Week 5 – Jacobians: Velocities and Static Forces

Advanced Robotic Systems – MANU2453

Dr Ehsan Asadi, School of Engineering
RMIT University, Victoria, Australia
Email: ehsan.asadi@rmit.edu.au

Lectures

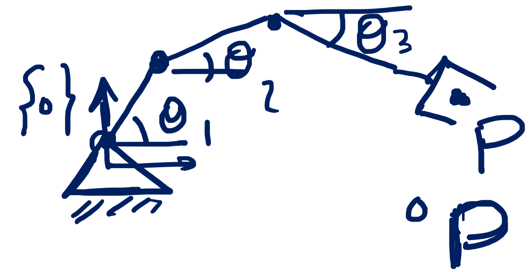
Wk	Date	Lecture (NOTE: video recording)	Maths Difficulty	Hands-on Activity	Related Assessment
1	24/7	• Introduction to the Course • Spatial Descriptions & Transformations			
2	31/7	• Spatial Descriptions & Transformations • Robot Cell Design			Robot Cell Design Assignment
3	7/8	• Forward Kinematics • Inverse Kinematics			
4	14/8	• ABB Robot Programming via Teaching Pendant • ABB RobotStudio Offline Programming		ABB RobotStudio Offline Programming	Offline Programming Assignment
5	21/8	• Jacobians: Velocities and Static Forces			
6	28/8	• Manipulator Dynamics			
7	11/9	• Manipulator Dynamics		MATLAB Simulink Simulation	
8	18/9	• Robotic Vision		MATLAB Simulation	Robotic Vision Assignment
9	25/9	• Robotic Vision		MATLAB Simulation	
10	2/10	• Trajectory Generation			
11	9/10	• Linear & Nonlinear Control		MATLAB Simulink Simulation	
12	16/10	• Introduction to I4.0 • Revision			Final Exam

Content

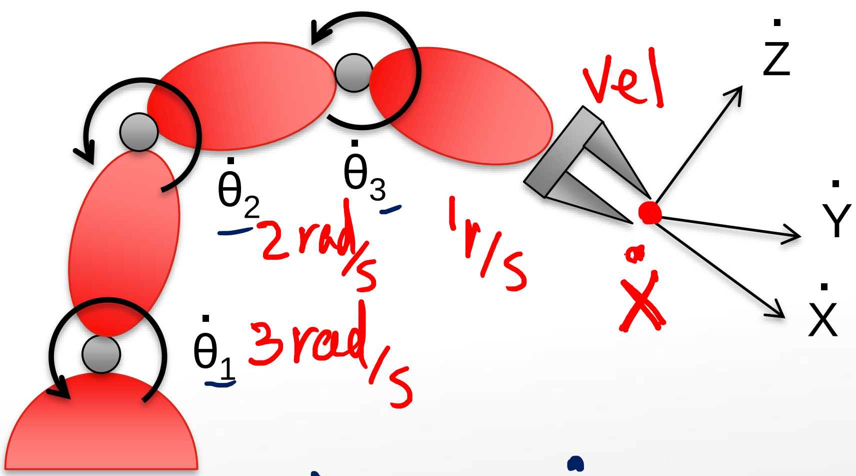
- Introduction - Jacobian
- Method 1 - Direct differentiation (for Linear Jacobian)
- Method 2 - Velocity Propagation from Link to Link
- Method 3 - Explicit Form (For your reading (not included in exam))
- Static Forces in Manipulators
- Singularities

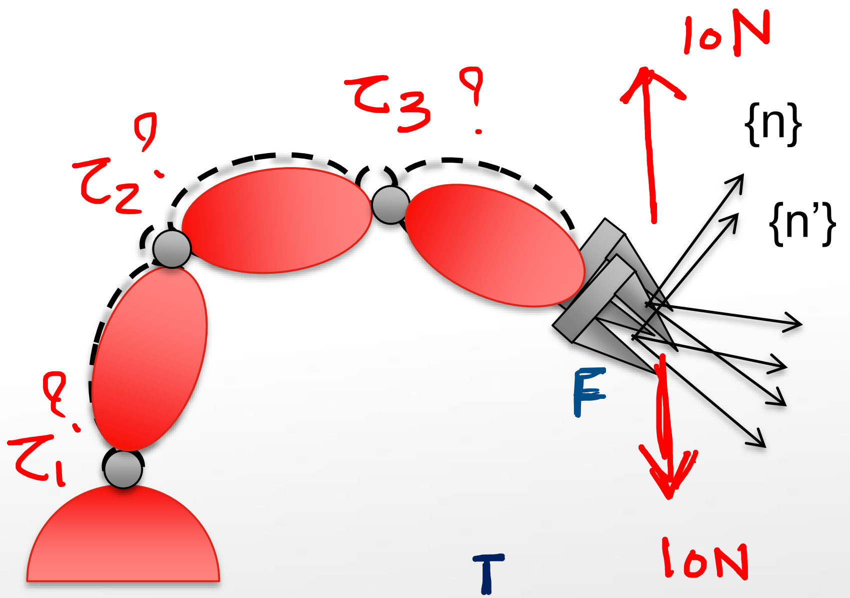
Introduction

Fwd-Kin



- In this lecture, we will learn:
  - Relationship between joint velocities and end-effector velocities
  - Relationship between the task-space force and the joint-space torques





$$\dot{X} = J\dot{\theta}$$

$$\tau = J^T F$$

Jacobian

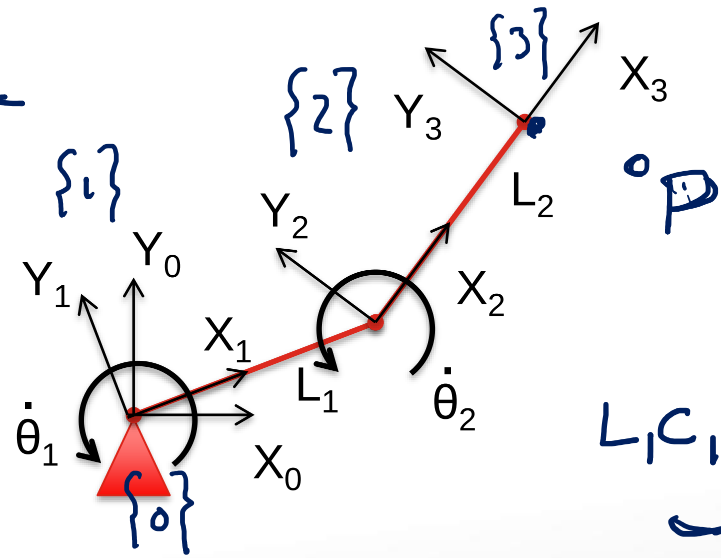
Content

- Introduction - Jacobian
- Method 1 - Direct differentiation (for Linear Jacobian)
- Method 2 - Velocity Propagation from Link to Link
- Method 3 - Explicit Form (For your reading (not included in exam))
- Static Forces in Manipulators
- Singularities

Differentiation to Calculate Velocity

- In our previous lectures, we calculated the tip position for a two-link robot:
- Recall that the tip position is:

$$ {}^{0}\vec{P} = \begin{bmatrix} L_1c_1 + L_2c_{12} \\ L_1s_1 + L_2s_{12} \\ 0 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} $$



Differentiate the position vector wrt time

The velocity is $\dot{x} = \frac{dx}{dt}$

$L_1C_1 + L_2C_{12}$

$\theta_1 \,\&\, \theta_2$

$t \downarrow$

$\frac{d\theta_1}{dt}, \frac{d\theta_2}{dt}$

$$\left.\begin{aligned} \dot{x} &= \frac{dx}{d\theta_1}\frac{d\theta_1}{dt} + \frac{dx}{d\theta_2}\frac{d\theta_2}{dt} \\ \dot{y} &= \frac{dy}{d\theta_1}\frac{d\theta_1}{dt} + \frac{dy}{d\theta_2}\frac{d\theta_2}{dt} \\ \dot{z} &= \ldots \end{aligned}\right\}$$

Chain Rule

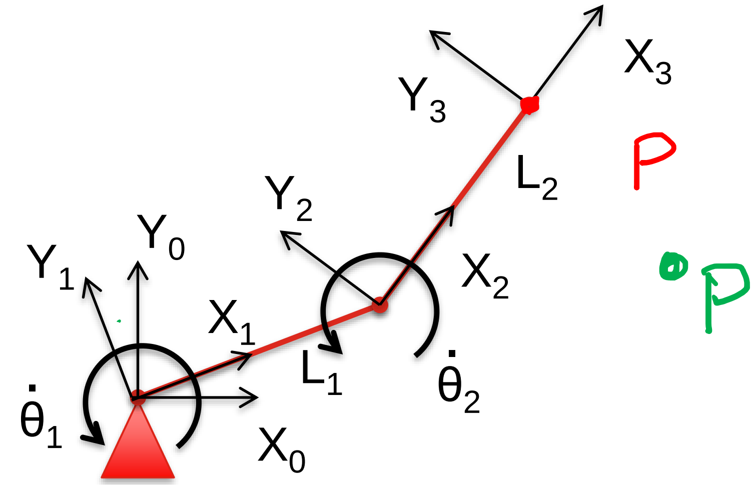
Differentiation to Calculate Velocity

Differentiate the position vector wrt time, using chain rule:

$$\dot{x} = \frac{dx}{d\theta_1}\underbrace{\frac{d\theta_1}{dt}}_{\dot{\theta}_1} + \frac{dx}{d\theta_2}\underbrace{\frac{d\theta_2}{dt}}_{\dot{\theta}_2}$$

$$\dot{y} = \frac{dy}{d\theta_1}\underbrace{\frac{d\theta_1}{dt}}_{\dot{\theta}_1} + \frac{dy}{d\theta_2}\underbrace{\frac{d\theta_2}{dt}}_{\dot{\theta}_2}$$

$$\dot{z} = \ldots$$



Compare with $\dot{X} = J\dot{\Theta}$ → Jacobian

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}_{3\times1} = \begin{bmatrix} dx/d\theta_1 & dx/d\theta_2 \\ dy/d\theta_1 & dy/d\theta_2 \\ dz/d\theta_1 & dz/d\theta_2 \end{bmatrix}_{3\times2} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}_{3\times1}$$

$\{0\}$ ${}^{0}J$ $\underline{V}$ → linear

Differentiation to Calculate Velocity

- Looking back at our two-link robot example:

$$\dot{X} = J\dot{\theta}$$

$$x \rightarrow \quad {}^{0}P = \begin{bmatrix} L_1c_1 + L_2c_{12} \\ L_1s_1 + L_2s_{12} \\ 0 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} dx/d\theta_1 & dx/d\theta_2 \\ dy/d\theta_1 & dy/d\theta_2 \\ dz/d\theta_1 & dz/d\theta_2 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

$$u = \theta_1 + \theta_2 \qquad x = L_1C_1 + L_2Cu$$

- Using the Jacobian method:

$$\frac{dx}{d\theta_1} = \frac{d(L_1C_1)}{d\theta_1} + \overbrace{\frac{d(L_2Cu)}{d\theta_1}}^{\rightarrow \text{Chain Rule} \rightarrow} = -L_1S_1 + \frac{d(L_2Cu)}{du}\frac{du}{d\theta_1}$$

$$= -L_1S_1 - L_2Su = -L_1S_1 - L_2S_{12}$$

$$\frac{dx}{d\theta_2} = 0 + \frac{d(L_2Cu)}{du}\frac{du}{d\theta_2}$$

$$= 0 - L_2S_{12}$$

Another Way to Calculate Velocity

- Differentiate the position vector wrt time, using chain rule:

$$\frac{d}{dt}(P) = \begin{bmatrix} \frac{d}{dt}(L_1 c_1 + L_2 c_{12}) \\ \frac{d}{dt}(L_1 s_1 + L_2 s_{12}) \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{d}{d\theta_1}(L_1 c_1 + L_2 c_{12}) & \frac{d}{d\theta_2}(L_1 c_1 + L_2 c_{12}) \\ \frac{d}{d\theta_1}(L_1 s_1 + L_2 s_{12}) & \frac{d}{d\theta_2}(L_1 s_1 + L_2 s_{12}) \\ & 0 \end{bmatrix} \begin{bmatrix} \frac{d\theta_1}{dt} \\ \frac{d\theta_2}{dt} \end{bmatrix}$$

$$= \underbrace{\begin{bmatrix} (-L_1 s_1 - L_2 s_{12}) & -L_2 s_{12} \\ (L_1 c_1 + L_2 c_{12}) & L_2 c_{12} \\ 0 & 0 \end{bmatrix}}_{J} \times \underbrace{\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}}_{\dot{\theta}}$$

.

Velocity

$${}^{0}\upsilon_3 = \begin{bmatrix} -L_1 s_1 \dot{\theta}_1 - L_2 s_{12}(\dot{\theta}_1 + \dot{\theta}_2) \\ L_1 c_1 \dot{\theta}_1 + L_2 c_{12}(\dot{\theta}_1 + \dot{\theta}_2) \\ 0 \end{bmatrix}$$

Jacobian (notations)

- This result is not surprising, as velocity is the derivatives of position.
- We can perform this derivation in a systematic way.
- First we introduce a generalized joint coordinate.
- We knew:

$$q_i = \begin{cases} \theta_i & \text{revolute} \\ d_i & \text{prismatic} \end{cases}$$

R

P

- Combine both:

$$q_i = \bar{\varepsilon}_i \theta_i + \varepsilon_i d_i$$

  - where

$$\varepsilon_i = \begin{cases} 0 & \text{revolute} \\ 1 & \text{prismatic} \end{cases}$$

R

P

$$\bar{\varepsilon}_i = 1 - \varepsilon_i$$

$$\bar{\varepsilon}_i = \begin{cases} 1 & \text{revolute} \\ 0 & \text{prismatic} \end{cases}$$

q

Jacobian – Direct Differentiation

- Using this notation, the joint coordinate vector is: $q = [q_1, q_2, \ldots q_n]^T$
- The Cartesian position of the tip of the arm is therefore:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} f_x(q) \\ f_y(q) \\ f_z(q) \end{bmatrix} \quad \text{or} \quad x = f(q)$$

- Differentiation gives the Cartesian velocity of the tip of the arm:

$$\underbrace{\begin{bmatrix} \delta x \\ \delta y \\ \delta z \end{bmatrix}}_{m\times 1} = \underbrace{\begin{bmatrix} \frac{\partial f_x}{\partial q_1} & \cdots & \frac{\partial f_x}{\partial q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_z}{\partial q_1} & \cdots & \frac{\partial f_z}{\partial q_n} \end{bmatrix}}_{m\times n} \underbrace{\begin{bmatrix} \delta q_1 \\ \vdots \\ \delta q_n \end{bmatrix}}_{n\times 1}$$

or

$$\dot{x} = J(q)\dot{q}$$

Jacobian

- Summary: Given $\dot{q}$, we can calculate $\dot{x}$.

Jacobian – Direct Differentiation

- Using this notation (q_i), the joint coordinate vector is: $q = [q_1, q_2, \cdots, q_n]^T$
- The Cartesian position of the tip of the arm is therefore:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} f_x(q) \\ f_y(q) \\ f_z(q) \end{bmatrix} \quad \text{or} \quad x = f(q)$$

- Differentiation gives the Cartesian velocity of the tip of the arm:

$J \rightarrow$

$$\underbrace{\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}}_{m\times 1} = \underbrace{\begin{bmatrix} \dfrac{\partial f_x}{\partial q_1} & \cdots & \dfrac{\partial f_x}{\partial q_n} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial f_z}{\partial q_1} & \cdots & \dfrac{\partial f_z}{\partial q_n} \end{bmatrix}}_{m\times n} \underbrace{\begin{bmatrix} \dot{q}_1 \\ \vdots \\ \dot{q}_n \end{bmatrix}}_{n\times 1} \quad \text{or} \quad \dot{x} = J(q)\dot{q}$$

Jacobian

- Summary: Given $\dot{q}$, we can calculate $\dot{x}$.

Jacobian – Notes:

- To indicate the frame in which the Cartesian velocity is expressed, we use leading superscripts:

$$^{0}v = {}^{0}J(q)\dot{q}$$

^{0}v

  - This superscript may be ignored if the frame is obvious.

- Also, remember that the Jacobian is linear but time varying.
  - This means, the joint rates are related to the velocity of the tip in a linear manner, but this relationship is only instantaneous.
  - At the next instant, we have a new Jacobian and new relationship. $J(q)$

- Direct Differentiation is suitable for linear velocity, but not for rotational velocity.

Changing Reference Frame

- Given a Jacobian written in frame {B}:

$$\begin{bmatrix} {}^{B}\upsilon \\ {}^{B}\omega \end{bmatrix} = {}^{B}v = {}^{B}J(q)\dot{q}$$

- We can transform it to frame {A}:

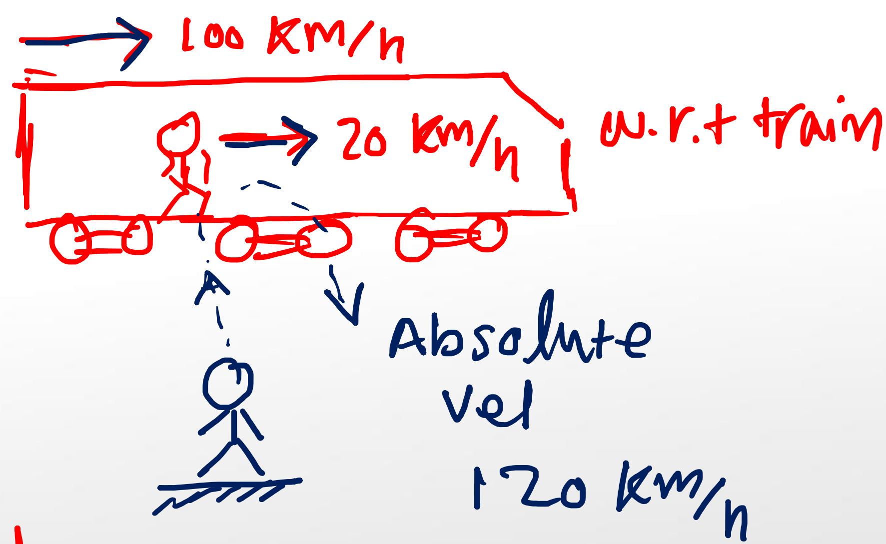
$$\begin{aligned} \begin{bmatrix} {}^{A}\upsilon \\ {}^{A}\omega \end{bmatrix} &= \begin{bmatrix} {}^{A}_{B}R & 0 \\ 0 & {}^{A}_{B}R \end{bmatrix} \begin{bmatrix} {}^{B}\upsilon \\ {}^{B}\omega \end{bmatrix} \\ &= \begin{bmatrix} {}^{A}_{B}R & 0 \\ 0 & {}^{A}_{B}R \end{bmatrix} \cdot {}^{B}J(q)\dot{q} \\ &= {}^{A}J(q)\dot{q} \end{aligned}$$

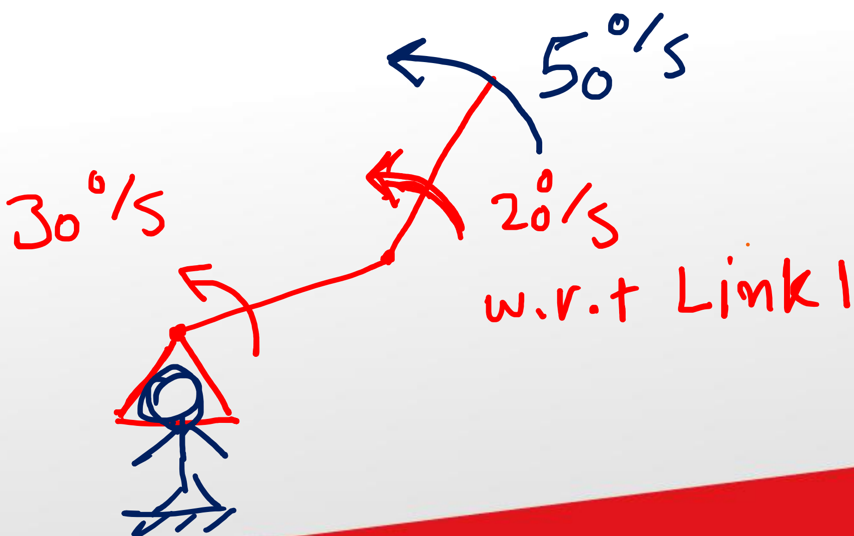
- Therefore:

$$\Rightarrow \quad {}^{A}J(q) = \begin{bmatrix} {}^{A}_{B}R & 0 \\ 0 & {}^{A}_{B}R \end{bmatrix} \cdot {}^{B}J(q)$$

Content

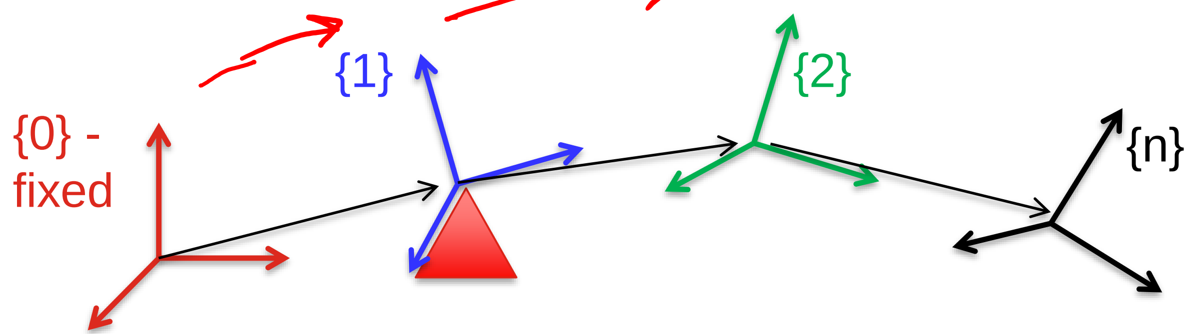
- Introduction - Jacobian
- Method 1 - Direct differentiation (for Linear Jacobian)
- Method 2 - Velocity Propagation from Link to Link → { linear vel, Rotational vel }
- Method 3 – Explicit Form (for Rotational Jacobian)
- Singularities
- Static Forces in Manipulators
- Resolved Motion Rate Control





Velocity Propagation

- For robots, we always use frame {0} as the reference frame.
- The robotic manipulator is a chain of bodies, each capable of moving relative to its neighbour.



- Therefore, the velocity of each link can be computed in order, starting from the base.

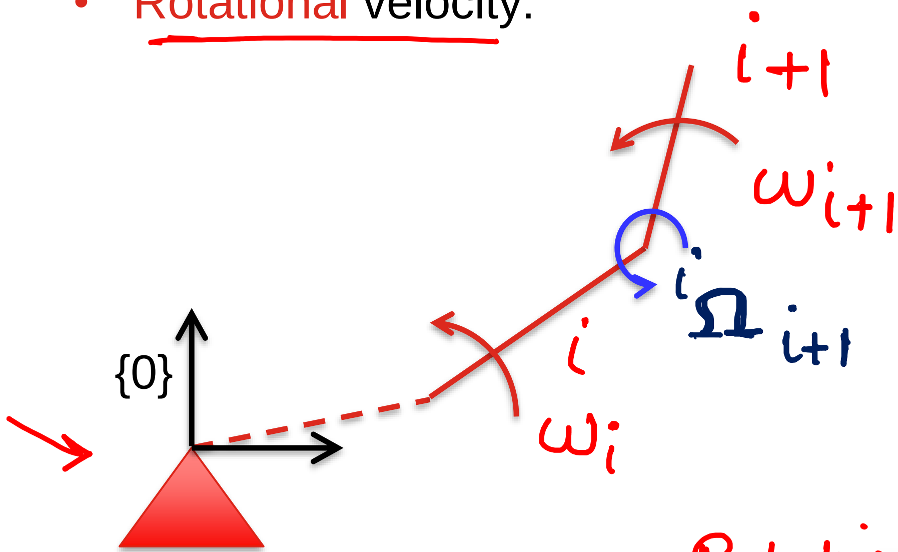
Velocity

- Velocity of link i+1 = Velocity of link i + new components due to joint i+1

- Note: "Velocity of a link": v_i is the linear velocity of the origin of link frame and ω_i is the rotational velocity of the link.

Notations

- We need to differentiate between relative velocity and absolute velocity.
- Rotational velocity:



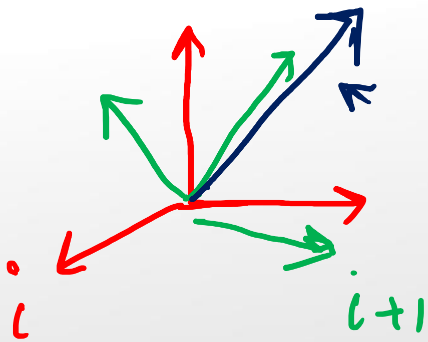
- ω_i, ω_{i+1}: "Absolute" velocity wrt. frame {0}.
- Ω_{i+1}: Angular speed of joint i+I wrt joint i, "relative".

How to get Ω & ω

Rotation from {i} to {i+1} = ${}^{i}_{i+1}R$

Angular vel from {i+1} to {i} = ${}^{i}\Omega_{i+1}$

3x1 vector



Magnitude = Speed

direction = instantaneous axis of Rot.

Notations

- How to get ω and Ω?
- We knew: Relative rotation from frame {i} to {i+1}:

$${}^{i}_{i+1}R$$

- Relative angular velocity of {i+1} wrt {i} is written as:

$${}^{i}\Omega_{i+1}$$

  - This is a 3x1 vector.
  - Direction = instantaneous axis of rotation
  - Magnitude = speed

Notations

$${}^{i}\Omega_{i+1} \xrightarrow{{}^{A}_{i}R} {}^{A}\Omega_{i+1} = {}^{A}_{i}R\,{}^{i}\Omega_{i+1}$$

- We can change the reference frame of expression using the rotation matrix.
- E.g. Relative angular velocity of {i+1} wrt {i}, expressed in frame {i+1} is:

$${}^{i+1}\Omega_{i+1} = {}^{i+1}_{i}R \cdot {}^{i}\Omega_{i+1}$$

- If we now change the reference frame of expression to frame {0}, we then get the absolute velocity. ω_{i+1} ?

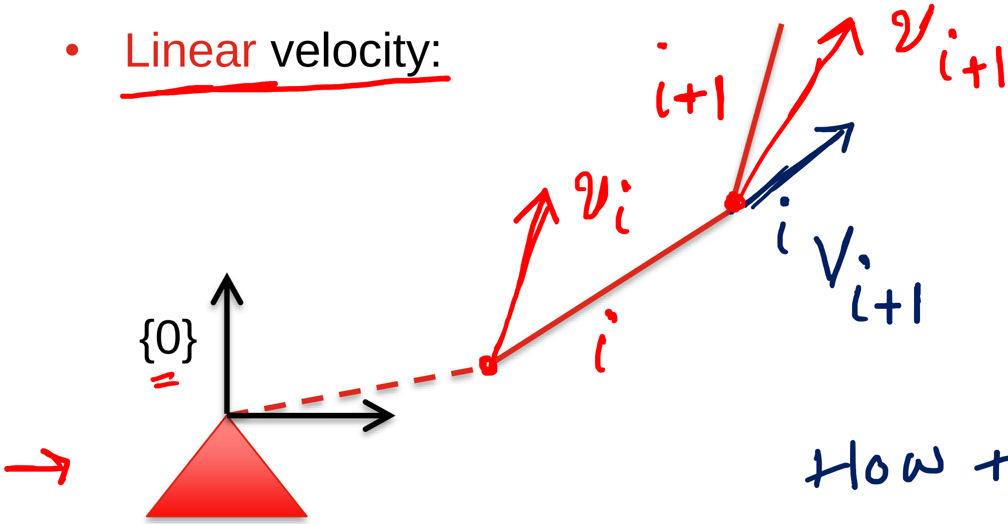
$${}^{0}\Omega_{i+1} \triangleq \omega_{i+1} = {}^{0}_{i}R\,{}^{i}\Omega_{i+1}$$

- Even the absolute velocity can have a change of reference frame, e.g.

$${}^{i+1}\omega_{i+1} = {}^{i+1}_{0}R\,\omega_{i+1} = {}^{i+1}_{0}R\,{}^{0}\Omega_{i+1}$$

Notations

- Linear velocity:

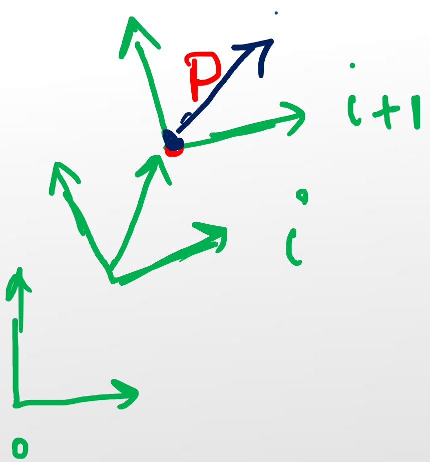


v

- v_i, v_{i+1}: "Absolute" velocity wrt. frame {0}.
- V_{i+1}: Linear speed of joint i+I wrt joint i, "relative".

V

How to get v & V



Position of frame {i+1} w.r.t {i} $^{i}P_{(i+1)ORG}$

vel of frame {i+1} w.r.t {i} $^{i}V_{(i+1)ORG}$

Notations

$${}^{i}V_{i+1} \xrightarrow{{}^{A}_{i}R} {}^{A}V_{i+1} = {}^{A}_{i}R \cdot {}^{i}V_{i+1}$$

- We can change the reference frame of expression using the rotation matrix.
- E.g. Relative linear velocity of {i+1} wrt {i}, expressed in frame {i+1} is:

$${}^{i+1}V_{i+1} = {}^{i+1}_{i}R \cdot {}^{i}V_{i+1}$$

- If we now change the reference frame of expression to frame {0}, we then get the absolute velocity. v_{i+1}

$${}^{0}V_{i+1} = {}^{0}_{i}R \cdot {}^{i}V_{i+1} \triangleq v_{i+1}$$

- Even the absolute velocity can have a change of reference frame, e.g.

$${}^{i+1}v_{i+1} = {}^{i+1}_{0}R\, v_{i+1} = {}^{i+1}_{0}R\, {}^{0}V_{i+1}$$

Notations

- We can change the reference frame of expression using the rotation matrix.
- E.g. Relative linear velocity of {i+1} wrt {i}, expressed in frame {i+1} is:

$$^{i+1}V_{i+1} = {}^{i+1}_{i}R \; {}^{i}V_{i+1}$$

- If we now change the reference frame of expression to frame {0}, we then get the absolute velocity.

$$^{0}V_{i+1} = {}^{0}_{i}R \; {}^{i}V_{i+1} \triangleq v_{i+1}$$

- Even the absolute velocity can have a change of reference frame, e.g.

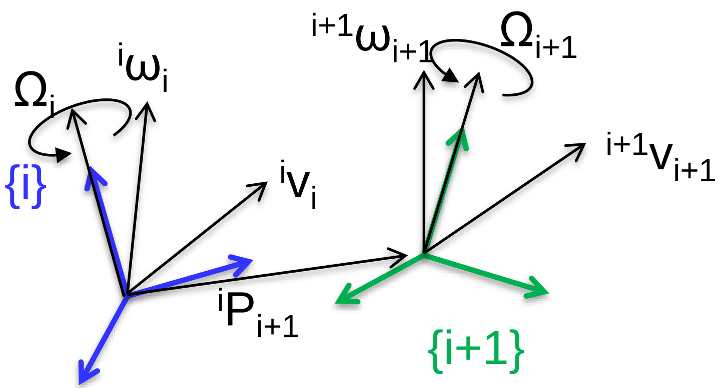
$$^{i+1}v_{i+1} = {}^{i+1}\left({}^{0}V_{i+1} \right) = {}^{i+1}_{0}R \; {}^{0}V_{i+1} \triangleq {}^{i+1}_{0}R \cdot v_{i+1}$$

Sumarry

	Relative	Absolute
Rotational Velocity	${}^{i}\Omega_{i+1}$	${}^{0}\Omega_{i+1} \triangleq \omega_{i+1}$
Linear Velocity	${}^{i}V_{i+1} = {}^{i}V_{(i+1)ORG}$	${}^{0}V_{i+1} \triangleq v_{i+1}$

Velocity Propagation

- Consider links i and i+1, together with their velocities.

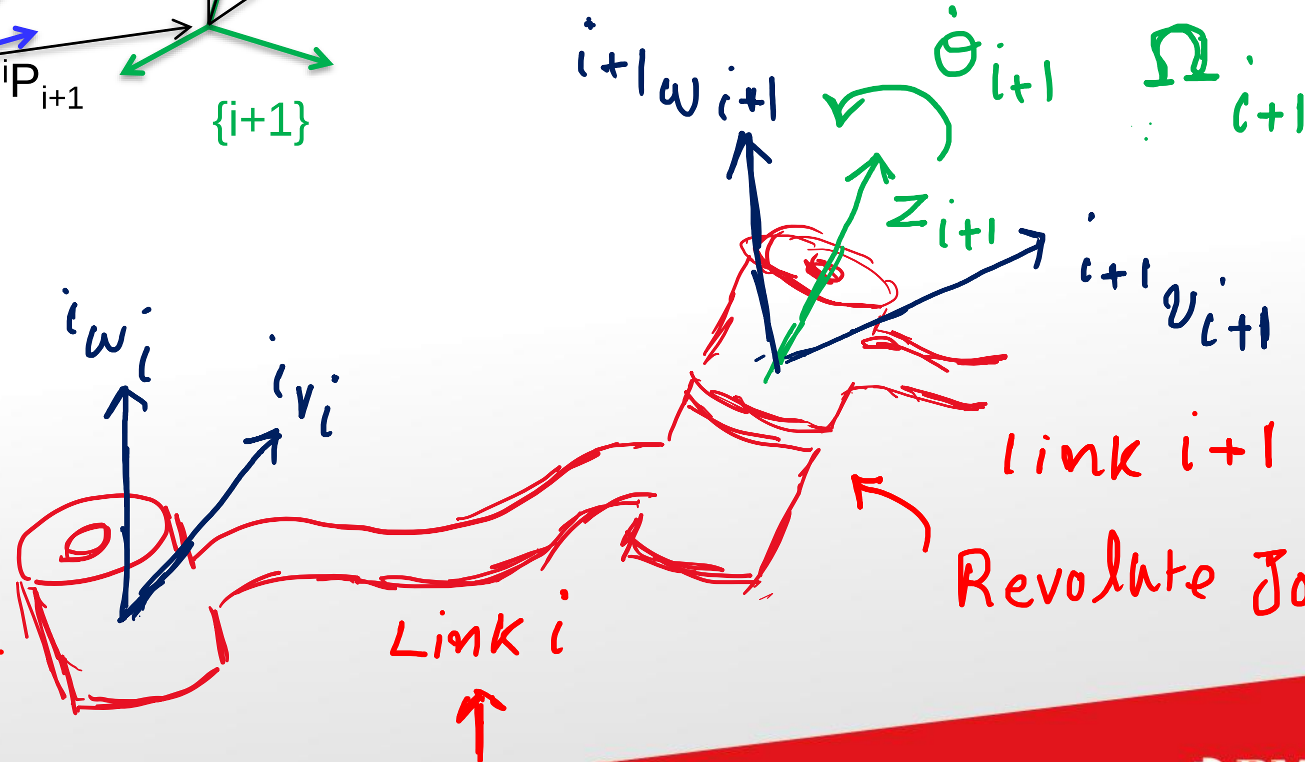


Velocity of link i+1 =

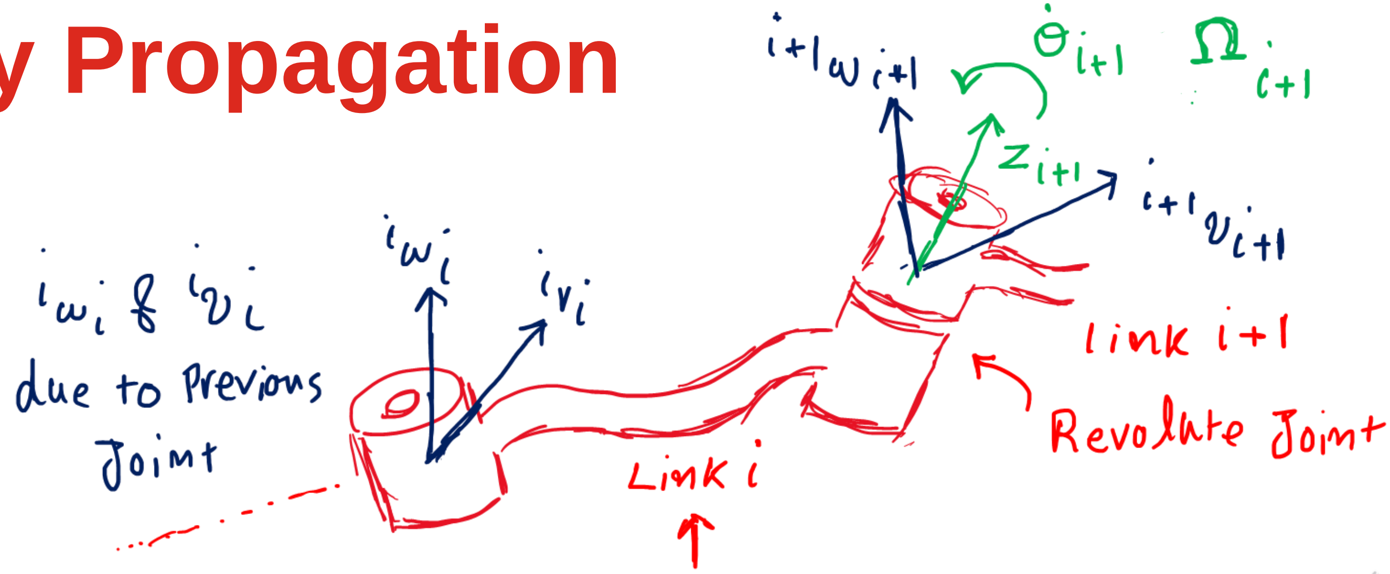
Velocity of link i +

new components due to joint i+1

$\dot{{}^i\omega_i}$ & $\dot{{}^iv_i}$ due to Previous Joint



Velocity Propagation



- Rotational velocities can be added, if both ω vectors are expressed in the same frame. Therefore:

$${}^{i}\omega_{i+1} = {}^{i}\omega_i + {}^{i}\Omega_{i+1}$$

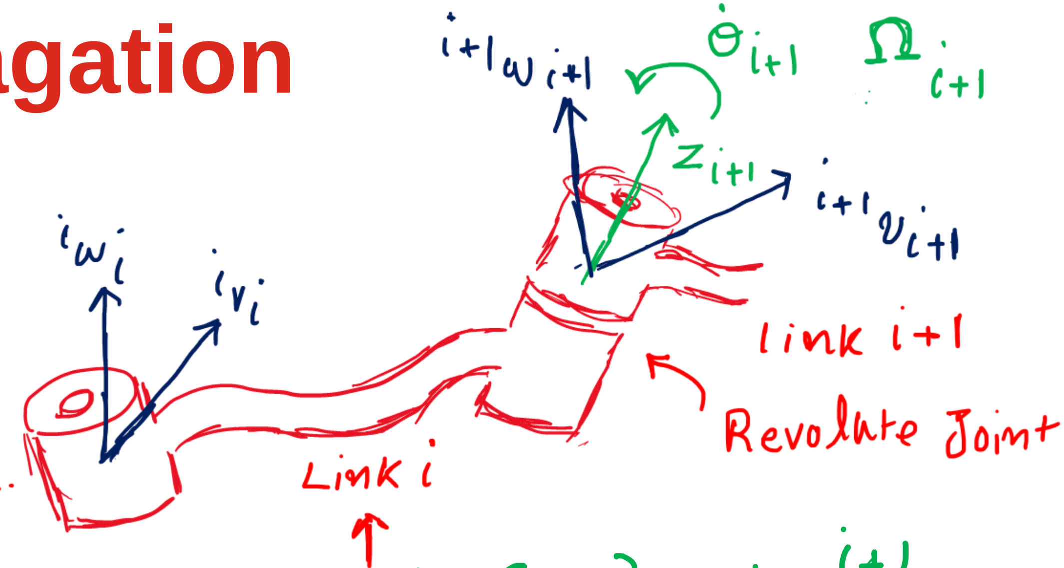
However, writing in frame {i+1} is better

- Pre-multiply every term with ${}^{i+1}_{i}R$ gives:

$${}^{i+1}\omega_{i+1} = {}^{i+1}_{i}R \cdot {}^{i}\omega_i + {}^{i+1}\Omega_{i+1}$$

Velocity Propagation

${}^{i}\omega_{i}$ & ${}^{i}v_{i}$
due to Previous
Joint



$${}^{i+1}\omega_{i+1} = {}^{i+1}_{\ \ i}R \cdot {}^{i}\omega_{i} + \underline{{}^{i+1}\Omega_{i+1}}$$

in $\{i+1\}$, the ${}^{i+1}\Omega_{i+1}$ is always in $\underline{Z\text{-axis}}$

in other words

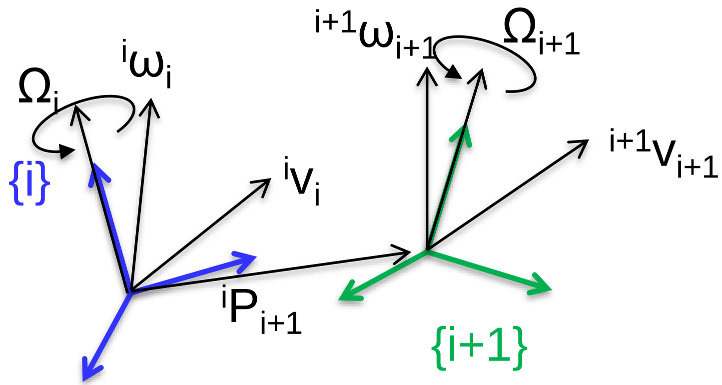
$${}^{i+1}\Omega_{i+1} = \dot\theta_{i+1} \cdot {}^{i+1}\hat{Z}_{i+1}$$

$$= \dot\theta_{i+1} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$${}^{i+1}\omega_{i+1} = {}^{i+1}_{\ \ i}R \cdot {}^{i}\omega_{i} + \dot\theta_{i+1}\, {}^{i+1}\hat{Z}_{i+1}$$

Rot. vel (Revolute Joint)

Velocity Propagation (Rot. Vel)



- Rotational velocities can be added, if both ω vectors are expressed in the same frame. Therefore:

$$\begin{aligned} {}^{i}\omega_{i+1} &= {}^{i}\omega_{i} + {}^{i}_{i+1}R \cdot {}^{i+1}\Omega_{i+1} \\ &= {}^{i}\omega_{i} + {}^{i}_{i+1}R \cdot \dot{\theta}_{i+1} \cdot {}^{i+1}\hat{Z}_{i+1} \end{aligned}$$

where

$$\dot{\theta}_{i+1} \cdot {}^{i+1}\hat{Z}_{i+1} = \begin{bmatrix} 0 \\ 0 \\ {}^{i+1}\dot{\theta}_{i+1} \end{bmatrix}$$

scalar value, thus no frame

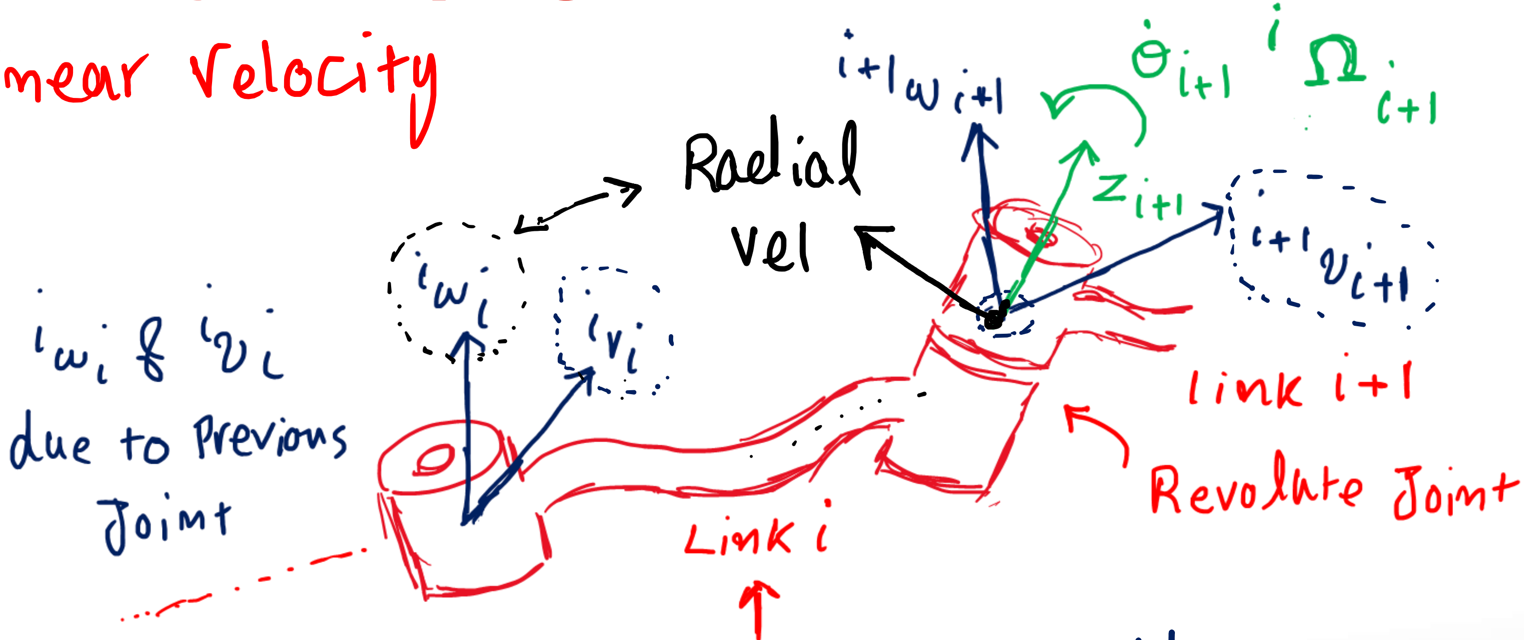
vector

- Pre-multiply every term with $^{i+1}{}_{i}R$ gives:

$${}^{i+1}\omega_{i+1} = {}^{i+1}_{i}R \cdot {}^{i}\omega_{i} + \dot{\theta}_{i+1} \cdot {}^{i+1}\hat{Z}_{i+1}$$

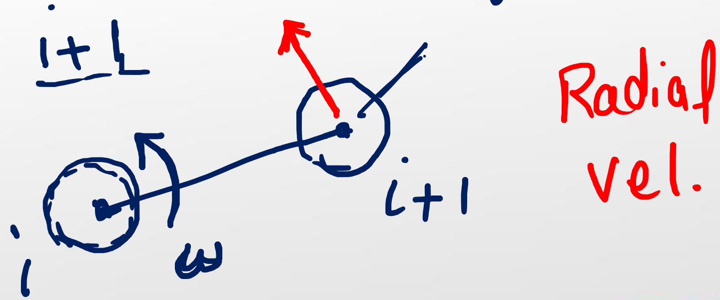
Velocity Propagation

Linear Velocity



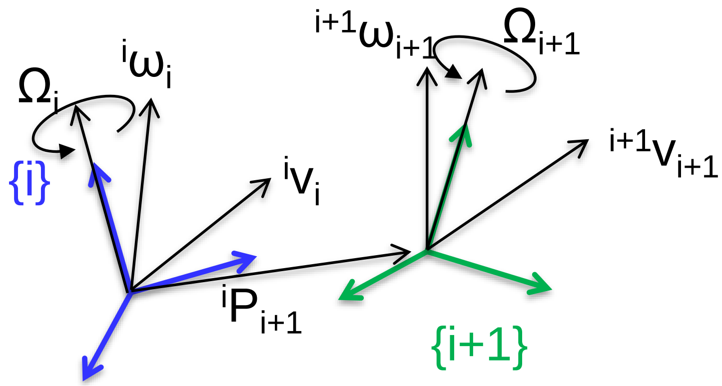
The rotation of joint i+1 wouldn't add any new linear velocity to the joint i+1

However, the rotation of joint i creates a "Radial" velocity for joint i+1



Velocity Propagation (Linear Vel.)

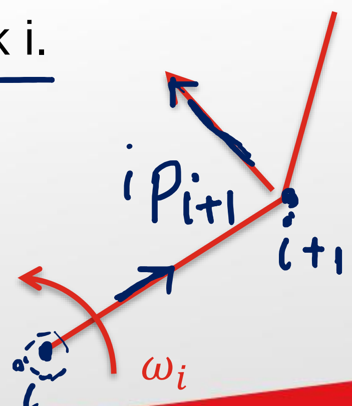
- As for linear velocities, if the joint is revolute, then there is no linear velocity between the two frames.



- The linear velocity of the origin of {i+1} = linear velocity of origin of {i} + new component caused by rotational velocity of link i.

$$^{i}v_{i+1} = {}^{i}v_i + {}^{i}\omega_i \times {}^{i}P_{i+1}$$

Radial linear vel caused by rotation

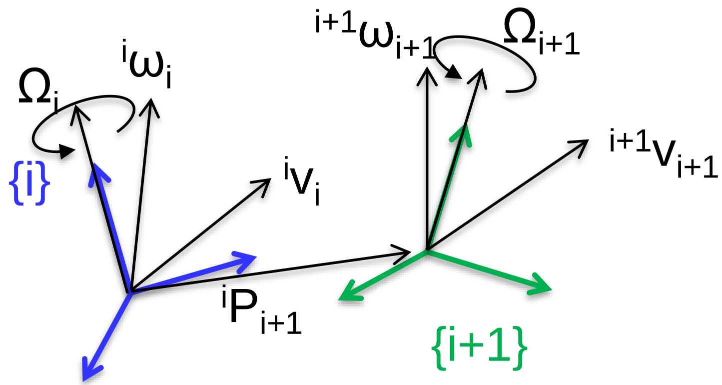


- Pre-multiply every term with $^{i+1}_{i}R$ gives:

$$^{i+1}v_{i+1} = {}^{i+1}_{i}R\left[{}^{i}v_i + {}^{i}\omega_i \times {}^{i}P_{i+1}\right]$$

Velocity Propagation (Linear Vel.)

- As for linear velocities, if the joint is revolute, then there is no linear velocity between the two frames.



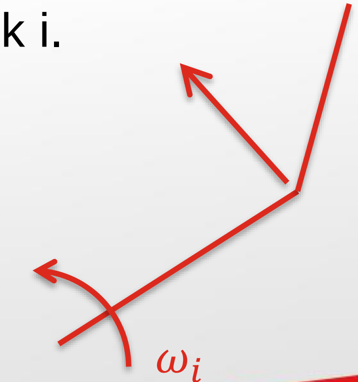
- The linear velocity of the origin of {i+1} = linear velocity of origin of {i}

  + new component caused by rotational velocity of link i.

$$ {}^{i}\upsilon_{i+1} = {}^{i}\upsilon_i + \underbrace{{}^{i}\omega_i \times {}^{i}P_{i+1}} $$

- Pre-multiply every term with ${}^{i+1}_{i}R$ gives:

$$ {}^{i+1}\upsilon_{i+1} = {}^{i+1}_{i}R \cdot \left({}^{i}\upsilon_i + {}^{i}\omega_i \times {}^{i}P_{i+1} \right) $$

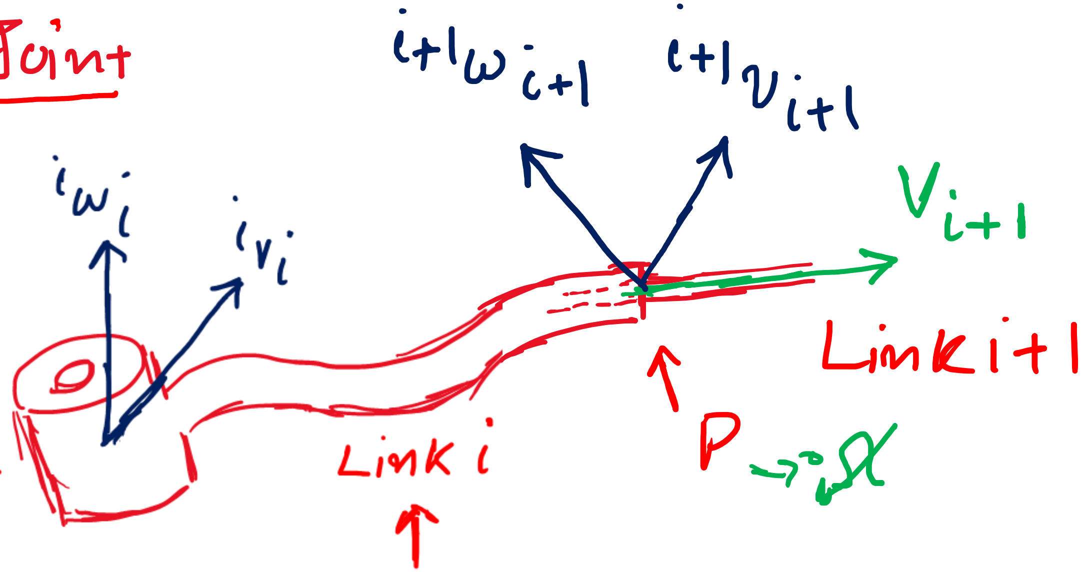


Radial linear vel caused by rotation

Velocity Propagation

→ Prismatic Joint

${}^{i}\omega_i$ & ${}^{i}v_i$
due to Previous
Joint



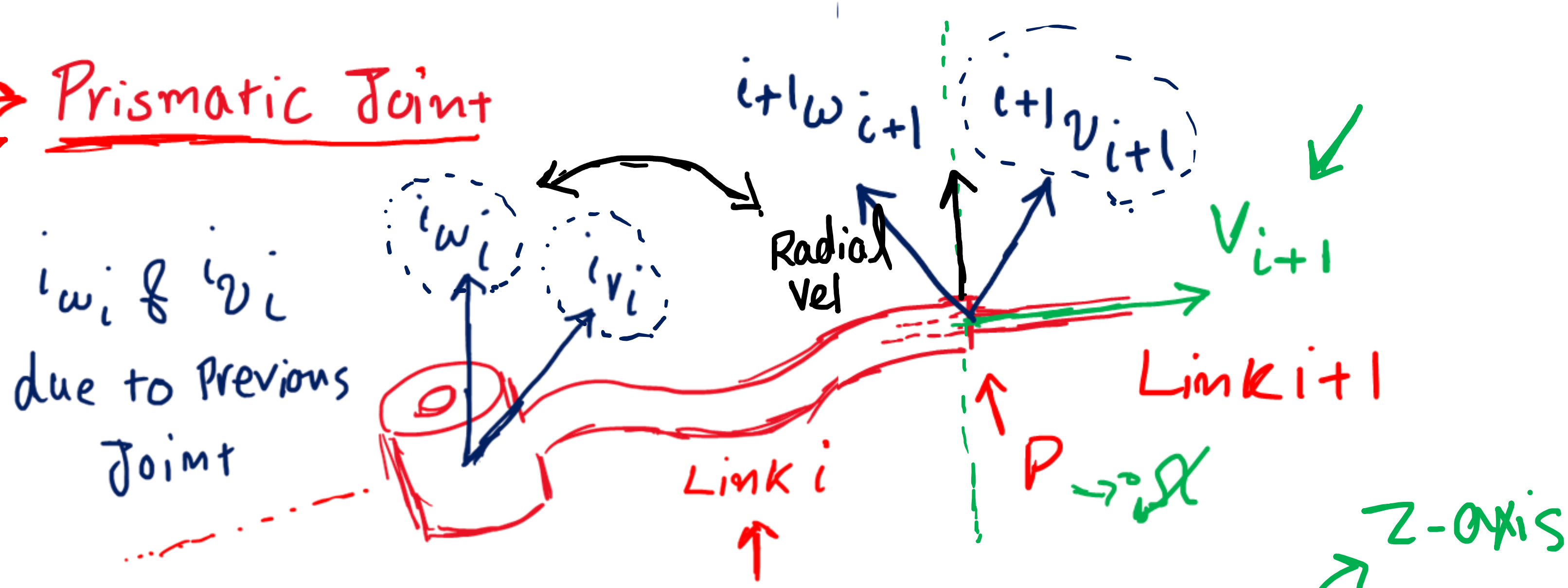
Rot. velocity : Prismatic Joint doesn't Rotate

thus ${}^{i}\omega_{i+1} = {}^{i}\omega_i$ ${}^{i+1}_{i}R$

Express in {i+1} ${}^{i+1}\omega_{i+1} = {}^{i+1}_{i}R \cdot {}^{i}\omega_i$

Velocity Propagation

→ Prismatic Joint



Linear Velocity:

$$^{i+1}\dot v_{i+1} = \underbrace{{}^{i+1}_{\;\;i}R\left({}^{i}\dot v_i + {}^{i}\dot\omega_i \times {}^{i}P_{i+1}\right)}_{\text{linear Velocities caused before the P. Joint}} + \overbrace{\dot d_{i+1} \cdot {}^{i+1}\hat Z_{i+1}}^{{}^{i+1}\dot V_{i+1}}$$

Velocity Propagation - Algorithm

1 • Start with $^{0}\omega_0, {}^{0}\upsilon_0$ $\quad$ $^{0}\omega_0, {}^{0}v_0$

2 • Calculate recursively from link to link:

- For revolute joints: $$^{i+1}\omega_{i+1} = {}^{i+1}_{i}R \cdot {}^{i}\omega_i + \dot{\theta}_{i+1} \cdot {}^{i+1}\hat{Z}_{i+1}$$ Rot.

R

$$^{i+1}\upsilon_{i+1} = {}^{i+1}_{i}R \cdot \left({}^{i}\upsilon_i + {}^{i}\omega_i \times {}^{i}P_{i+1}\right)$$ Linear

- For prismatic joints: $$^{i+1}\omega_{i+1} = {}^{i+1}_{i}R \cdot {}^{i}\omega_i$$ ← Rot.

P

$$^{i+1}\upsilon_{i+1} = {}^{i+1}_{i}R \cdot \left({}^{i}\upsilon_i + {}^{i}\omega_i \times {}^{i}P_{i+1}\right) + \dot{d}_{i+1} \cdot {}^{i+1}\hat{Z}_{i+1}$$ Linear

untill Join n

3 • Finally, we will arrive at $^{n}\omega_n, {}^{n}\upsilon_n$. $\quad$ $^{n}\omega_n, {}^{n}v_n$

4 • Transform these back to base frame by: $$\begin{bmatrix} {}^{0}\upsilon_n \\ {}^{0}\omega_n \end{bmatrix} = \begin{bmatrix} {}^{0}_{n}R & 0 \\ 0 & {}^{0}_{n}R \end{bmatrix} \begin{bmatrix} {}^{n}\upsilon_n \\ {}^{n}\omega_n \end{bmatrix}$$

{0}

⇒ Velocity Propagation - Algorithm

1
- Start with $^{0}\omega_0, ^{0}\upsilon_0$ $^{0}\omega_0, ^{0}v_0$

2
- Calculate recursively from link to link:
  - For revolute joints: $^{i+1}\omega_{i+1} = ^{i+1}_{i}R \cdot ^{i}\omega_i + \dot{\theta}_{i+1} \cdot ^{i+1}\hat{Z}_{i+1}$ Eq. 1

    R

    $^{i+1}\upsilon_{i+1} = ^{i+1}_{i}R \cdot \left(^{i}\upsilon_i + ^{i}\omega_i \times ^{i}P_{i+1} \right)$ Eq. 2
  - For prismatic joints: $^{i+1}\omega_{i+1} = ^{i+1}_{i}R \cdot ^{i}\omega_i$ Eq. 3

    P

    $^{i+1}\upsilon_{i+1} = ^{i+1}_{i}R \cdot \left(^{i}\upsilon_i + ^{i}\omega_i \times ^{i}P_{i+1} \right) + \dot{d}_{i+1} \cdot ^{i+1}\hat{Z}_{i+1}$ Eq. 4

Untill joint n

3
- Finally, we will arrive at $^{n}\omega_n, ^{n}\upsilon_n$. $^{n}\omega_n, ^{n}v_n$

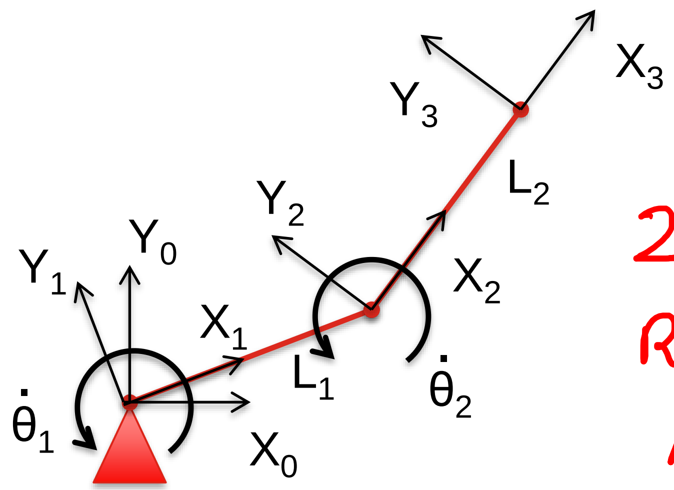
4
- Transform these back to base frame by: $\{0\}$

$$\begin{bmatrix} ^{0}\upsilon_n \\ ^{0}\omega_n \end{bmatrix} = \begin{bmatrix} ^{0}_{n}R & 0 \\ 0 & ^{0}_{n}R \end{bmatrix} \begin{bmatrix} ^{n}\upsilon_n \\ ^{n}\omega_n \end{bmatrix}$$

3

Example

- Given a two link robot as shown.
- What is the velocity at its tip?



2-Link Robotic Arm

- Because the formulae needs the transformation matrices, they are computed first.

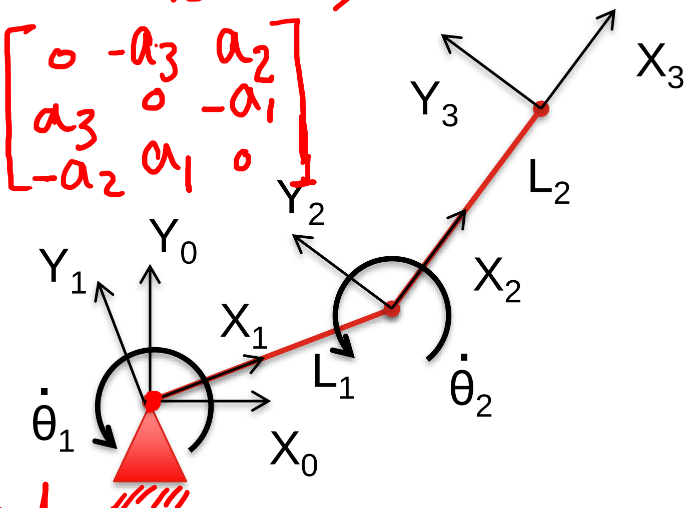
${}^0_1R$ ${}^1_2R$ ${}^2_3R$

$$ {}^0_1T = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^1_2T = \begin{bmatrix} c_2 & -s_2 & 0 & L_1 \\ s_2 & c_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^2_3T = \begin{bmatrix} 1 & 0 & 0 & L_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Example

$a \times b = [a]_\times b$

$[a]_\times = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \rightarrow$ skew-symmetric



- Next, we apply the iterative formulae:

1 • Start with: $\{0\}$

$$^0\omega_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \qquad ^0\upsilon_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

2 • Then: Joint1, R, Eq.1

Fixed

$^1v_1$: Joint1, R, Eq.2

$$^1\omega_1 = {}^1_0R \cdot {}^0\omega_0 + \dot\theta_1 \cdot {}^1\hat Z_1$$

$$= {}^0_1R^T \cdot {}^0\omega_0 + \dot\theta_1 \cdot {}^1\hat Z_1$$

$$= \begin{bmatrix} \cdot \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dot\theta_1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 \\ 0 \\ \dot\theta_1 \end{bmatrix}$$

$$^1\upsilon_1 = {}^1_0R \cdot \left({}^0\upsilon_0 + {}^0\omega_0 \times {}^0P_1\right) = {}^0_1R^T \cdot \left({}^0\upsilon_0 + {}^0\omega_0 \times {}^0P_1\right)$$

$$= \begin{bmatrix} {}^1_0R \end{bmatrix} \cdot \left(\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & -{}^0\omega_{0z} & {}^0\omega_{0y} \\ {}^0\omega_{0z} & 0 & -{}^0\omega_{0x} \\ -{}^0\omega_{0y} & {}^0\omega_{0x} & 0 \end{bmatrix}}_{[{}^0\omega_0]_\times} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \right)$$

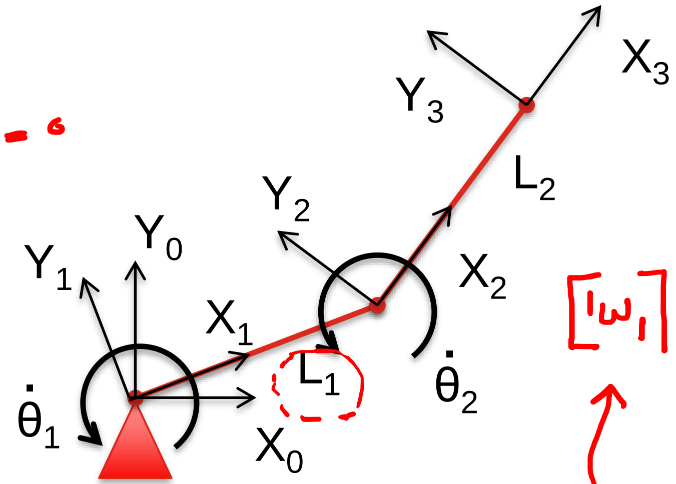
$$= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = 0$$

Example

- Next we have:

$${}^1\omega_1 = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 \end{bmatrix} \qquad {}^1v_1 = 0$$

${}^1\omega_{1z}$



$[{}^1\omega_1]_\times \, {}^1P_2$

Step2 : Link 2

set i=1

${}^2\omega_2$: Joint 2, R, Eq.1

$${}^2\omega_2 = {}^2_1R \cdot {}^1\omega_1 + \dot{\theta}_2 \cdot {}^2\hat{Z}_2$$

$$= {}^1_2R^T \cdot {}^1\omega_1 + \dot{\theta}_2 \cdot {}^2\hat{Z}_2$$

$$= \begin{bmatrix} C_2 & S_2 & 0 \\ -S_2 & C_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_2 \end{bmatrix}$$

$$= \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 + \dot{\theta}_2 \end{bmatrix}$$

Joint 2, R, Eq.2

$${}^2v_2 = {}^2_1R \cdot \left({}^1v_1 + {}^1\omega_1 \times {}^1P_2\right) = {}^1_2R^T \cdot \left(\cancel{{}^1v_1}^{\,0} + \underline{{}^1\omega_1 \times {}^1P_2}\right)$$

$$= \begin{bmatrix} C_2 & S_2 & 0 \\ -S_2 & C_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \left(\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & -{}^1\omega_{1z} & {}^1\omega_{1y} \\ {}^1\omega_{1z} & 0 & -{}^1\omega_{1x} \\ -{}^1\omega_{1y} & {}^1\omega_{1x} & 0 \end{bmatrix} \begin{bmatrix} L_1 \\ 0 \\ 0 \end{bmatrix} \right)$$

$$= \begin{bmatrix} \downarrow \\ // \\ \end{bmatrix} \begin{bmatrix} 0 \\ L_1 \, {}^1\omega_{1z} \\ \cancel{-L_1 \, {}^1\omega_{1y}}^{\,0} \end{bmatrix} = \begin{bmatrix} L_1 \, {}^1\omega_{1z} S_2 \\ L_1 \, {}^1\omega_{1z} C_2 \\ 0 \end{bmatrix} = \begin{bmatrix} L_1 S_2 \dot{\theta}_1 \\ L_1 C_2 \dot{\theta}_1 \\ 0 \end{bmatrix}$$

Example

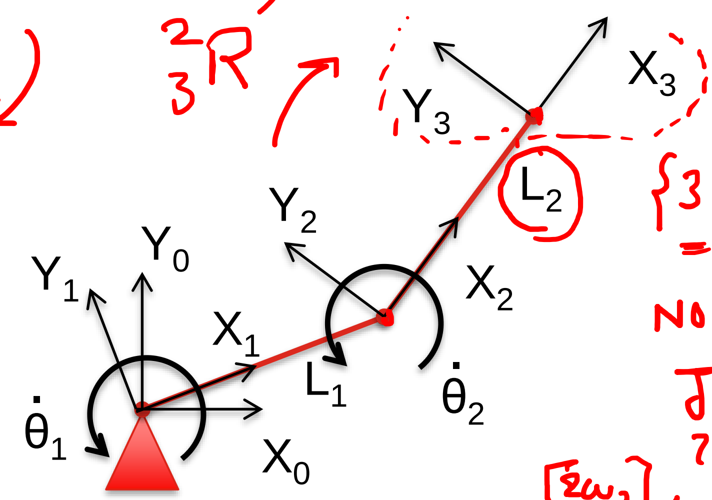
2-Link Robot

${}^2\omega_2$ & ${}^2\upsilon_2$

Previous step

Step 3

- If the end-effector is of interest, we can propagate the velocity from the last link to the end-effector as well:



${}^2_3R \rightarrow I$

$\{3\}$

No R. or P. Joint

Eq. 1

$$
\begin{aligned}
{}^3\omega_3 &= {}^3_2R \cdot {}^2\omega_2 + \dot{\theta}_3 \cdot {}^3\hat{Z}_3 \\
&= {}^2_3R^T \cdot {}^2\omega_2 + \dot{\theta}_3 \cdot {}^3\hat{Z}_3 \\
&= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 + \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 + \dot{\theta}_2 \end{bmatrix} \\
&= {}^2\omega_2
\end{aligned}
$$

Eq. 2

$[{}^2\omega_2]_\times \, {}^2P_3$

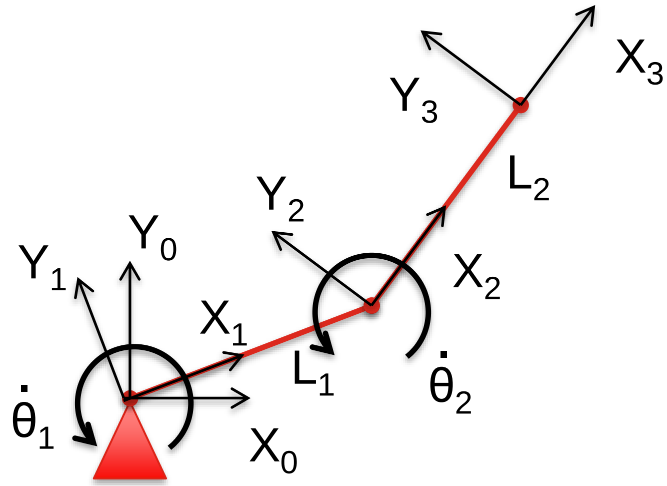
$$
\begin{aligned}
{}^3\upsilon_3 &= {}^3_2R \cdot \left({}^2\upsilon_2 + {}^2\omega_2 \times {}^2P_3\right) = {}^2_3R^T \cdot \left({}^2\upsilon_2 + {}^2\omega_2 \times {}^2P_3\right) \\
&= \begin{bmatrix} & & \\ & I & \\ & & \end{bmatrix} \cdot \left(\begin{bmatrix} L_1 S_2 \dot{\theta}_1 \\ L_1 C_2 \dot{\theta}_1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & -{}^2\omega_{2z} & {}^2\omega_{2y} \\ {}^2\omega_{2z} & 0 & -{}^2\omega_{2x} \\ -{}^2\omega_{2y} & {}^2\omega_{2x} & 0 \end{bmatrix} \begin{bmatrix} L_2 \\ 0 \\ 0 \end{bmatrix} \right) \\
&= \begin{bmatrix} L_1 S_2 \dot{\theta}_1 \\ L_1 C_2 \dot{\theta}_1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ L_2 \, {}^2\omega_{2z} \\ -L_2 \, {}^2\omega_{2y} \end{bmatrix} = \begin{bmatrix} L_1 S_2 \dot{\theta}_1 \\ L_1 C_2 \dot{\theta}_1 + L_2 (\dot{\theta}_1 + \dot{\theta}_2) \\ 0 \end{bmatrix}
\end{aligned}
$$

$\hookrightarrow {}^2\omega_{2z}$

0

Example

Step 4



- The last step is to transform these back to the base frame.
- We need: ${}^3v_3, {}^3\omega_3$

$$\begin{bmatrix} {}^0\upsilon_n \\ {}^0\omega_n \end{bmatrix} = \begin{bmatrix} {}^0_nR & 0 \\ 0 & {}^0_nR \end{bmatrix} \begin{bmatrix} {}^n\upsilon_n \\ {}^n\omega_n \end{bmatrix}$$

- where: ${}^0_3R = {}^0_1R \cdot {}^1_2R \cdot {}^2_3R$

$$= \begin{bmatrix} c_{12} & -s_{12} & 0 \\ s_{12} & c_{12} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$ {}^0\omega_3 = \begin{bmatrix} c_{12} & -s_{12} & 0 \\ s_{12} & c_{12} & 0 \\ 0 & 0 & 1 \end{bmatrix} \overbrace{\begin{bmatrix} 0 \\ 0 \\ \dot\theta_1+\dot\theta_2 \end{bmatrix}}^{{}^3\omega_3} = \begin{bmatrix} 0 \\ 0 \\ \dot\theta_1+\dot\theta_2 \end{bmatrix}$$

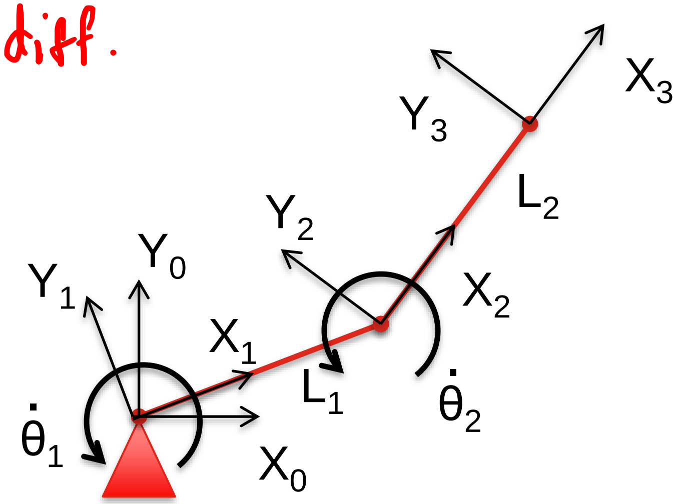
$$ {}^0v_3 = \begin{bmatrix} c_{12} & -s_{12} & 0 \\ s_{12} & c_{12} & 0 \\ 0 & 0 & 1 \end{bmatrix} \overbrace{\begin{bmatrix} L_1 s_2 \dot\theta_1 \\ L_1 c_2 \dot\theta_1 + L_2(\dot\theta_1+\dot\theta_2) \\ 0 \end{bmatrix}}^{{}^3v_3} = \begin{bmatrix} -L_1 s_1 \dot\theta_1 - L_2 s_{12}\dot\theta_1 - L_2 s_{12}\dot\theta_2 \\ L_1 c_1 \dot\theta_1 + L_2 c_{12}\dot\theta_1 + L_2 c_{12}\dot\theta_2 \\ 0 \end{bmatrix}$$

Example - Jacobian

diff.

- What about Jacobian Matrices:

$$\begin{cases} {}^0J_v \\ {}^0J_\omega \end{cases}$$



$$0w_3 = \begin{bmatrix} c_{12} & -s_{12} & 0 \\ s_{12} & c_{12} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 + \dot{\theta}_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 + \dot{\theta}_2 \end{bmatrix}$$

${}^0\omega_z$

$$0v_3 = \begin{bmatrix} c_{12} & -s_{12} & 0 \\ s_{12} & c_{12} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} L_1 s_2 \dot{\theta}_1 \\ L_1 c_2 \dot{\theta}_1 + L_2(\dot{\theta}_1 + \dot{\theta}_2) \\ 0 \end{bmatrix} = \begin{bmatrix} -L_1 s_1 \dot{\theta}_1 - L_2 s_{12}(\dot{\theta}_1 + \dot{\theta}_2) \\ L_1 c_1 \dot{\theta}_1 + L_2 c_{12}(\dot{\theta}_1 + \dot{\theta}_2) \\ 0 \end{bmatrix}$$

${}^0J_v$: the same as direct diff. $\Rightarrow$

$\dot{X} = J\dot{\theta}$ ${}^0J_\omega$

$${}^0J_\omega : \begin{bmatrix} {}^0\omega_x \\ {}^0\omega_y \\ {}^0\omega_z \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

Thank you!

Have a good evening.