,0,pi]; alp1 = [0,0,0];

i=0; theta1s = zeros(201, 1);
vBs = theta1s; aBs = theta1s;

for theta1=0:(2*pi/200):2*pi
    i=i+1; theta1s(i) = theta1;
    vB2 = vel(theta1); vBs(i) = vB2(1);
    aB2 = acc(theta1); aBs(i) = aB2(1);
end
plot(theta1s, vBs)
plot(theta1s, aBs)
```

Output figures