CHAPTER 3

Rigid Body Motion and Robot Kinematics of Velocity

Relative Velocities between
Rigid Bodies

Learning Objectives

1. Understand and Relate time derivatives of position and orientation representations with translational and angular velocities.
2. Transform velocities in different spaces
3. Relate joint velocities with end-effector velocities
4. Understand robot singularities (limitations of robot motion)
5. Use velocity relationships to command motion of robots (resolved rate motion control)

Translational Velocities

${}^{A}\mathbf{u}_{B} \in \Re^{3x1}$ = translational velocity of frame B (i.e., origin of frame B) relative of frame A

$$ {}^{A}\mathbf{u}_{B} = \frac{d}{dt}\,{}^{A}\mathbf{p}_{B} = \lim_{\Delta t \to 0} \frac{{}^{A}\mathbf{p}_{B}(t+\Delta t) - {}^{A}\mathbf{p}_{B}(t)}{\Delta t} $$

"frame of differentiation" is A (how fast is B moving with respect to A)

Velocity, like any vector may be expressed in another frame, say W

$$ {}^{W}_{A}\mathrm{u}_{B} = {}^{W}R_{A}\,{}^{A}\mathrm{u}_{B} $$

If missing, A = W

Does not depend on origins of Frames A and W

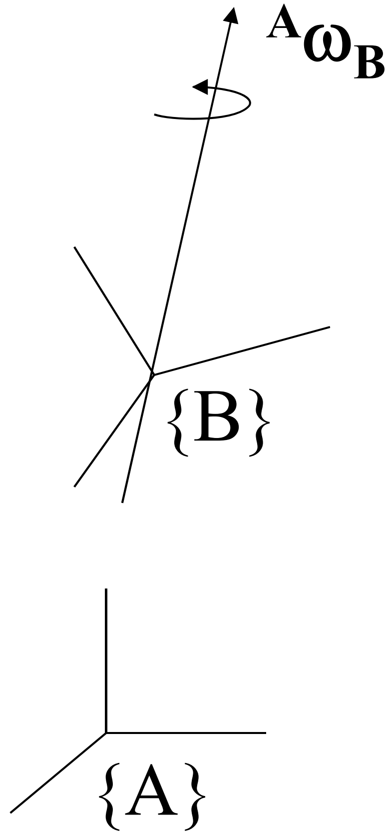
Rotational Velocities

- At a certain instant, frame B has an orientation ${}^{A}\mathbf{R}_{B}$ and its rotational motion may be represented by the rotational (angular) velocity vector

${}^{A}\boldsymbol{\omega}_{B} \in \Re^{3x1}$

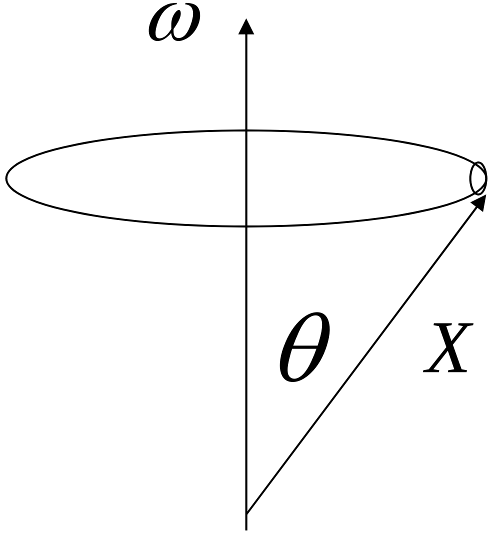
unit vector along ${}^{A}\boldsymbol{\omega}_{B}$ = instantaneous axis of rotation = ${}^{A}\mathbf{k}_{B}$

magnitude of ${}^{A}\boldsymbol{\omega}_{B}$ = speed of rotation



- ${}^{A}\boldsymbol{\omega}_{B}$ is related to $\frac{d}{dt}{}^{A}\mathbf{R}_{B} = {}^{A}\dot{\mathbf{R}}_{B} = \lim_{\Delta t \to 0} \frac{{}^{A}\mathbf{R}_{B}(t+\Delta t) - {}^{A}\mathbf{R}_{B}(t)}{\Delta t}$

Angular velocity ⇔ Time Derivative of Rotation Matrix



X can be one of the Cartesian axes, in a body that is rotating

$$mag = |\omega||x|\sin\theta$$

Direction = along plane normal to both ω and x

X = vector rotating about ω

$$\dot{x} = \omega \times x$$

Time derivative of rotation matrix

$$\dot{x} = \omega \times x = \hat{\omega} x$$

$$\dot{y} = \omega \times y = \hat{\omega} y$$

$$\dot{z} = \omega \times z = \hat{\omega} z$$

$\hat{\omega} =$ cross product operator

$$\begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix}$$

A skew symmetric matrix

3 x 3 matrix version of 3 x 1 vector
Cross product equivalent

Skew Symmetric Matrix as a Cross Product Operator

$$\text{Let } \hat{\omega} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \text{ and } \omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

Then

$\hat{\omega}\,\mathbf{p} = \boldsymbol{\omega} \text{ x } \mathbf{p}$

where $\mathbf{p} \in \Re^{3x1}$ vector

Allows consistent treatment of cross-product as a matrix vector product

Combining the three equations:

$$\dot{x} = \omega \times x = \hat{\omega} x$$

$$\dot{y} = \omega \times y = \hat{\omega} y$$

$$\dot{z} = \omega \times z = \hat{\omega} z$$

Time derivatives of
Unit vectors

$$\begin{pmatrix} \dot{x} & \dot{y} & \dot{z} \end{pmatrix} = \hat{\omega} \begin{pmatrix} x & y & z \end{pmatrix}$$

$$\dot{R} = \hat{\omega} R \quad \Leftrightarrow \quad \dot{R} R^T = \hat{\omega}$$

Time derivative
of rotation
matrix

$$\hat{\omega} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \text{ and } \omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

Angular Velocity Vector

$$\omega = \begin{pmatrix} k_x \\ k_y \\ k_z \end{pmatrix} \dot{\theta}$$

- 3 x 1 vector
  - Magnitude = speed of rotation
  - Direction = axis of rotation

$$= \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} = \begin{pmatrix} k_x \dot{\theta} \\ k_y \dot{\theta} \\ k_z \dot{\theta} \end{pmatrix}$$

Orientation Error

- 3 x 1 representation like position error
- Is there a meaning to "Δx, Δy, Δz"?
- One interpretation:
  - ΔRoll, ΔPitch, ΔYaw
- Another interpreation = Δ Rotation Matrix
  - $\Delta \begin{pmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{pmatrix}$

3 x 3

Can it be reduced to 3 x 1?

Orientation Error

- Angular velocity – measure of instantaneous 3 x 1 orientation error
- Allows the relationship between

$$\underset{3\times 3}{\Delta R} \Leftrightarrow \underset{3\times 1}{\Delta \Phi} = \begin{pmatrix} \Delta\phi_x \\ \Delta\phi_y \\ \Delta\phi_z \end{pmatrix}$$

$$\Delta \Phi = \int \omega$$

$$\Delta R = \int \dot{R}$$

$$\dot{R} = \hat{\omega} R \underset{\wedge}{\rightarrow} \begin{pmatrix} \dot{x} & \dot{y} & \dot{z} \end{pmatrix} = \hat{\omega} \begin{pmatrix} x & y & z \end{pmatrix}$$

$$\underset{3\times 3}{\Delta R R^T} = \underset{3\times 3}{\Delta \hat{\Phi}} = \begin{pmatrix} 0 & -\Delta\phi_z & \Delta\phi_y \\ \Delta\phi_z & 0 & -\Delta\phi_x \\ -\Delta\phi_y & \Delta\phi_x & 0 \end{pmatrix}$$

skew sym

Angular Velocites and Time Derivates of Roll, Pitch, Yaw

$$ {}^{A}\mathbf{R}_{B} = \mathbf{Rot}\,(\,z,\ \phi\,)\ \mathbf{Rot}\,(\,y,\ \theta\,)\ \mathbf{Rot}\,(\,x,\ \varphi) $$

$$ {}^{A}\omega_{B} \quad \Longleftrightarrow \quad \dot{\phi}, \dot{\theta}, \dot{\varphi} $$

$$ = \quad \dot{x}_{RPY} $$

Roll Pitch Yaw Rates & Angular Velocities

$$^{A}\mathbf{R}_{\mathbf{B}} = \mathbf{Rot\ (\ z,\ \phi\ )\ Rot\ (\ y,\ \theta\ )\ Rot\ (\ x,\ \varphi)}$$

$$\omega = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \dot{\phi} + \mathrm{Rot}(z,\phi) \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \dot{\theta} + \mathrm{Rot}(z,\phi)\,\mathrm{Rot}(y,\theta) \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \dot{\varphi}$$

$$\omega = \underbrace{\begin{pmatrix} 0 & -\sin\phi & \cos\phi\ \cos\theta \\ 0 & \cos\phi & \sin\phi\ \cos\theta \\ 1 & 0 & -\sin\theta \end{pmatrix}}_{\mathbf{E_{RPY}}} \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\varphi} \end{pmatrix} \qquad \omega = E_{RPY}\,\dot{x}_{RPY}$$

but $E_{RPY}{}^{-1}$ may not exist

Mathematical singularities

Roll Pitch Yaw Ratio & Angular Velocities

$$\dot{x}_{RPY} = \underbrace{\begin{pmatrix} \dfrac{\cos\phi \ \sin\theta}{\cos\theta} & \dfrac{\sin\phi\sin\theta}{\cos\theta} & 1 \\ -\sin\phi & \cos\phi & 0 \\ \dfrac{\cos\phi}{\cos\theta} & \dfrac{\sin\phi}{\cos\theta} & 0 \end{pmatrix}} \omega$$

If $\cos\theta = 0$, matrix does not exist

$\rightarrow$ Math. Singularity

$\boldsymbol{\omega} \rightarrow \dot{\mathbf{x}}_{\mathbf{r}}$ not always possible

Not all possible angular matrix can be represented

This is a problem with 3 parameter representations for orientation

Rotational Velocities

As with any vector, the rotational velocity vector ${}^{A}\omega_{B}$ may be expressed in another frame C:

$$ {}^{C}\omega_{B} = {}^{C}R_{A}\,{}^{A}\omega_{B} \qquad \text{3 x 3 times 3 x 1 vector} $$

the equivalent matrix product representation is:

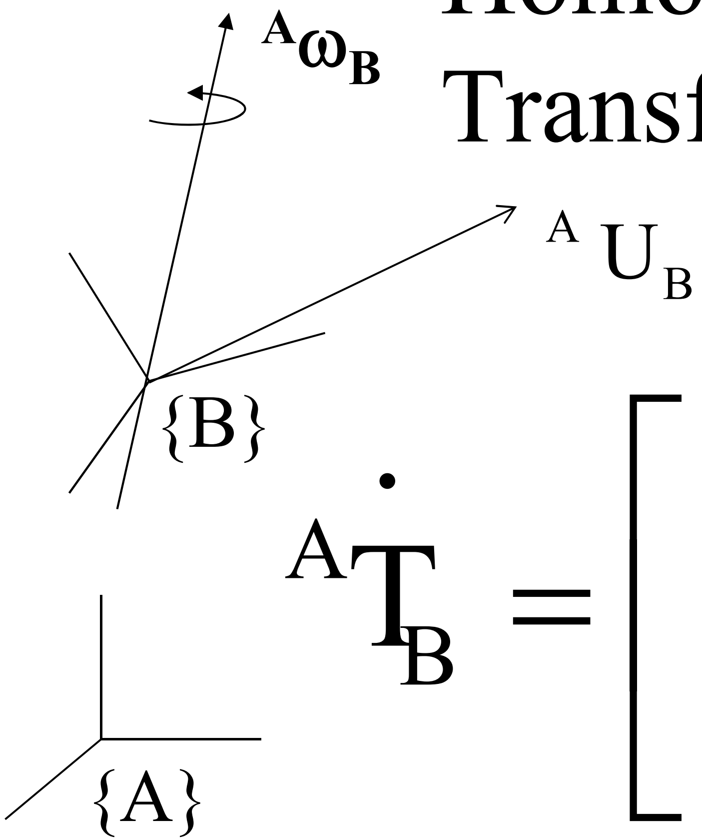
$$ {}^{C}\hat{\omega}_{B} = {}^{C}R_{A}\,{}^{A}\hat{\omega}_{B}\,{}^{C}R_{A}^{T} = {}^{C}R_{A}\,{}^{A}\hat{\omega}_{B}\,{}^{A}R_{C} $$

$\uparrow$ 3x3 equivalent of 3 x 1 vector

$\nearrow$ 3x3 equivalent of 3 x 1 vector

Verify yourself!

Homogeneous Transformation Matrix



Translational velocity

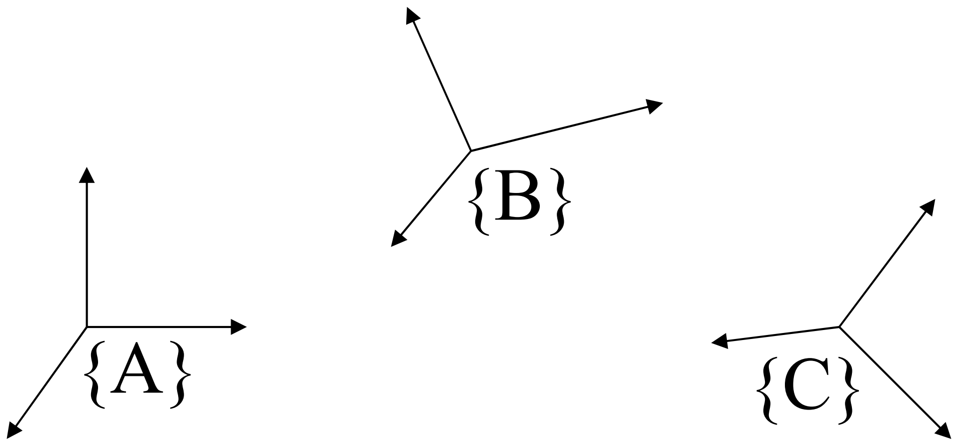
$$
{}^{A}\dot{T}_{B} = \begin{bmatrix} {}^{A}\hat{\omega}_{B}\;{}^{A}R_{B} & {}^{A}U_{B} \\ 0 & 0 \end{bmatrix}
$$

Time derivative of 4 x 4 homogeneous transformation matrix

${}^{A}\dot{\omega}_{B}$ Angular velocity

Simultaneous Rotational & Translational Velocities

Given: Frames A, B & C



{B} & {C} in motion with respect to {A}

Find: Relationships between velocities

$$\mathbf{{}^{A}T_{C} = {}^{A}T_{B}\,{}^{B}T_{C}}$$

Differentiate with respect to time

$${}^{A}T_C = {}^{A}T_B\,{}^{B}T_C$$

$${}^{A}\dot{T}_C = {}^{A}T_B\,{}^{B}\dot{T}_C + {}^{A}\dot{T}_B\,{}^{B}T_C$$

$$\begin{pmatrix} {}^{A}\hat{\omega}_C\,{}^{A}R_C & {}^{A}u_C \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} {}^{A}R_B & {}^{A}p_B \\ 0 & 1 \end{pmatrix}\begin{pmatrix} {}^{B}\hat{\omega}_C\,{}^{B}R_C & {}^{B}u_C \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} {}^{A}\hat{\omega}_B\,{}^{A}R_B & {}^{A}u_B \\ 0 & 0 \end{pmatrix}\begin{pmatrix} {}^{B}R_C & {}^{B}p_C \\ 0 & 1 \end{pmatrix}$$

$${}^{A}u_C = {}^{A}R_B\,{}^{B}u_C + \underbrace{{}^{A}\hat{\omega}_B\,{}^{A}R_B\,{}^{B}p_C}_{{}^{A}\omega_B \times ({}^{A}R_B\,{}^{B}p_C)} + {}^{A}u_B$$

$${}^{A}\hat{\omega}_C\,{}^{A}R_C = {}^{A}R_B\,{}^{B}\hat{\omega}_C\,{}^{B}R_C + {}^{A}\hat{\omega}_B\,\underbrace{{}^{A}R_B\,{}^{B}R_C}_{{}^{A}R_C}$$

$${}^{A}\hat{\omega}_{C}\,\cancel{{}^{A}R_{C}} = {}^{A}R_{B}\,{}^{B}\hat{\omega}_{C}\,\underbrace{{}^{B}R_{A}\,\cancel{{}^{A}R_{C}}}_{{}^{B}R_{C}\text{ (from previous slide)}} + {}^{A}\hat{\omega}_{B}\,\cancel{{}^{A}R_{C}}$$

$${}^{A}\hat{\omega}_{C} = \underbrace{{}^{A}R_{B}\,{}^{B}\hat{\omega}_{C}\,{}^{B}R_{A}}_{3\times 3} + {}^{A}\hat{\omega}_{B}$$

$$\underbrace{}_{}{}^{A}\omega_{C} = {}^{A}R_{B}\,{}^{B}\omega_{C} + {}^{A}\omega_{B}$$

(see slide 15 before)

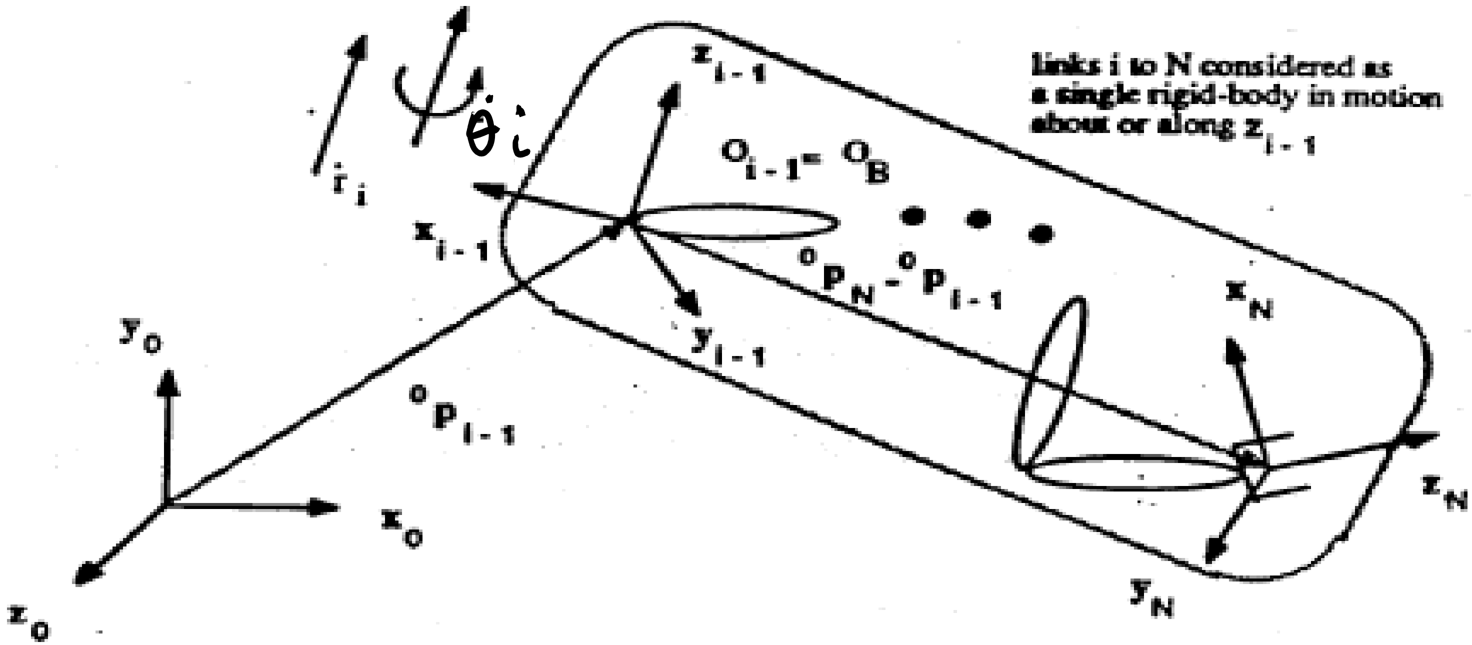
Summary: Simultaneous Rotational & Translational Velocities

$$
\begin{aligned}
{}^{A}u_{C} &= {}^{A}u_{B} + {}^{A}R_{B}\,{}^{B}u_{C} + {}^{A}\hat{\omega}_{B}\left({}^{A}P_{C} - {}^{A}P_{B}\right) \\
&= \; u_{B} + {}^{A}R_{B}\,{}^{B}u_{C} + {}^{A}\omega_{B} \times \left({}^{A}P_{C} - {}^{A}P_{B}\right) \\
&= \; u_{B} + {}^{A}R_{B}\,{}^{B}u_{C} + {}^{A}\omega_{B} \times \left({}^{A}R_{B}\,{}^{B}P_{C}\right)
\end{aligned}
$$

$$
{}^{A}\omega_{C} = {}^{A}\omega_{B} + {}^{A}R_{B}\,{}^{B}\omega_{C}
$$

Computation Of End-Effector Velocity

$$(6\text{x}1) \quad v_N = \begin{pmatrix} u_N \\ \omega_N \end{pmatrix} = f(\underbrace{q}_{\text{joint position}}, \underbrace{\dot{q}}_{\text{joint velocities}})$$



Computation Of End-Effector Velocity

Let us examine the contribution of the ith joint motion to end-effector velocity. We set all other joint velocities ϕ :

$\dot{\mathbf{q}}_\mathbf{i} \neq 0 \qquad \dot{\mathbf{q}}_\mathbf{1} = \dot{\mathbf{q}}_\mathbf{2} = \ldots = \dot{\mathbf{q}}_\mathbf{i-1} = \dot{\mathbf{q}}_\mathbf{i+1} = \ldots \dot{\mathbf{q}}_\mathbf{N} = \phi$

so motion is occurring with respect to $\mathbf{z_{i-1}}$ axis

For joint i rotational

$$\boldsymbol{\omega}_\mathbf{i} = \mathbf{z_{i-1}}\, \dot{\mathbf{q}}_\mathbf{i}$$

$$\mathbf{u_i} = \boldsymbol{\omega}_\mathbf{i} \text{ x } \mathbf{R_{i-1}}\, {}^{\mathbf{i-1}}\mathbf{p_N} = \mathbf{z_{i-1}}\, \dot{\mathbf{q}}_\mathbf{i} \text{ x } (\, \mathbf{p_N} - \mathbf{p_{i-1}}\,)$$

$$= \mathbf{z_{i-1}} \text{ x } (\, \mathbf{p_N} - \mathbf{p_{i-1}}\,)\, \dot{\mathbf{q}}_\mathbf{i}$$

Note that $\mathbf{o_{i-1}}$ has no translational velocity

origin of frame i-1 which contains $\mathbf{z_{i-1}}$
since joint is rotational

Computation Of End-Effector Velocity

For a translational joint i,

$$\omega_i = 0$$

$$\mathbf{u}_i = \mathbf{z}_{i-1}\, \dot{q}_i$$

The total velocity of the end-effector during coordinated motion is the superposition of all the elementary velocities that represent single joint motion:

$$v_N = \begin{bmatrix} u_N \\ \omega_N \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{N} u_i \\ \sum_{i=1}^{N} \omega_i \end{bmatrix}$$

Computation Of End-Effector Velocity

$$\mathbf{v_N} = \underbrace{\begin{pmatrix} \overset{6\times 1}{J_1} & J_2 & J_3 & \dots & J_N \end{pmatrix}}_{6\text{xN } J(q)} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_N \end{pmatrix}$$

Column J_i represents motion contribution of joint i

J(q) = Jacobian matrix
Cartesian $\leftrightarrow$ joint space

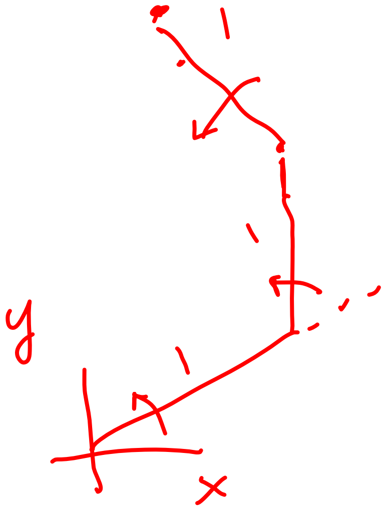
Computation Of End-Effector Velocity

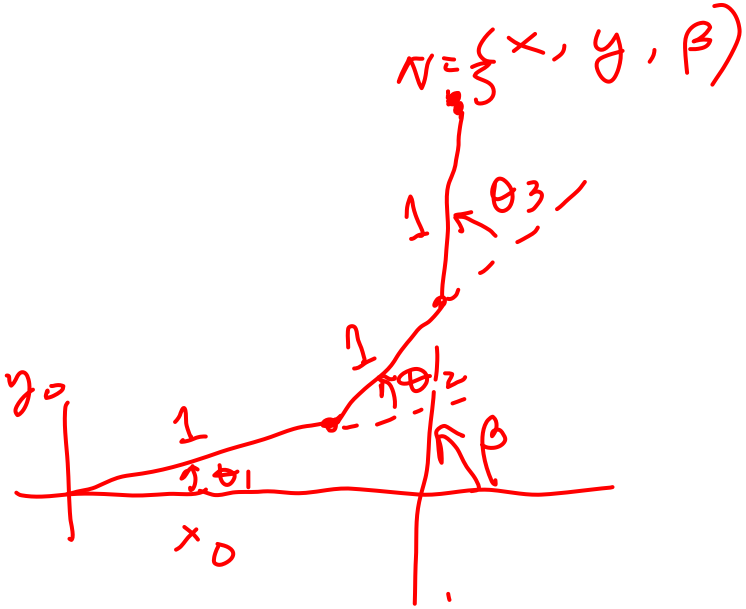
For a translational joint i

$$\mathbf{J_i} = \begin{bmatrix} z_{i-1} \\ 0 \end{bmatrix}$$

For a rotational joint i

$$\mathbf{J_i} = \begin{bmatrix} z_{i-1} \times (p_N - p_{i-1}) \\ z_{i-1} \end{bmatrix}$$





Find the Robot Jacobian

$$\begin{pmatrix} {}^A u_N \\ {}^A w_N \end{pmatrix} = \underset{6\times 3}{J}\,\dot{q} = \begin{pmatrix} J_1 & J_2 & J_3 \end{pmatrix} \begin{pmatrix} \dot\theta_1 \\ \dot\theta_2 \\ \dot\theta_3 \end{pmatrix}$$

$J = ?$

$${}^A V = {}^A J \dot{q}$$

$$\begin{pmatrix} {}^0 u_3 \\ {}^0 w_3 \end{pmatrix} = J \dot{q}$$

1. Assign Frames
2. DH Table
3. $J_i = \begin{pmatrix} {}^A z_{i-1} \times ({}^A p_3 - {}^A p_{i-1}) \\ {}^A z_{i-1} \end{pmatrix}$

${}^0 z_{i-1}$ = 3 col ${}^0 z_0$ ${}^0 z_1$ ${}^0 z_2$

p = 4th col ${}^0 p_0$, ${}^0 p_1$, ${}^0 p_2$, ${}^0 p_3$

Jacobian Transformations

- Velocities expressed in different frames

$$ {}^{A}\mathbf{v}_{\mathbf{N}} \leftrightarrow {}^{B}\mathbf{v}_{\mathbf{N}} \qquad \left\{ \begin{array}{l} \text{N = End Effector} \\ \text{B = may be a link coord} \\ \quad\ \text{frame that is held} \\ \quad\ \text{instantaneously constant} \end{array} \right. $$

For ${}^{A}\mathbf{R}_{\mathbf{B}}$ and ${}^{A}\mathbf{p}_{\mathbf{B}}$ constants

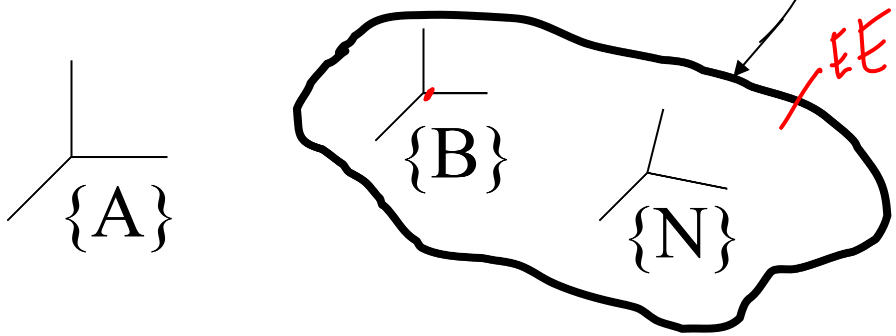
$$ {}^{A}v_{N} = \begin{pmatrix} {}^{A}u_{N} \\ {}^{A}\omega_{N} \end{pmatrix} = \underbrace{\left[\begin{array}{c:c} {}^{A}R_{B} & 0 \\ \hdashline 0 & {}^{A}R_{B} \end{array} \right]}_{\mathbf{J}} \underbrace{\begin{pmatrix} {}^{B}u_{N} \\ {}^{B}\omega_{N} \end{pmatrix}}_{{}^{B}\mathbf{v}_{\mathbf{N}}} $$

Jacobian Transformations

- Diff pts on End-Effector

For ${}^{B}\mathbf{R}_N$ and ${}^{B}\mathbf{p}_N$ constants

B & N are attached to a <u>rigid body</u> moving with respect to A:



N = } Two Frames Attached to End-Effector
B =

Jacobian Transformations

$$ {}^{A}\mathbf{u}_{N} = {}^{A}\mathbf{u}_{B} + {}^{A}\boldsymbol{\omega}_{B} \times ({}^{A}\mathbf{p}_{N} - {}^{A}\mathbf{p}_{B}) $$

$$ {}^{A}\boldsymbol{\omega}_{N} = {}^{A}\boldsymbol{\omega}_{B} $$

$$ {}^{A}v_{N} = \begin{pmatrix} {}^{A}u_{N} \\ {}^{A}\omega_{N} \end{pmatrix} = \underbrace{\left[\begin{array}{c|c} I & -({}^{A}p_{N} - {}^{A}p_{B}) \\ \hline 0 & I \end{array}\right]}_{\text{another } \mathbf{J}} \begin{bmatrix} {}^{A}u_{B} \\ {}^{A}\omega_{B} \end{bmatrix} $$

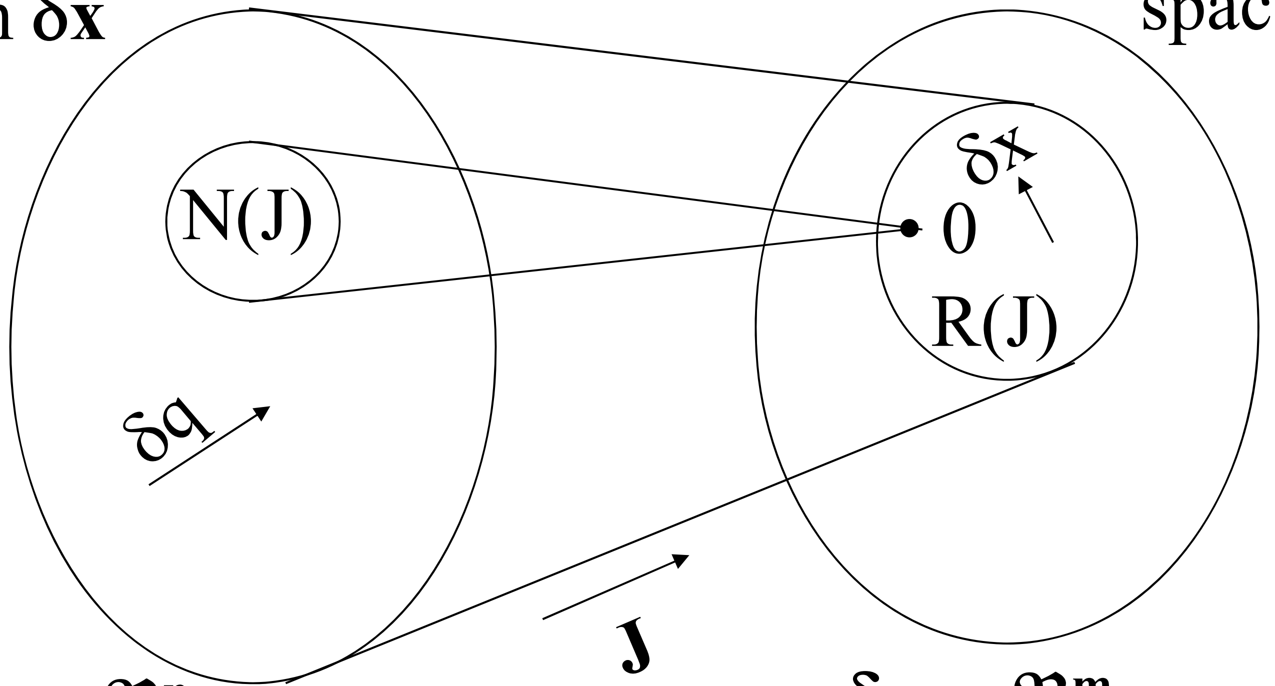
Robot Kinematics of Velocity

N(J) = Null space of J

δq → produces no motion **δx**

R(J) = Range Space of **J**

= or column space of **J**



$\boldsymbol{\delta q} \in \Re^{\mathbf{n}}$
Joint Space

$\boldsymbol{\delta x} \in \Re^{\mathbf{m}}$
End-Effector Space

Inverse Kinematics of Velocity

First: Convert $\delta \mathbf{x}$ to $\delta \mathbf{x_0} \in \mathbf{R}^{\mathbf{m_o}}$(velocity, basic kinematic model)

$$\underbrace{\delta \mathbf{x_0}}_{\mathbf{R}^{\mathbf{m_o}}} = \mathbf{J_0}\, \underbrace{\mathbf{c}}_{\mathbf{R}^{\mathbf{n}}} \qquad ({}^{\mathbf{0}}\dot{\mathbf{v}}_{\mathrm{N}} = \mathbf{J_0}\, \dot{\mathbf{q}})$$

$$m_0 \leq 6$$

General solution exists if & only if Rank $\mathbf{J_0} = \min(m_0, n)$

Inverse Kinematics of Velocity

Case 1: $m_0 = n \quad \leq 6$

$\mathbf{J}^{\#}(\mathbf{q}) = \mathbf{J}^{-1}$ (possible problem with singularity, $\mathbf{J}^{-1}$ may not exist)

Case 2: $m_0 > n, \quad m_0 \leq 6$ (not interesting/useful case, task shall be $\leq$ n)

over-determined system: more eqns than unknowns.

$\mathbf{J}^{\#} = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T$ = left pseudo inverse

= exists only if Rank $\mathbf{J}$ = n

Sol'n minimizes $\| \mathbf{J}\delta\mathbf{q} - \delta\mathbf{x}_0 \|_2$

Inverse Kinematics of Velocity

Case 3: $m_0 < n$, $m_0 \leq 6$ (Redundant Robots)

underdetermined system = less eqns than unknowns

$\mathbf{J}^{\#} = \mathbf{J}^T (\mathbf{J}\,\mathbf{J}^T)^{-1}$ = right pseudo inverse

= exists only if Rank $\mathbf{J} = m_0$

Sol'n minimizes $\| \boldsymbol{\delta}\mathbf{q} \|_2$

Manipulator Singularities

- Joint configuration (set of joint positions) where the Jacobian is not full-rank
- Determinant of $JJ^T = 0$, or $Det(J^TJ) = 0$
- There are no joint motions to achieve the end-effector motion when the robot is at a singular configuration

Determining Manipulator Singularities

- J is m x n in general, with $n \geq m$
  - Number of joints (n) must at least equal the number of task degrees of freedom (m)
  - Can have more joints needed to accomplish the task (Redundant robots)
- Determinant $(J\ J^T)$ is equal zero at singularities; or
- There is no subset of joints (m joints) that can do the m-DOF task.

Resolved Motion Rate Control

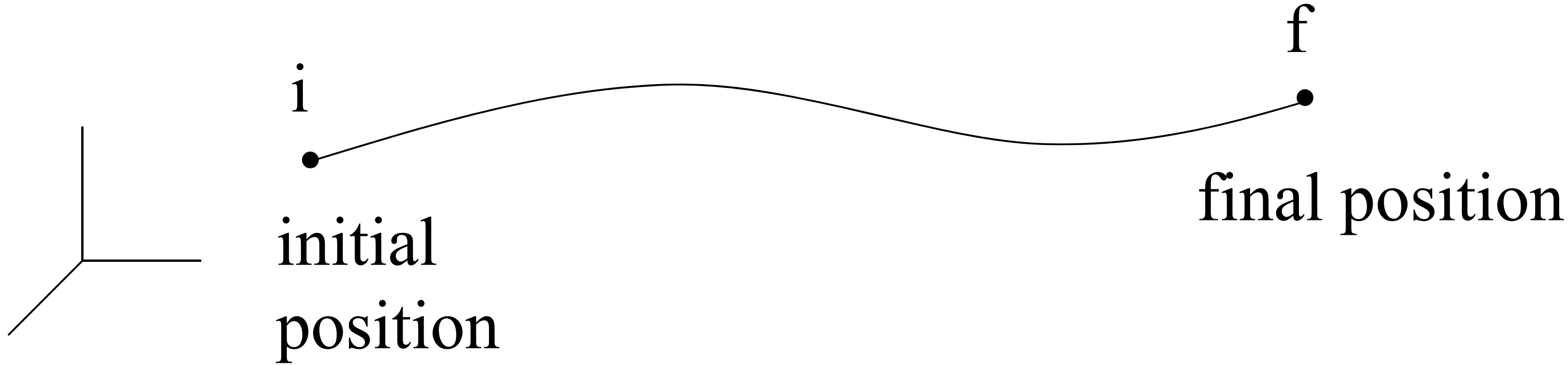
- Kinematic Control without the need for solving the inverse Kinematics of Position

- Need Joint level control which is available in robot controllers

- Command joint motion (δq) such that desired e-e motion (δx) is achieved

Resolved Motion Rate Control

1) Given a Trajectory $x(t) \in R^m$ in task space



2) Divide Trajectory into small segments according to sample time on reference Trajectory update rate

3) At x_k, compute $J(q_k)$

Resolved Motion Rate Control

4) Compute $\Delta x_k = x_{k+1} - x_k$ (position and orientation error)

5) Convert orientation error into 3 x 1 vector ($d\Phi$)

6) Compute $\Delta q = J^{\#}(q_k)\, \Delta\, x_{0,\, k}$

7) Command δq to robot controller
(Robot moves from q_k to q_{k+1}) ($\delta q = q_{k+1} - q_k$)

8) Go to step 3 until x_k reaches x_f