CHAPTER 1

Position and Orientation Transformations and Robot Kinematics of Position

After this chapter, the students are expected to learn the following:

1. Definitions of positions and orientations of rigid bodies
2. Analysis of different representations for orientation
3. Transformation of coordinates

After this chapter, the students are expected to learn the following:

4. Determination of new positions and orientations after a sequence of rigid body motions

5. Kinematics modeling of robotic manipulators

6. Forward and inverse kinematics of position

SPATIAL DESCRIPTION

– used to specify spatial attributes of various objects with which a manipulation system deals

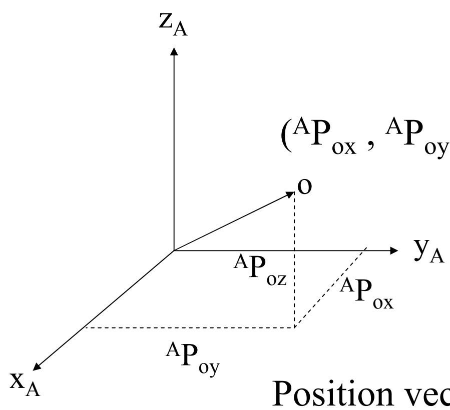
– universe coordinate frame is implicit

– use Cartesian coordinate frames

POSITION

– attribute of a point

$$^{A}P_{o} = \begin{bmatrix} ^{A}P_{ox} \\ ^{A}P_{oy} \\ ^{A}P_{oz} \end{bmatrix} \in \Re^{3}$$



$(^{A}P_{ox}, ^{A}P_{oy}, ^{A}P_{oz})$ = Cartesian coordinates of pt.0 expressed in frame A

Position vector - (coords has magnitude & direction, origin impt)

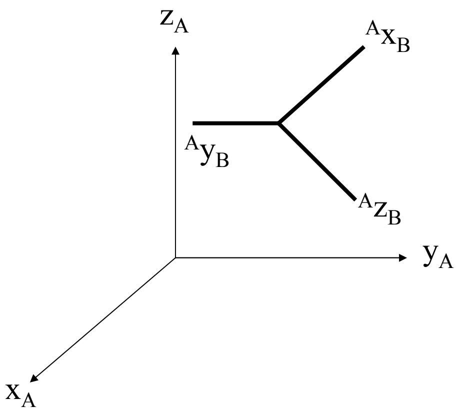
ORIENTATION

– attribute of a body

– attach a Cartesian coordinate to the body

– orientation of Cartesian coordinate system axis

ORIENTATION



$(^A x_B, {}^A y_B, {}^A z_B)$

- unit vectors along each axis of frame B
- free vectors, only direction is relevant

$$^A R_B = \left(^A x_B \;\; ^A y_B \;\; ^A z_B \right)$$

$$= \begin{pmatrix} ^A x_{Bx} & ^A y_{Bx} & ^A z_{Bx} \\ ^A x_{By} & ^A y_{By} & ^A z_{By} \\ ^A x_{Bz} & ^A y_{Bz} & ^A z_{Bz} \end{pmatrix} \in \Re^{3\times 3}$$

Rotation Matrix

- Since each column of the rotation matrix represents unit vectors along the x, y and z directions of the Cartesian coordinates, then they should be orthogonal to each other
- Each column obeys unit length constraints
- Cartesian coord frame → right hand rule

$\therefore \det(^A R_B) = +1$

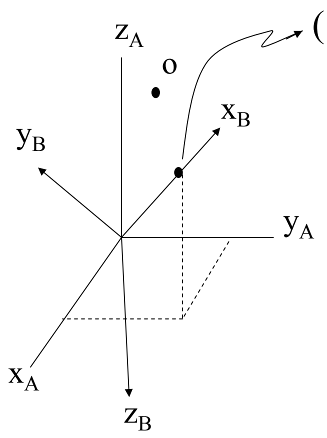
→ $^A R_B$ is a Proper Orthogonal Matrix

$^A R_B^{-1} = {}^A R_B^T = {}^B R_A$ or $^A R_B \, {}^A R_B^T = I$

Rotation Matrix

- All possible orientations of a rigid body (i.e. coordinate frame attached to the body) can be uniquely specified by a rotation matrix
- 3 independent parameter are only needed

Coordinate Transformations & Rigid Body Motion



(1,1,0) in Frame B

↳ Same point has coordinates ($^A x_{B,x}$, $^A x_{B,y}$, $^A x_{B,z}$) in Frame A

– Similarly, (0,1,0) in Frame B has coordinates
→ 2nd column of $^A R_B$ → in Frame A

– And (0,0,1) in Frame B has coordinates
→ 3rd column of $^A R_B$ → in Frame A

Coordinate Transformations & Rigid Body Motion

Generalizing – a point, O, with coordinates in Frame B can be transformed to coordinates in Frame A if the relative orientation between A & B ($^{A}R_{B}$) is known

$$^{A}P_{o} = {}^{A}R_{B}\,{}^{B}P_{o}$$

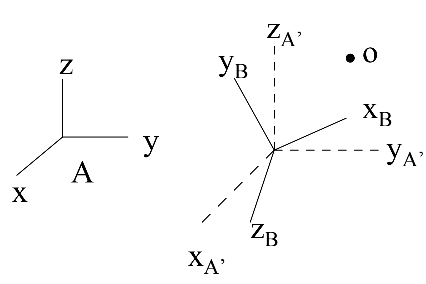
$^{B}P_{o}$: Coordinates of O in B

$^{A}P_{o}$: Coordinates of O in A

Note the ASSUMPTION : – angles of Frames A & B are coincident
– only difference is in orientation

Coordinate Transformations & Rigid Body Motion

What if the two origins are not coincident?
– i.e. , translation between frames A & B



A' = frame parallel to A (only different origins)

$$^{A'}P_{o} = {}^{A'}R_{B}\,{}^{B}P_{o} \qquad \text{(from before)}$$

Coordinate Transformations & Rigid Body Motion

$^{A}P_{o} = {}^{A}P_{B} + {}^{A'}P_{o}$ } can only add two position vectors if they are expressed in Frames that are parallel to each other

A is parallel to A' – so that adding respective coordinates make sense

Therefore

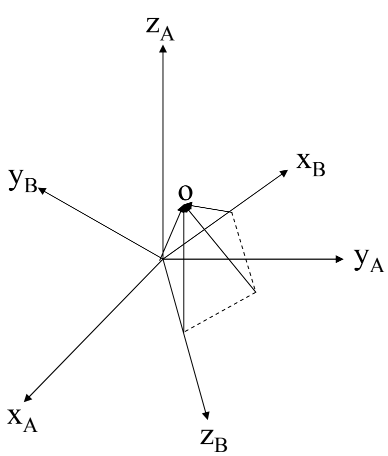
$$^{A}P_{o} = {}^{A}P_{B} + {}^{A}R_{B}\,{}^{B}P_{o}$$

since $^{A}R_{B} = {}^{A'}R_{B}$ because A is // A'

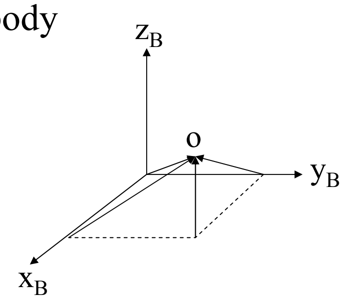
Transforming O to a frame // to A ($^{A}R_{B}\,{}^{B}P_{o}$)

Then adding the relative displacement of A & B

Duality (of Coordinate Transformation) with Rigid Body Motion

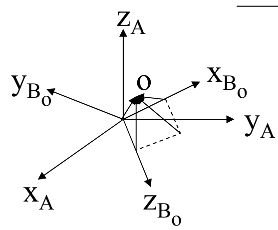


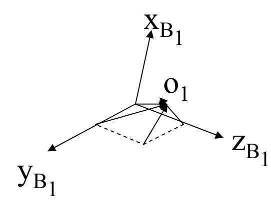
- Imagine a Rigid Body (Pyramid) whose apex is at pt. O
- Frame B is attached to the rigid body



Duality (of Coordinate Transformation) with Rigid Body Motion

– The body B is initially at ${}^{A}R_{Bo} + {}^{A}P_o = {}^{A}R_{Bo}{}^{B}P_o$
– Body B undergoes a motion such that the new Position & Orientation of B are ${}^{A}R_{B_1}$ & ${}^{A}P_{B_1}$

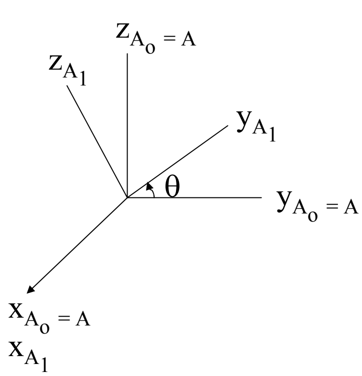




– New Coordinates of O in A, ${}^{A}P_{O_1}$ can be found :

$${}^{A}P_{o_1} = {}^{A}P_{B_1} + {}^{A}R_{B_1}{}^{B_1}P_o$$

Elementary (Basic, Fundamental) Motions



– Initially, Body is at A_o
– A undergoes a Rotation about xAo by θ & A is now at A_1 after the rotation

$${}^{Ao}R_{A1} = \text{Rot}(x,\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix}$$

Similarly, a Rotation about y by θ :

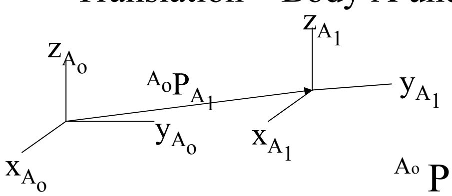
$${}^{Ao}R_{A1} = \text{Rot}(y,\theta) = \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix}$$

Elementary (Basic, Fundamental) Motions

Similarly, a rotation about z by θ :

$$^{A_o}R_{A_1} = \text{Rot}(z,\theta) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Translation – Body A undergoes pure rotation (only)



$$^{A_o}P_{A_1} = \begin{pmatrix} ^{A_o}P_{A_{1,x}} \\ ^{A_o}P_{A_{1,y}} \\ ^{A_o}P_{A_{1,z}} \end{pmatrix}$$

→ Coordinates of A_1 in A_o

Homogeneous Transformations

- created so that rotations and translations are treated uniformly. (i.e. as matrix multiplications)
- vectors have 4 components
  - ~ scaling factor = 0 (free vectors, unit vectors along axis)
  - ~ scaling factor = 1 (position vectors)

$$^{A}P_{B} = \begin{bmatrix} ^{A}P_{Bx} \\ ^{A}P_{By} \\ ^{A}P_{Bz} \\ 1 \end{bmatrix} \qquad ^{A}x_{B} = \begin{bmatrix} ^{A}x_{Bx} \\ ^{A}x_{By} \\ ^{A}x_{Bz} \\ \phi \end{bmatrix}$$

Homogeneous Transformations

$$\begin{bmatrix} {}^{A}P_{C} \\ \hline 1 \end{bmatrix} = \begin{bmatrix} {}^{A}R_{B} & {}^{A}P_{B} \\ \hline 0 & 1 \end{bmatrix} \begin{bmatrix} {}^{B}P_{C} \\ \hline 1 \end{bmatrix}$$

$$\Rightarrow \quad {}^{A}P_{C} = {}^{A}R_{B}\,{}^{B}P_{C} + {}^{A}P_{B}$$

$${}^{A}T_{B} = \begin{bmatrix} {}^{A}R_{B} & {}^{A}P_{B} \\ \hline 0 & 1 \end{bmatrix} \in \Re^{4\times 4}$$

= homogeneous transformation matrix

= describes the position and orientation of frame B in frame A

Chain rule :

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

Inverse of a Homogeneous Transformation Matrix

Given ${}^{A}T_{B}$, Find ${}^{B}T_{A}$

$${}^{A}T_{B}\,{}^{B}T_{A} = I$$

$$\begin{bmatrix} {}^{A}R_{B} & {}^{A}P_{B} \\ \hline \underline{0} & 1 \end{bmatrix} \begin{bmatrix} X & Y \\ \hline \underline{0} & 1 \end{bmatrix} = \begin{bmatrix} I & \underline{0} \\ \hline \underline{0} & I \end{bmatrix}$$

${}^{A}R_{B}\, X = I \quad \Rightarrow \quad X = {}^{A}R_{B}^{-1} = {}^{A}R_{B}^{T} = {}^{B}R_{A}$

${}^{A}R_{B}\, Y + {}^{A}P_{B} = \underline{0}$

${}^{A}R_{B}\, Y = -{}^{A}P_{B}$

$Y = -{}^{A}R_{B}^{-1}\,{}^{A}P_{B} = {}^{A}R_{B}^{T}\,{}^{A}P_{B}$

Inverse of a Homogeneous Transformation Matrix

$$\therefore {}^{A}T_{B}^{-1} = \left[\begin{array}{c:c} {}^{A}R_{B}^{T} & -{}^{A}R_{B}^{T}\ {}^{A}P_{B} \\ \hdashline \underline{0} & 1 \end{array}\right]$$

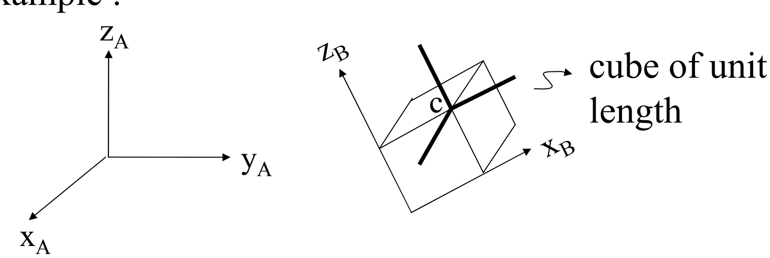
Note that

$$(\,\underline{n}\quad \underline{0}\quad \underline{a}\,)^{T} = {}^{A}R_{B}^{T}$$

$${}^{A}R_{B}^{T} = {}^{A}P_{B} = \begin{array}{l} \underline{n} \bullet {}^{A}P_{B} \\ \underline{0} \bullet {}^{A}P_{B} \\ \underline{a} \bullet {}^{A}P_{B} \end{array}$$

Inverse of a Homogeneous Transformation Matrix

Example :



Given : Initial Position & Orientation of cube specified by ${}^{A}T_{B}$

Find : new coordinates of pt. c (corner of cube) after the cube is rotated by θ about z_A

Inverse of a Homogeneous Transformation Matrix

Example :

$$^{A}P_{C} = \underbrace{Rot(z,90^{\circ})\ {}^{A}T_{B}}_{^{A}P_{B'}} \underbrace{\begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix}}_{^{B'}P_C = {}^{B''}P_C}$$

since C is attached rigidly to B

Other Orientation Representation

Euler angles α, β, γ

$$^{A}R_{B} = Rot(z,\alpha)Rot(y,\beta)Rot(z,\gamma)$$

$$= \begin{pmatrix} \cos\alpha\cos\beta\cos\gamma - \sin\alpha\sin\gamma & -\cos\alpha\cos\beta\sin\gamma - \sin\alpha\cos\gamma & \cos\alpha\sin\beta \\ \sin\alpha\cos\beta\cos\gamma + \cos\alpha\sin\gamma & -\sin\alpha\cos\beta\sin\gamma + \cos\alpha\cos\gamma & \sin\alpha\sin\beta \\ -\sin\beta\cos\gamma & \sin\beta\sin\gamma & \cos\beta \end{pmatrix}$$

Inverse Transf. : Given $^{A}R_{B}$, Find α, β, γ

$$\begin{pmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{pmatrix}$$

Other Orientation Representation

If $|a_z| \neq 1$

$$\alpha = \text{ATANZ}(a_y, a_x), \beta = \text{ATANZ}(\pm\sqrt{a_x^2 + a_y^2}, a_z), \gamma = \text{ATANZ}(o_z, -n_z)$$

(2 solutions)

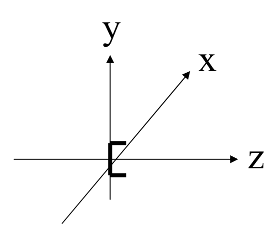
If $a_z = 1$, $\beta = 0°$, $\gamma + \alpha = \text{ATANZ}(n_y, n_x)$ (infinte # of sol'ns)

If $a_z = -1$, $\beta = 180°$, $\gamma - \alpha = \text{ATANZ}(n_y, -n_x)$ (infinite # of sol'ns)

Mathematical Singularity occurs when $|a_z| = 1$,
α and γ describe the same rotation & cannot be computed separately

ROLL - PITCH - YAW Angles (ϕ, θ, φ)

$${}^{A}R_{B} = \text{Rot}(z, \phi)\ \text{Rot}(y, \theta)\ \text{Rot}(x, \varphi)$$



Roll ↑ Pitch ↑ Yaw ↑

$$= \begin{pmatrix} \cos\phi\cos\theta & \cos\phi\sin\theta\sin\varphi - \sin\phi\cos\varphi & \cos\phi\sin\theta\cos\varphi + \sin\phi\sin\varphi \\ \sin\phi\cos\theta & \sin\phi\sin\theta\sin\varphi + \cos\phi\cos\varphi & \sin\phi\sin\theta\cos\varphi - \cos\phi\sin\varphi \\ -\sin\theta & \cos\theta\sin\varphi & \cos\theta\cos\varphi \end{pmatrix}$$

Inverse Transformation

If $|n_z| \neq 1$

ϕ=ATANZ(n_y,n_x) θ=ATANZ($-n_z$, $\pm\sqrt{n_x^2+n_y^2}$) φ=ATANZ(o_z,a_z)

(2 Solutions)

If n_z = +1, θ=270°, $\varphi+\phi$ =ATANZ($-o_x$,o_y)

If n_z = -1, θ=90°, $\varphi-\phi$ =ATANZ (o_x,o_y)

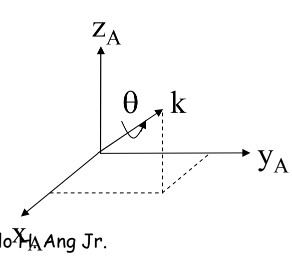
Mathematical Singularity occurs when $|n_z|$ = 1
φ & ϕ describe the same rotation & cannot be computed separately

Four-Parameter Representation For Orientation

– quadruple of ordered real parameters consisting of one scalar (angle of rotation) and one vector (axis of rotation)

Any rotation matrix can be represented by rotation θ of space about a fixed axis k = $\begin{pmatrix} k_x \\ k_y \\ k_z \end{pmatrix}$

$^{A}R_{B}$ = Rot (k , θ)



A → B after rotation
K is a vector expressed in A

Four-Parameter Representation For Orientation

$$Rot(k, \theta) = \begin{pmatrix} k_x^2 \mathrm{vers}\theta + \cos\theta & k_y k_x \mathrm{vers}\theta - k_z \sin\theta & k_z k_x \mathrm{vers}\theta + k_y \sin\theta \\ k_x k_y \mathrm{vers}\theta + k_z \sin\theta & k_y^2 \mathrm{vers}\theta + \cos\theta & k_z k_y \mathrm{vers}\theta - k_x \sin\theta \\ k_x k_z \mathrm{vers}\theta - k_y \sin\theta & k_y k_z \mathrm{vers}\theta + k_x \sin\theta & k_z^2 \mathrm{vers}\theta + \cos\theta \end{pmatrix}$$

where $\mathrm{vers}\,\theta = 1 - \cos\theta$

Try proving this

Hint : imagine $\vec{k}$ to be the x, y or z axis of a new frame. Rot (h ,θ) then becomes one of the elementary rotations [Rot (x, θ) , Rot (y, θ) or Rot (z, θ)]

Inverse Problem

Given ${}^AR_B$ → Find k, θ

$$\text{Let } {}^AR_B = \begin{pmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{pmatrix} = \begin{pmatrix} \text{function of } (k, \theta) \end{pmatrix}$$

given — previous page

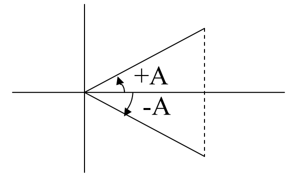
From the diagonal elements,

$\mathrm{Trace}({}^AR_B) = 1 + 2\cos\theta$

$\cos\theta = \tfrac{1}{2}(\mathrm{Trace}({}^AR_B) - 1)$

→ Two solutions for θ

$\theta = \pm A$



Inverse Problem

From : n_z & a_x elements

$$k_y = \frac{a_x - n_z}{2\sin\theta}$$

From : o_z & a_y elements

$$k_x = \frac{o_z - a_y}{2\sin\theta}$$

From : n_y & o_x elements

$$k_z = \frac{n_y - o_x}{2\sin\theta}$$

Valid only for $\sin\theta \neq 0$

Inverse Problem

If $\theta = 0^o$

$$^{A}R_{B} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{Identity}$$

k = any axis (infinite sol'ns)

If $\theta = 180^o$

$$^{A}R_{B} = \begin{pmatrix} -1+2k_z^2 & 2k_xk_y & 2k_xk_z \\ 2k_xk_y & -1+2k_y^2 & 2k_yk_z \\ 2k_xk_z & 2k_yk_z & -1+2k_z^2 \end{pmatrix}$$

= symmetric

$= Tr(^{A}R_{B}) = -3 + 2 = -1 = 1 + 2\cos\theta$

Inverse Problem

If all off diagonal terms are ϕ, one of k_x, k_y or $k_z = 1$ & the rest of the components of k are ϕ.
The component of k that is 1 has a value of 1 in the diagonal of ${}^{A}R_{B}$.

If no off diagonal term is zero,

$$k_x = \pm\sqrt{\frac{n_z n_y}{2o_z}}$$

$$k_y = \frac{n_y}{2k_x}$$

$$k_z = \frac{a_z}{2k_x}$$

2 solution for k
If one solution is $\vec{k}_1$, the other is $-\vec{k}_1$

Euler Parameters, or Quaternions

$$\lambda_0 = \cos\theta/2$$

$$\begin{pmatrix}\lambda_1\\ \lambda_2\\ \lambda_3\end{pmatrix} = \sin\ \theta/2\ \ \vec{k}$$

Note :
$\lambda_0^2 + \lambda_1^2 + \lambda_2^2 + \lambda_3^2 = 1$

$${}^{A}R_{B} = \begin{pmatrix} 2(\lambda_0^2 + \lambda_1^2) - 1 & 2(\lambda_1\lambda_2 - \lambda_0\lambda_3) & 2(\lambda_1\lambda_3 + \lambda_0\lambda_2) \\ 2(\lambda_1\lambda_2 + \lambda_0\lambda_3) & 2(\lambda_0^2 + \lambda_2^2) - 1 & 2(\lambda_2\lambda_3 - \lambda_0\lambda_1) \\ 2(\lambda_1\lambda_3 - \lambda_0\lambda_2) & 2(\lambda_2\lambda_3 + \lambda_0\lambda_1) & 2(\lambda_0^2 + \lambda_3^2) - 1 \end{pmatrix}$$

Inverse Transformation

Given ${}^{A}R_{B}$, Find λ_0, λ_1, λ_2, λ_3

$$\lambda_0 = \pm \tfrac{1}{2}\sqrt{n_x + o_y + a_z + 1} \quad \text{(2 Sol'ns)}$$

$$\lambda_1 = \frac{o_z - a_y}{4\lambda_0} \qquad \lambda_2 = \frac{a_x - n_z}{4\lambda_0} \qquad \lambda_3 = \frac{n_y - o_x}{4\lambda_0} \qquad \lambda_0 \neq 0$$

If $\lambda_0 = 0$,

($\theta = 180^o$)

$${}^{A}R_{B} = \begin{pmatrix} 2\lambda_1^2 - 1 & 2\lambda_1\lambda_2 & 2\lambda_1\lambda_3 \\ 2\lambda_1\lambda_2 & 2\lambda_2^2 - 1 & 2\lambda_2\lambda_3 \\ 2\lambda_1\lambda_3 & 2\lambda_2\lambda_3 & 2\lambda_3^2 - 1 \end{pmatrix}$$

Trace $({}^{A}R_{B}) = 2(\lambda_1^2 + \lambda_2^2 + \lambda_3^2) - 3 = 2 - 3 = -1$

Inverse Transformation

All diagonal terms are either :

- all zero, or
- all non-zero

$$\lambda_1 = \pm\sqrt{\frac{n_z n_y}{2o_z}}$$

$$\lambda_2 = \frac{n_y}{2\lambda_1}$$

$$\lambda_3 = \frac{a_z}{2\lambda_1}$$

2 solutions

If one sol'n is λ_1, λ_2, λ_3
other sol'n is $-\lambda_0$, $-\lambda_2$, $-\lambda_3$

Robot Kinematic Modeling

THE DENAVIT-HARTENBERG REPRESENTATION

In the robotics literature, the Denavit-Hartenberg (D-H) Representation has been used, almost universally, to derive the kinematic description of robotic manipulators. The appeal of the D-H representation lies in its algorithmic approach. In this handout, we provide an algorithm for the assignment of robotic coordinate frames, highlight the conventions associated with the D-H approach, and exemplify the development through the Puma and Stanford manipulators.

Robot Kinematic Modeling

STEP 1 : Number the Robot Joints and Links

Robotic manipulators are articulated, open kinematic chains of N rigid bodies (links) which are connected serially by joints. The links are numbered consecutively from the base (link 0) to the final end (link N). The joints are the points of articulation between the links and are numbered from 1 to N so that joint i connects links (i-1) and i. Each joint provides one degree-of-freedom which can either be a rotation or translation.There is no joint at the end of the final link.

Robot Kinematic Modeling

STEP 2 : Assign Link Coordinate Frames

To describe the geometry of robot motion, we assign a Cartesian coordinate frame (O_i;x_i,y_i,z_i) to each link, as follows:

- the z_i axis is directed along the axis of motion of joint (i + 1), that is, link (i + 1) rotates about or translates along z_i;
- the x_i axis lies along the common normal from the z_{i-1} axis to the z_i axis (if z_{i-1} is parallel to z_i, then x_i is specified arbitrarily, subject only to x_i being perpendicular to z_i); and

Robot Kinematic Modeling

STEP 2 : Assign Link Coordinate Frames

- the y_i axis completes the right-handed coordinate system.

The origin of the robot base frame O_0 can be placed anywhere in the supporting base and the origin of the last (end-effector) coordinate frame O_N is specified by the geometry of the end-effector.

Robot Kinematic Modeling

STEP 3 : Define the Joint Coordinates

The joint coordinate q_i is the angular displacement around z_{i-1} if joint *i* is rotational, or the linear displacement along z_{i-1} if joint *i* is translational. The N-dimensional space defined by the joint coordinates $(q_1,\ldots,q_N)$ is called the configuration space of the N DOF mechanism.

Robot Kinematic Modeling

STEP 4 : Identify the Link Kinematic Parameter

In general, four elementary transformations are required to relate the *i*-th coordinate frame to the (*i*-1)-th coordinate frame:

- Rotate an angle of θ_i (in the right-handed sense) about the z_{i-1} axis, so that the x_{i-1} axis is parallel to the x_i axis.
- Translate a distance of r_i along the positive direction of the z_{i-1} axis, to align the x_{i-1} axis with the x_i axis.

Robot Kinematic Modeling

STEP 4 : Identify the Link Kinematic Parameter

- Translate a distance of d_i along the positive direction of the $x_{i-1} = x_i$ axis, to coalesce the origins O_{i-1} and O_i.
- Rotate an angle of α_i (in the right-handed sense) about the $x_{i-1} = x_i$ axis, to coalesce the two coordinate systems.

The *i*-th coordinate frame is therefore characterized by the four D-H kinematic link parameters θ_i, r_i, d_i and α_i. If joint *i* is rotational, then $q_i = \theta_i$, and α_i, d_i and r_i are constant parameters which depend upon the

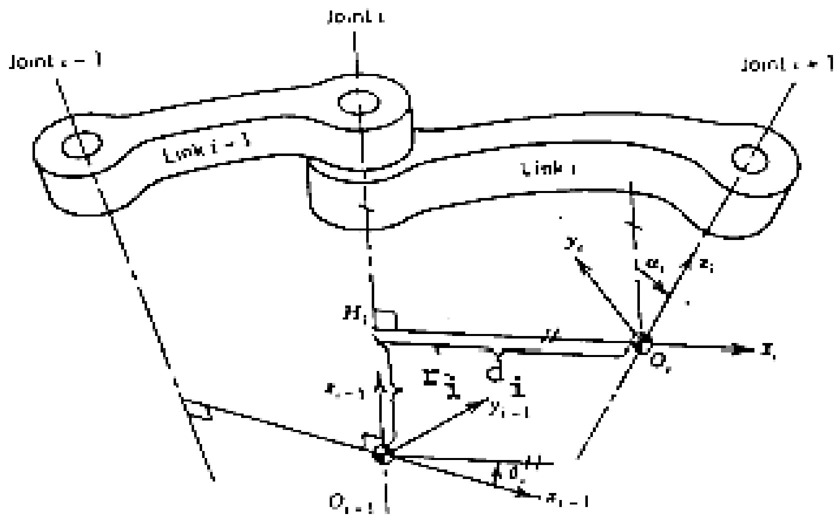
Robot Kinematic Modeling

STEP 4 : Identify the Link Kinematic Parameter

geometric properties and configuration of link *i*. If joint *i* is translational, then $q_i = r_i$, and d_i, α_i and θ_i are constant parameters which depend upon the configuration of link *i*. For both rotational and translational joints, r_i and θ_i are the distance and angle between links $(i-1)$ and i; d_i and α_i are the length and twist of link *i*.

Robot Kinematic Modeling

STEP 4 : Identify the Link Kinematic Parameter



Robot Kinematic Modeling

STEP 5 : Define the Link Transformation Matrices

The position and orientation of the i-th coordinate frame can be expressed in the $(i – 1)$-th coordinate frame by the following homogeneous transformation matrix:

$$A_i = \text{Rot}(z, \theta)\ \text{Trans}(0, 0, r_i)\ \text{Trans}(d_i, 0, 0)\ \text{Rot}(z, \alpha)$$

$$A_i(q_i) = {}^{i\text{-}1}T_i = \begin{pmatrix} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & d_i \cos\theta_i \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \cos\theta_i & d_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & r_i \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Robot Kinematic Modeling

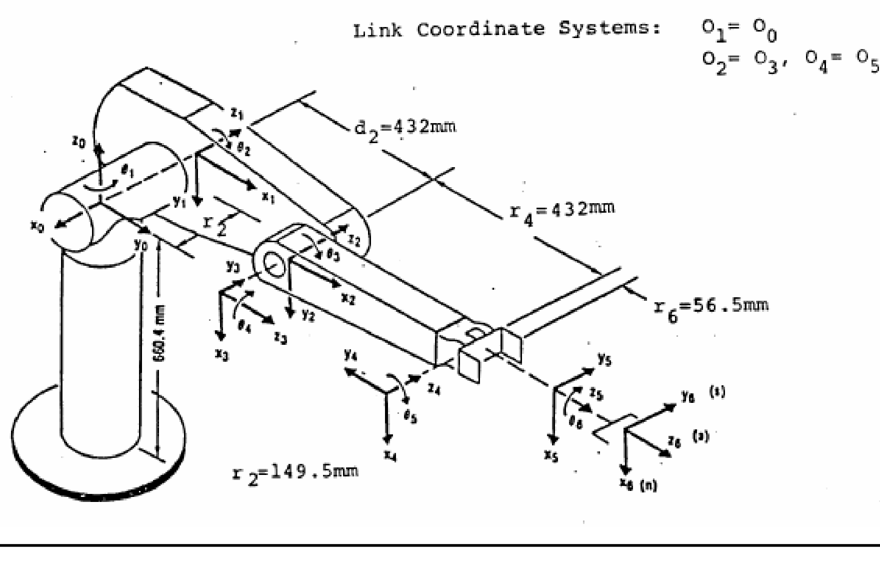
<u>*STEP 6 : Compute the Forward Transformation Matrix*</u>

The position and orientation of the end-effector coordinate frame is expressed in the base coordinate frame by the forward transformation matrix:

$$^{0}T_N(q_1, q_2, \ldots, q_N) = {}^{0}T_N = A_1 A_2 \ldots A_N = \begin{pmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Robot Kinematic Modeling

<u>EXAMPLE 1: The Puma Robot</u>

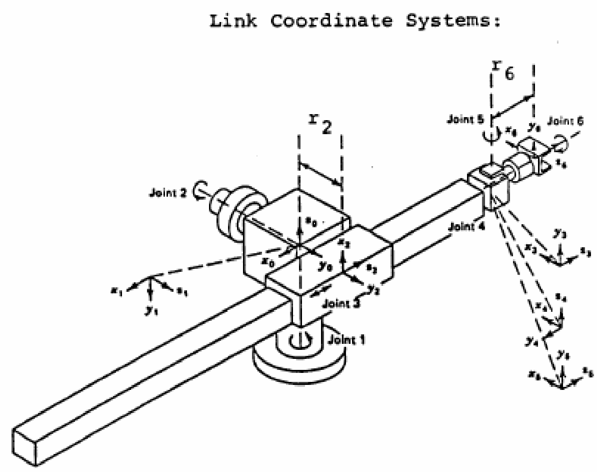


Robot Kinematic Modeling

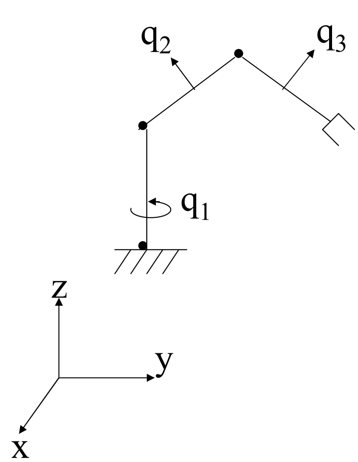
EXAMPLE 2: The Stanford Arm

$o_1 = o_0$
$o_3 = o_4 = o_5$

$r_2 = 16.2\text{cm}$
$r_6 = 24.7\text{cm}$



Forward Kinematic Problem



Given: q1, q2, q3….
(joint positions)

Find: End-Effector position P_E
and orientation R_E

Forward Kinematic Problem

1. Assign Cartesian Coordinate frames to each link (including the base ϕ & end-effector N)

2. Identify the joint variables and link kinematic parameters

3. Define the link transformation matrices. ${}^{i-1}T_i = A_i$

4. Compute the forward transformation
   $${}^{0}T_N(q_1, q_2, \ldots, q_N) = A_1 A_2 A_3 \ldots A_N = \begin{pmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Inverse Kinematic Problem

Given: Position & Orientation of END-EFFECTOR

Find: joint coordinates

$$\underline{{}^{0}T_N} \longrightarrow q_1, q_2, q_3, \ldots, q_N$$

Need to solve at most six independent equations in N unknowns.

Inverse Kinematic Problem

ISSUES

- Existence of solutions
  - Workspace
  - Dextrous Workspace
  - Less than 6 joints
  - Joint limits (practical)
- Multiple solutions
  - Criteria
  - Solvability — closed form (Algebraic / Geometric), numerical
  - number of solutions
    = 16 $\quad$ $d_i, r_i \neq 0$ $\quad$ for six points

Solution To Inverse Kinematics

$$^{0}T_N = {}^{0}T_1\,{}^{1}T_2\,{}^{2}T_3 \ldots {}^{N-1}T_N = A_1 A_2 A_3 \ldots A_N$$

$$\text{Given: } {}^{0}T_N = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ \phi & \phi & \phi & 1 \end{bmatrix} \quad Ai = \begin{bmatrix} c\theta_i & -cd_i s\theta_i & sd_i s\theta_i & d_i c\theta_i \\ s\theta_i & cd_i c\theta_i & -sd_i c\theta_i & d_i s\theta_i \\ \phi & sd_i & cd_i & r_i \\ \phi & \phi & \phi & 1 \end{bmatrix}$$

Find: $q = q_1, q_2, q_3, \ldots, q_N$ $\qquad$ (joint coordinates)

Solution To Inverse Kinematics

$$\begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ \phi & \phi & \phi & 1 \end{bmatrix} = A_1 A_2 A_3 \ldots A_N$$

12 Equations → 6 independent, 6 redundant

N unknowns

LHS(i,j) = RHS(i,j)

row: i = 1, 2, 3

column: j = 1, 2, 3, 4

Solution To Inverse Kinematics

General Approach: Isolate one joint variable at a time

$$\underbrace{A_1^{-1}\,{}^{0}T_N}_{\text{function of } q_1} = A_2 A_3 \ldots A_N = \underbrace{{}^{1}T_N}_{\text{function of } q_2, \ldots, q_N}$$

- Look for constant elements in ${}^{1}T_N$
- Equate LHS(i,j) = RHS(i,j)
- Solve for q_1

Solution To Inverse Kinematics

$A_2^{-1}A_1^{-1}{}^0T_N = A_3 \ldots A_N = {}^2T_N$

$\underbrace{A_2^{-1}A_1^{-1}{}^0T_N}_{\text{function of } q_1,\, q_2}$ $\qquad$ $\underbrace{{}^2T_N}_{\text{function of } q_3, \ldots, q_N}$

function of q_3, … , q_N

function of q_1, q_2

↳ only one unknown q_2 since q_1 has been solved for

- Look for constant elements of ${}^2T_N$
- Equate LHS(i,j) = RHS(i,j)
- Solve for q_2
- Maybe can find equation involving q_1 only

Note:

– There is no algorithmic approach that is 100% effective
– Geometric intuition is required

Solution To Inverse Kinematics

There are Two Classes of Robot Geometries for which closed-form inverse kinematic solutions are guaranteed. They are:

1. Robots with any 3 joints TRANSLATIONAL

2. Robots with any 3 rotational joint axes co-intersecting at a common point

These are DECOUPLED ROBOT GEOMETRIES

↙ meaning

- can reduce system to a lower order subsystem (i.e. 3rd-order) for which closed form solutions are guaranteed

General Analytical Inverse Kinematic Formula

Case 1: $\sin\theta = a$, $\cos\theta = b$; $a \in [-1,1]$, $b \in [-1,1]$ → $\theta = \text{ATANZ}(a, b)$ unique

Case 2: $\sin\theta = a$, $\cos\theta = \pm\sqrt{1 - a^2}$; $a \in [-1,1]$; $\theta = \text{ATANZ}(a, \pm\sqrt{1 - a^2})$

2 solutions

θ, $180^o - \theta$

@ $\theta = \pm 90^o$, $|a| = 1$,

"boundary" → singularity

$\cos\theta = b$, $\sin\theta = \pm\sqrt{1 - b^2}$; $b \in [-1,1]$; $\theta = \text{ATANZ}(\pm\sqrt{1 - b^2}, b)$

2 solutions

θ, $-\theta$

@ $\theta = 0^o, 180^o$, $|b| = 1$,

degeneracy of order 2 — "boundary" → singularity

General Analytical Inverse Kinematic Formula

Case 3: $a\cos\theta + b\sin\theta = 0$ → $\theta = \text{ATANZ}(a, -b)$ or $\text{ATANZ}(-a, b)$

2 solutions, 180^o apart

Singularity when $a = b = 0$

→ infinite order degeneracy

Case 4: $a\cos\theta + b\sin\theta = c$; $a, b, c \neq 0$; 2 solutions

$$\theta = \text{ATANZ}(b, a) + \text{ATANZ}(\pm\underbrace{\sqrt{a^2 + b^2 + c^2}}_{\geq 0 \text{ For solution to exist}}, c)$$

$a^2 + b^2 + c^2 < 0$ → outside workspace

$a^2 + b^2 + c^2 = 0$ → 1 solution (singularity)

degeneracy of order 2

General Analytical Inverse Kinematic Formula

Case 5: $\sin\theta\sin\phi = a$
$\cos\theta\sin\phi = b$

θ = ATANZ(a, b) if $\sin\phi$ is ⊕ positive
θ = ATANZ(-a, -b) if $\sin\phi$ is ⊖ negative

If $\cos\phi = c \rightarrow \phi$ = ATANZ($\pm\sqrt{a^2+b^2}$, c) (2 solutions for ϕ)
Then 2 solutions:

θ = ATANZ(a, b)	θ = ATANZ(-a, -b)
ϕ = ATANZ($\sqrt{a^2+b^2}$, c)	ϕ = ATANZ($-\sqrt{a^2+b^2}$, c)

Singularity: a = b = 0 $|c| = 1$
θ = undefined ϕ = 1 solution

General Analytical Inverse Kinematic Formula

Case 6: $a\cos\theta - b\sin\theta = c$ (1)
$a\cos\theta + b\sin\theta = d$ (2)

Then θ = ATANZ(ad – bc, ac + bd)
1 solution

Note that for (1) & (2) to be satisfied, or at (1) & (2), we have

$$a^2 + b^2 = c^2 + d^2$$

Decoupling (Kinematic)

"Finding a subset of joints primarily responsible for the completion of a subset of the manipulator task"

Involves the identification of:

- decoupled task ⟵ Total Task
- decoupled robot subsystem responsible for the decoupled task

Decoupled Robot Geometry – refers to a manipulator Geometry for which decoupling is guaranteed

Decoupling (Kinematic)

Decoupled Robot Geometries: (6-axes)

1. Any Three (3) Translational Joints
2. Any Three Co-Intersecting Rotational Axes

} Identified by Pieper, 1968

3. Any 2 Transl. Joints Normal to a Rot. Joint
4. Transl. Joint Normal to 2 Parallel Joints
5. Any 3 Rot, Joints Parallel

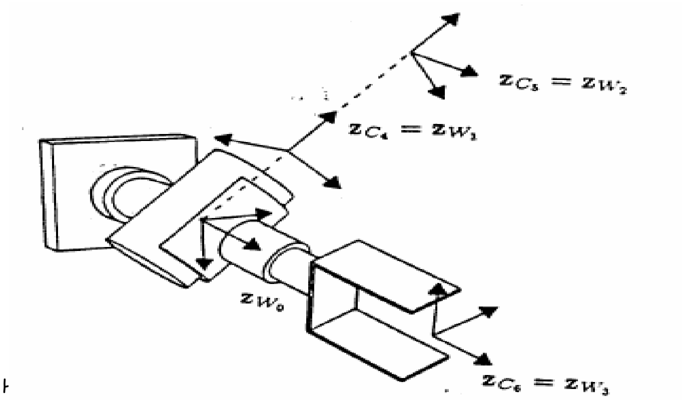
} New geometries Identified by Ang, 1992*

V.D. Tourassis and M.H. Ang Jr., "Task Decoupling in Robot Manipulators," Journal of Intelligent and Robotic Systems 14:283-302, 1995. (Technical Report in 1992).

Decoupling (Kinematic)

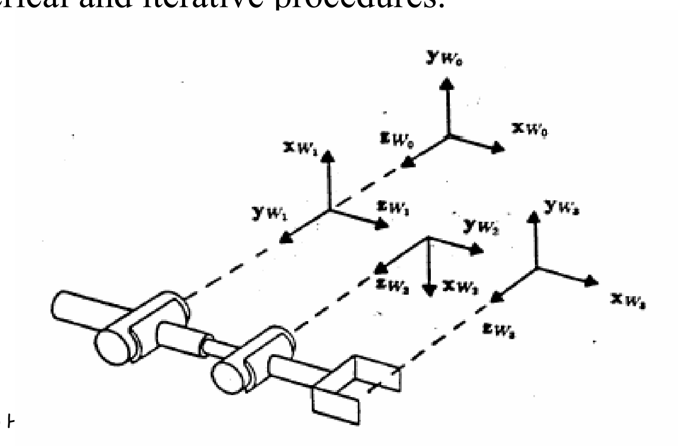
Robots with Spherical Wrists is a popular decoupled robot geometry

→ 3 wrist axes co-intersecting at a common point



Decoupling (Kinematic)

For robots that do not have decoupled geometries, a closed Form solution may not exist, → one has to resort to numerical and iterative procedures.



Numerical Solutions

Three important requirements for the numerical algorithm are:

i. a priori conditions for convergence
ii. insensitivity to initial estimates
iii. provision for multiple solutions

- The most common methods are based on the Newton-Raphson approach.

Ref: A.A.Goldenberg, B. Benhabib, & R.G.Fenton, "A complete Generalized Solution to the Immense Kinematics of Robots" IEEE Journal of Rob. & Auto. 1(1): March 1985, pp. 14-20.

Numerical Solutions

$$T_N = \prod_{i=1}^{N} A_i(q) = \begin{pmatrix} n & o & a & p \\ 0 & 0 & 0 & 1 \end{pmatrix}$$ actual positions/orientation

Let Desired position & orientation be $T_D = \begin{pmatrix} n_d & o_d & a_d & p_d \\ 0 & 0 & 0 & 1 \end{pmatrix}$

Define a residual vector, $r(q) \in \Re^6$

$$r(q) = (\underbrace{r_x\ r_y\ r_z}_{\text{position error}}\ \underbrace{r_\phi\ r_\theta\ r_\varphi}_{\text{orientation error}})^T$$

Numerical Solutions

$r_x = n \bullet (p_d - p)$
$r_y = o \bullet (p_d - p)$
$r_z = a \bullet (p_d - p)$

errors expressed in Base frame

OR define
${}^{D}T_N = (T_D)^{-1}T_N$ = position & orientation of end-effector frame with respect to origin of target frame

$\Updownarrow$

$r_x \; r_y \; r_z \;\; r_\phi \; r_\theta \; r_\varphi$

Numerical Solutions

For the residual orientation vector, one can use any suitable set of rotation angles with a predefined sequence of rotations, by transforming ${}^{D}T_N$ to a 3-parameter representation.

Example: for x-y-z rotation axes,

$r_\phi = ½(a \bullet o_d - a_d \bullet o)$
$r_\theta = ½(n \bullet a_d - n_d \bullet a)$
$r_\varphi = ½(o \bullet n_d - o_d \bullet n)$

Numerical Solutions

OR orientation error can be expressed as

$$\begin{pmatrix} r_{\phi} \\ r_{\theta} \\ r_{\varphi} \end{pmatrix} = \frac{1}{2}\left[(n \times n_d) + (o \times o_d) + (a \times a_d)\right]$$

The solution q* is obtained when $T_D = T_N$ or

$r(q^*) = 0$

Newton Raphson Solutions

$q^{(k+1)} = q^{(k)} + \delta^{(k)}$ iteration until $\delta^{(k)} \rightarrow 0$

where $\delta^{(k)}$ solves the linear system

$$r_j(q^{(k)}) + \sum_{i=1}^{N} J_{ji}^{(k)} \delta_i^{(k)} = 0 \qquad j = 1,\ldots,6$$

$$\text{where } J_{ji} \triangleq \left[\frac{\delta r_j}{\delta q_i}\right]_{q=q^{(k)}} \qquad \begin{matrix} i = 1, \ldots, N \\ j = 1, \ldots, 6 \end{matrix}$$

OR J = can be the usual 6xN manipulator Jacobian

Newton Raphson Solutions

Note that Newton Raphson takes into account only the first order terms of the Taylor Series Expansion if $r(q)$.

Problem:

Newton Raphson fails to converge when the initial estimate of $q^{(0)}$ is not sufficiently close to the solution
→ Several Modifications