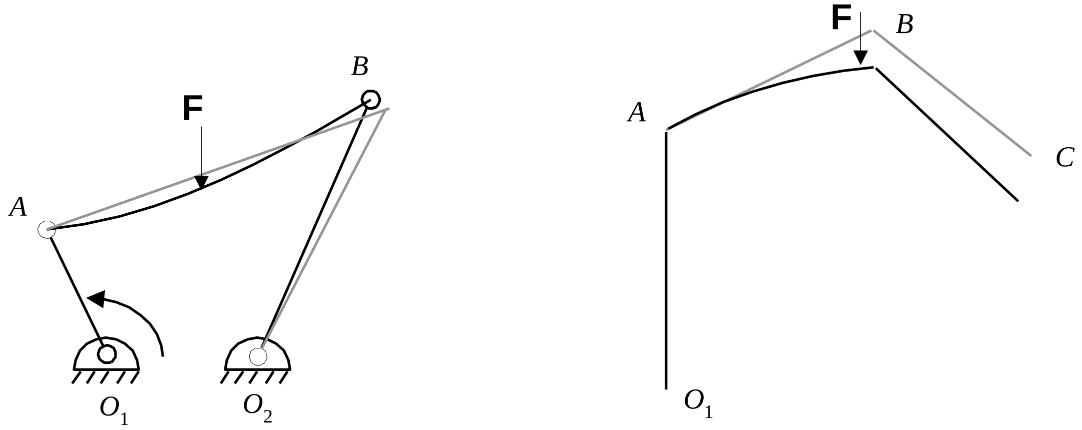
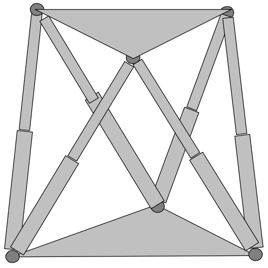
CHAPTER TWO

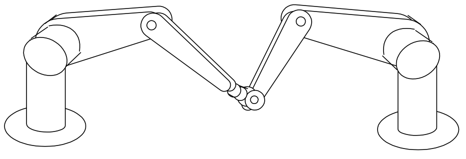
KINEMATICS OF ROBOTS: POSITION ANALYSIS

- Manipulator-type robots are multi degree-of-freedom (DOF), three dimensional, open-loop, chain mechanisms



The difference between open loop and closed-loop mechanisms



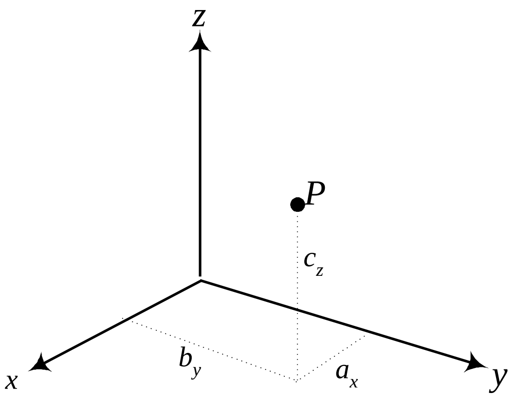


Parallel mechanisms

Representation of a Point in Space

- A point *P* in space can be represented by its three coordinates relative to a reference frame as:

$$P = a_x \mathbf{i} + b_y \mathbf{j} + c_z \mathbf{k}$$

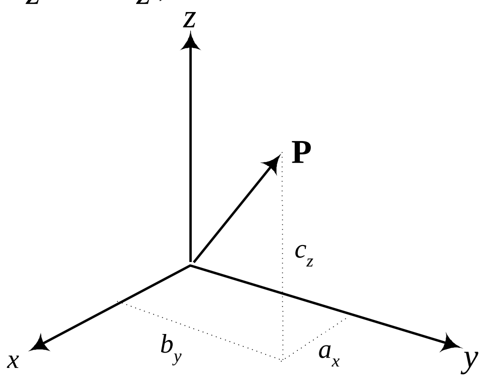


Representation of a Vector in Space

A vector can be represented by three coordinates of its tail and its head. If the vector starts at point *A* and ends at point *B*, then it can be represented by:

$$\mathbf{P}_{AB} = (B_x - A_x)\mathbf{i} + (B_y - A_y)\mathbf{j} + (B_z - A_z)\mathbf{k}$$

$$\mathbf{P} = \begin{bmatrix} a_x \\ b_y \\ c_z \end{bmatrix}$$



Application of a scale factor

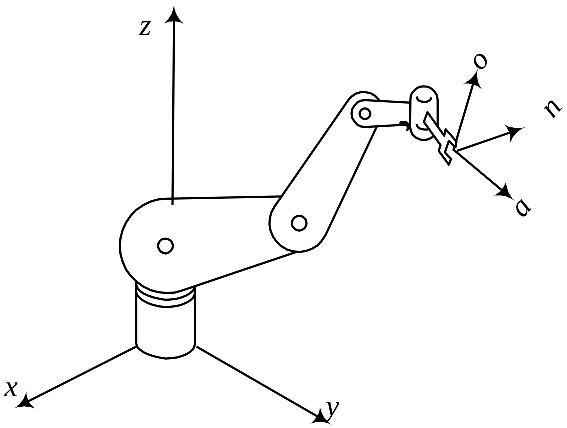
- Makes the matrix 4 by 1
- Allows for introducing directional vectors

$$\mathbf{P} = \begin{bmatrix} P_x \\ P_y \\ P_z \\ w \end{bmatrix}$$

$$a_x = \frac{P_x}{w}, \; b_y = \frac{P_y}{w}$$

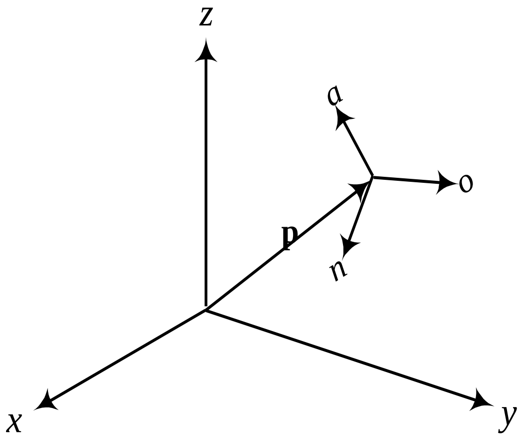
The n-o-a Frame designation

- Approach, Orientation, Normal directions



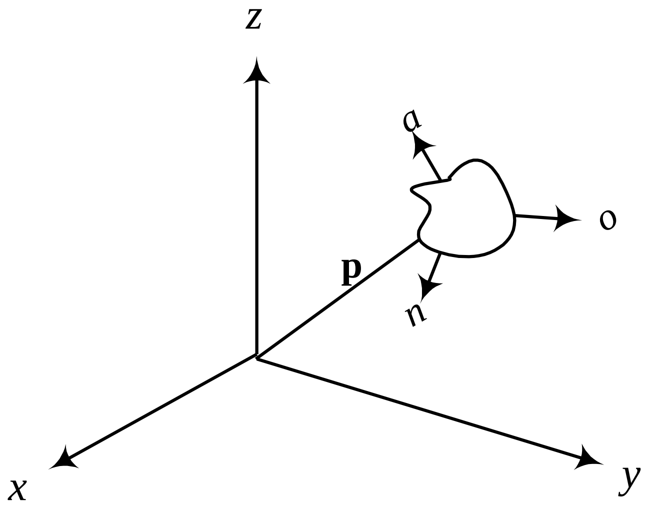
Representation of a Frame Relative to a Fixed Reference Frame

$$F = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Representation of a Rigid Body

$$F_{object} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Frame representation Requirements

- the three unit vectors **n**, **o**, **a** are mutually perpendicular
- each unit vector's length, represented by its directional cosines, must be equal to 1
- These constraints translate into the following six constraint equations:
- $\mathbf{n} \cdot \mathbf{o} = 0$ (the dot-product of **n** and **o** vectors must be zero)

  $\mathbf{n} \cdot \mathbf{a} = 0$

  $\mathbf{a} \cdot \mathbf{o} = 0$
- $|\mathbf{n}| = 1$ (the magnitude of the length of the vector must be 1)
- $|\mathbf{o}| = 1$ and $|\mathbf{a}| = 1$

- The same can be achieved by:

$$\mathbf{n} \times \mathbf{o} = \mathbf{a}$$

Homogeneous Transformation Matrices

- 4 by 4 matrices:
  - Can be pre- or post-multiplied
  - Easy to find inverse of the matrix
  - Represents both orientation and position information, including directional vectors

$$F = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Representation of Transformations

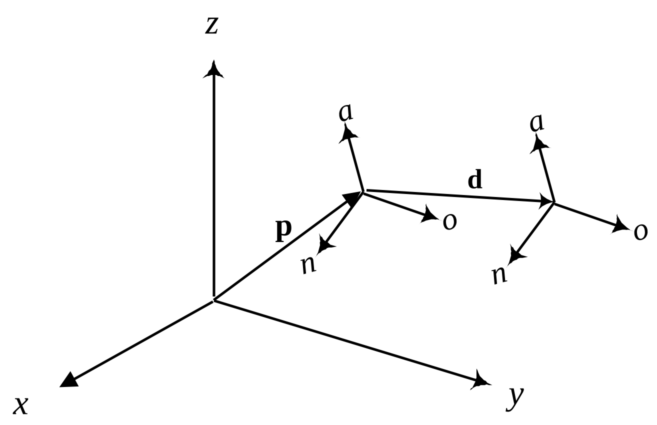
A transformation may be in one of the following forms:

- A pure translation
- A pure rotation about an axis
- A combination of translations and/or rotations

Representation of a Pure Translation

$$F_{new} = Trans\,(d_x, d_y, d_z) \times F_{old}$$

$$F_{new} = \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n_x & o_x & a_x & p_x + d_x \\ n_y & o_y & a_y & p_y + d_y \\ n_z & o_z & a_z & p_z + d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Representation of a Pure Rotation about an Axis

x,y,z → n, o, a

Projections as seen from x axis

- **Assumption**: The frame is at the origin of the reference frame and parallel to it.

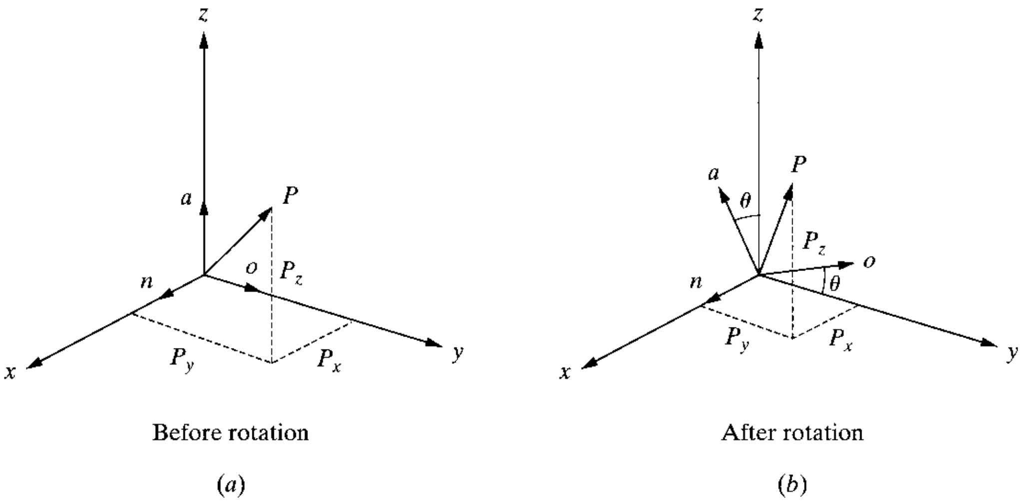


Fig. 2.10 Coordinates of a point in a rotating frame before and after rotation around axis x.

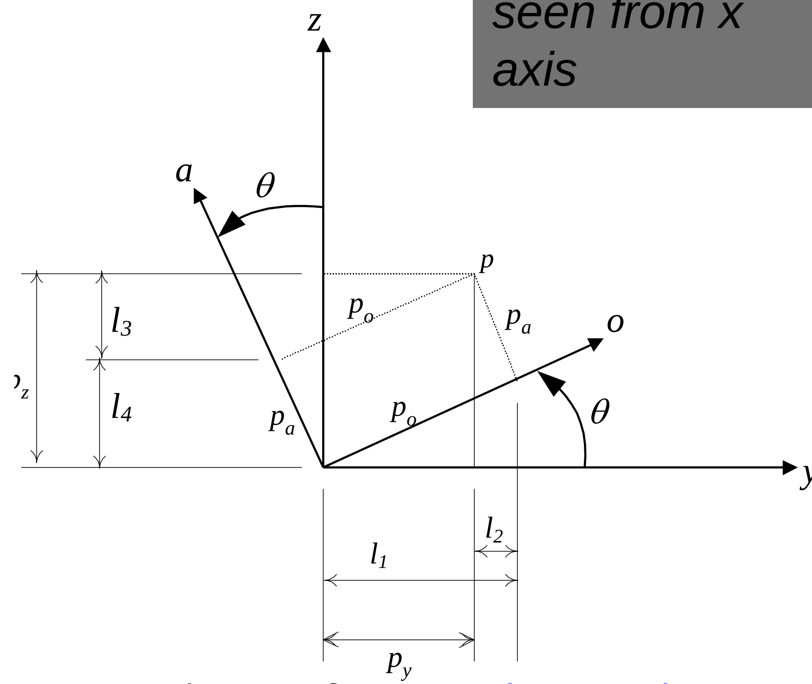


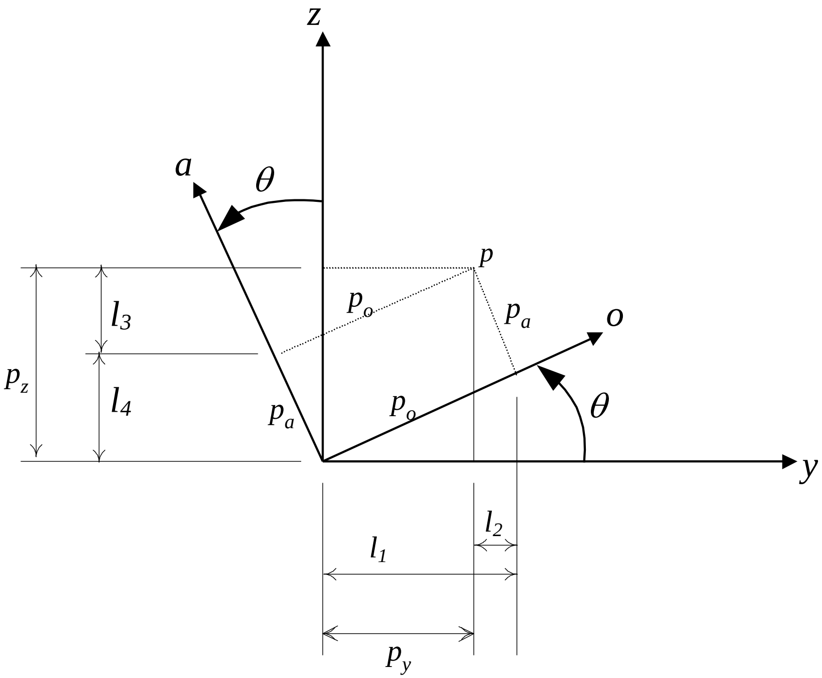
Fig. 2.11 Coordinates of a point relative to the reference frame and rotating frame as viewed from the *x*-axis.

Representation of a Pure Rotation about an Axis

$$p_x = p_n$$

$$p_y = l_1 - l_2 = p_o \cos\theta - p_a \sin\theta$$

$$p_z = l_3 + l_4 = p_o \sin\theta + p_a \cos\theta$$



$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} p_n \\ p_o \\ p_a \end{bmatrix}$$

$$p_{xyz} = Rot(x,\theta) \times p_{noa}$$

Rotation Matrices

$$Rot(x,\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\theta & -S\theta \\ 0 & S\theta & C\theta \end{bmatrix}$$

$$Rot(y,\theta) = \begin{bmatrix} C\theta & 0 & S\theta \\ 0 & 1 & 0 \\ -S\theta & 0 & C\theta \end{bmatrix}$$

$$Rot(z,\theta) = \begin{bmatrix} C\theta & -S\theta & 0 \\ S\theta & C\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Representation of Combined Transformations

- Example:

1. Rotation of α degrees about the *x*-axis,
2. Followed by a translation of $[l_1,l_2,l_3]$ (relative to the *x*-, *y*-, and *z*-axes respectively),
3. Followed by a rotation of β degrees about the *y*-axis.

- Pre-multiply by each matrix:

$$p_{1,xyz} = Rot(x,\alpha)\times p_{noa}$$

$$p_{2,xyz} = Trans(l_1,l_2,l_3)\times p_{1,xyz} = Trans(l_1,l_2,l_3)\times Rot(x,\alpha)\times p_{noa}$$

$$p_{xyz} = p_{3,xyz} = Rot(y,\beta)\times p_{2,xyz} = Rot(y,\beta)\times Trans(l_1,l_2,l_3)\times Rot(x,\alpha)\times p_{noa}$$

Representation of Combined Transformations

- **Example:** A point P $(7,3,2)^T$ is attached to a frame (n,o,a) and is subjected to the transformations described next. Find the coordinates of the point relative to the reference frame at the end of transformations.

1. Rotation of 90 degrees about the *z*-axis,
2. Followed by a Rotation of 90 degrees about the *y*-axis
3. Followed by a translation of [4,-3,7]

Representation of Combined Transformations

- Example: A point P $(7,3,2)^T$ is attached to a frame (n,o,a) and is subjected to the transformations described next. Find the coordinates of the point relative to the reference frame at the end of transformations.

1. Rotation of 90 degrees about the *z*-axis,
2. Followed by a Rotation of 90 degrees about the *y*-axis
3. Followed by a translation of [4,-3,7]

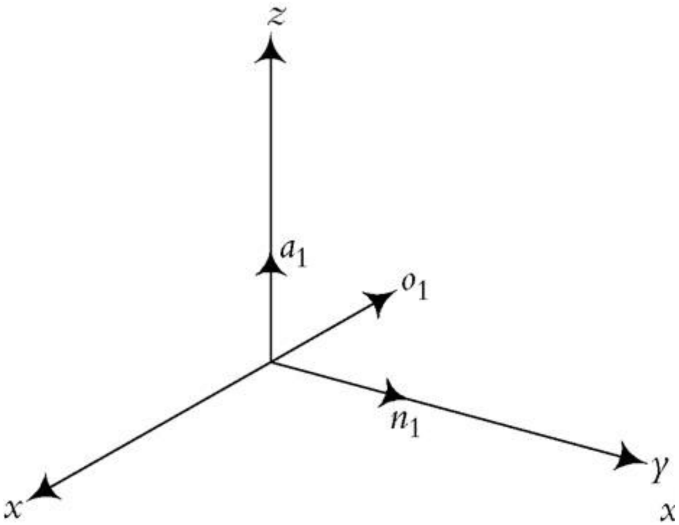
- Pre-multiply by each matrix:

$$P_{xyz} = Trans_{(4,-3,7)} Rot_{(y,90)} Rot_{(z,90)} P_{noa}$$

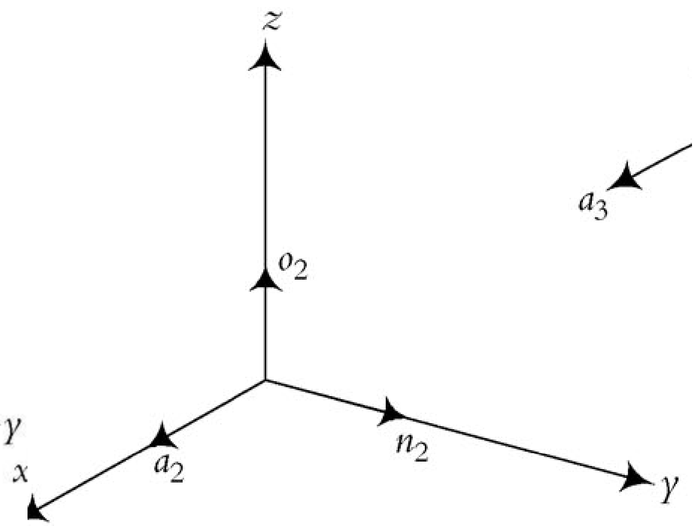
Representation of Combined Transformations

- Pre-multiply by each matrix:

$$P_{xyz} = Trans_{(4,-3,7)} Rot_{(y,90)} Rot_{(z,90)} P_{noa}$$



After the 1st transformation

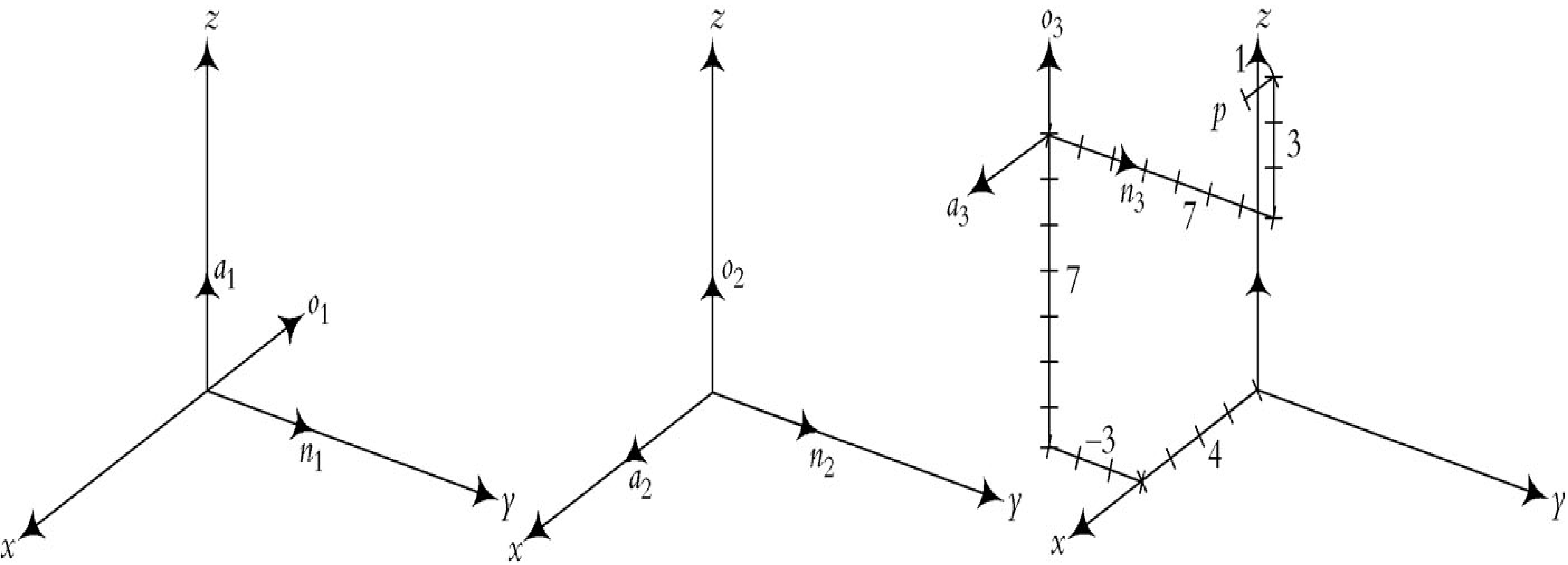


After the 2nd transformation

Representation of Combined Transformations

- Pre-multiply by each matrix:

$$P_{xyz} = Trans_{(4,-3,7)} Rot_{(y,90)} Rot_{(z,90)} P_{noa}$$



After the 1st transformation After the 2nd transformation After the 3rd transformation

Representation of Combined Transformations

- Example: A point P $(7,3,2)^T$ is attached to a frame (n,o,a) and is subjected to the transformations described next. Find the coordinates of the point relative to the reference frame at the end of transformations.

1. Rotation of 90 degrees about the *z*-axis,
2. Followed by a translation of [4,-3,7]
3. Followed by a Rotation of 90 degrees about the *y*-axis

Representation of Combined Transformations

- **Example:** A point P $(7,3,2)^T$ is attached to a frame (n,o,a) and is subjected to the transformations described next. Find the coordinates of the point relative to the reference frame at the end of transformations.

1. Rotation of 90 degrees about the *z*-axis,
2. Followed by a translation of [4,-3,7]
3. Followed by a Rotation of 90 degrees about the *y*-axis

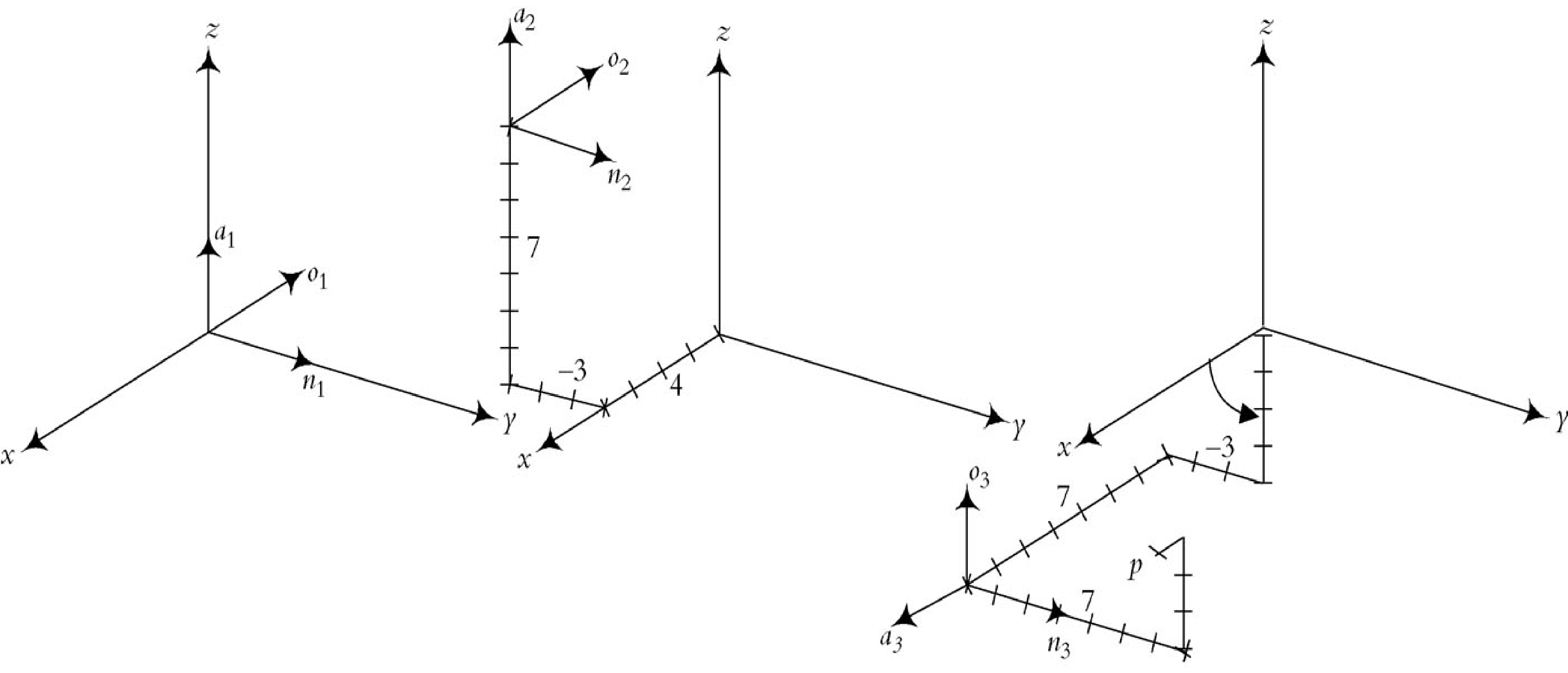
- Pre-multiply by each matrix:

$$P_{xyz} = Rot_{(y,90)} Trans_{(4,-3,7)} Rot_{(z,90)} P_{noa}$$

Representation of Combined Transformations

- Pre-multiply by each matrix:

$$P_{xyz} = Rot_{(y,90)} Trans_{(4,-3,7)} Rot_{(z,90)} P_{noa}$$



After the 1st transformation

After the 2nd transformation

After the 3rd transformation

Transformations Relative to the Rotating (current) Frame

- In this case, matrices representing each transformation are post-multiplied.
- If transformations are relative to both the Universe frame and the current frame, each matrix is accordingly multiplied, either pre- or post-.

Representation of Combined Transformations

- Example: A point P $(7,3,2)^T$ is attached to a frame (n,o,a) and is subjected to the transformations described next. Find the coordinates of the point relative to the moving frame at the end of transformations.

1. Rotation of 90 degrees about the *a*-axis,
2. Followed by a translation of [4,-3,7] along n,o,a
3. Followed by a Rotation of 90 degrees about the *o*-axis

Representation of Combined Transformations

- **Example:** A point P $(7,3,2)^T$ is attached to a frame (n,o,a) and is subjected to the transformations described next. Find the coordinates of the point relative to the moving frame at the end of transformations.

1. Rotation of 90 degrees about the *a*-axis,
2. Followed by a translation of [4,-3,7] along n,o,a
3. Followed by a Rotation of 90 degrees about the *o*-axis

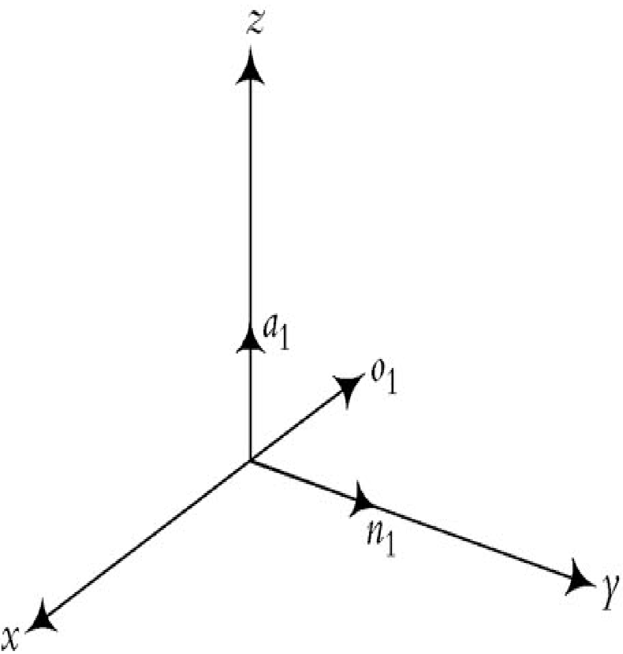
- Post-multiplying each matrix:

$$P_{xyz} = Rot_{(a,90)} Trans_{(4,-3,7)} Rot_{(o,90)} P_{noa}$$

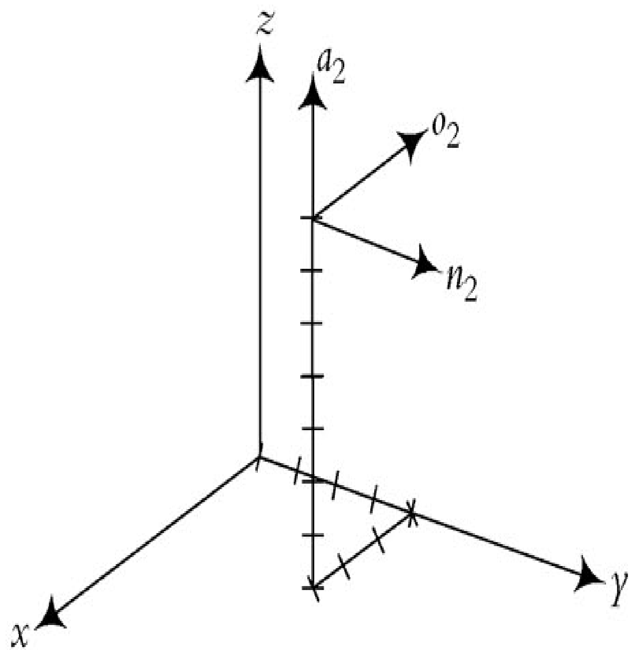
Representation of Combined Transformations

- Pre-multiplying each matrix:

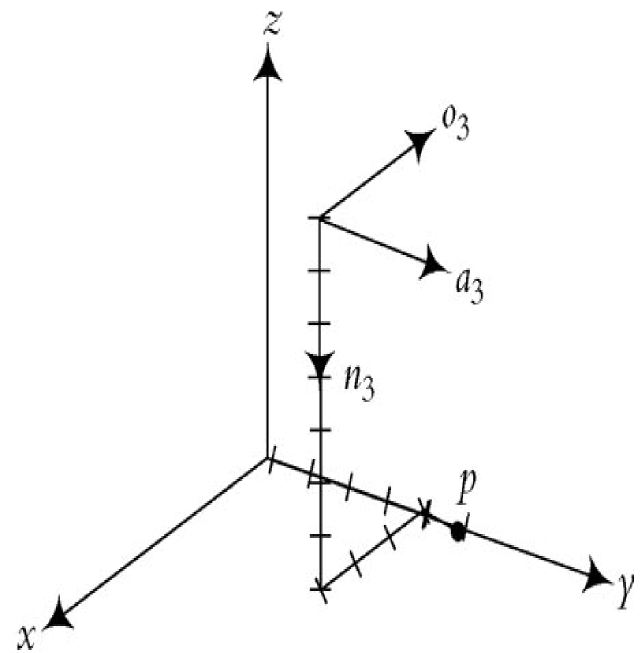
$$P_{xyz} = Rot_{(a,90)} Trans_{(4,-3,7)} Rot_{(o,90)} P_{noa}$$



After the 1st transformation



After the 2nd transformation



After the 3rd transformation

Representation of Combined Transformations

- Example: A frame B was rotated about the x axis by 90^o; it was then translated about the current a-axis by 3 inches before being rotated about z-axis by 90^0, Finally it was translated about current o-axis by 5 inches.

1. Write an equation describing the motion
2. Find the final location of a point P (1,5,4) attached to the frame relative to the reference frames.

Representation of Combined Transformations

- **Example:** A frame B was rotated about the x axis by 90°; it was then translated about the current a-axis by 3 inches before being rotated about z-axis by 90^0, Finally it was translated about current o-axis by 5 inches.

1. Write an equation describing the motion
2. Find the final location of a point P (1,5,4) attached to the frame relative to the reference frames.

- Pre or Post-multiplying each matrix we get:

$$T = Rot_{(z,90)}\, Rot_{(x,90)} Trans_{(0,0,3)} Trans_{(0,5,0)}$$

Inverse of Matrices

- The following steps must be taken to calculate the inverse of a matrix:
  - Calculate the determinant of the matrix.
  - Transpose the matrix.
  - Replace each element of the transposed matrix by its own minor (adjoint matrix).
  - Divide the converted matrix by the determinant.

Inverse of Rotation Matrices

- The inverse of a rotation matrix is its transpose because rotation matrices are “unitary”.

Inverse of Transformation Matrices

- The inverse of a transformation (or a frame) matrix is the following:

$$T = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad T^{-1} = \begin{bmatrix} n_x & n_y & n_z & -\mathbf{p}\cdot\mathbf{n} \\ o_x & o_y & o_z & -\mathbf{p}\cdot\mathbf{o} \\ a_x & a_y & a_z & -\mathbf{p}\cdot\mathbf{a} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

  - 1. Transpose the rotation portion of the matrix.
  - 2. Take the negative of the dot-product of the P and n, P and o, and P and a vectors.
  - The scale factors remain the same.

Forward and Inverse Kinematic Equations

- Forward kinematics includes substituting the known joint values into the equations to find the location and orientation
- Inverse kinematics includes finding an equation that results in joint values if the desired position and orientation are specified.

Forward and Inverse Kinematics for Positioning

- Four possibilities are common:

  a. Cartesian (gantry, rectangular) coordinates

  b. Cylindrical coordinates

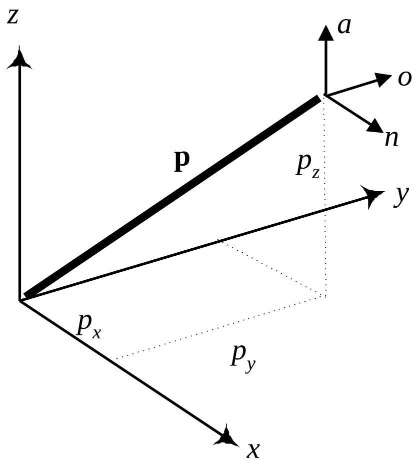
  c. Spherical coordinates

  d. Articulated (anthropomorphic or all-revolute) coordinates

Cartesian Coordinates

- Three linear motions.

$$^{R}T_{p} = T_{cart}(p_x, p_y, p_z) = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

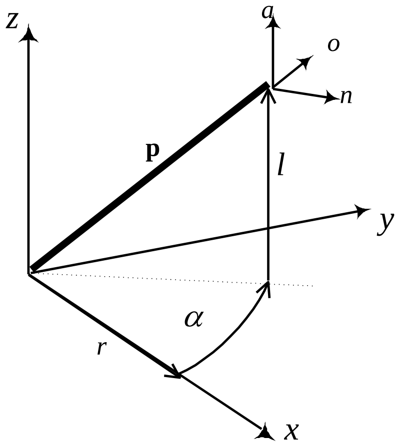


Cylindrical Coordinates

- Two linear and one revolute joints

$$ {}^{R}T_{p} = T_{cyl}(r,\alpha,l) = Trans(0,0,l)Rot(z,\alpha)Trans(r,0,0) $$

$$ {}^{R}T_{p} = T_{cyl}(r,\alpha,l) = \begin{bmatrix} C\alpha & -S\alpha & 0 & rC\alpha \\ S\alpha & C\alpha & 0 & rS\alpha \\ 0 & 0 & 1 & l \\ 0 & 0 & 0 & 1 \end{bmatrix} $$



Inverse Solution

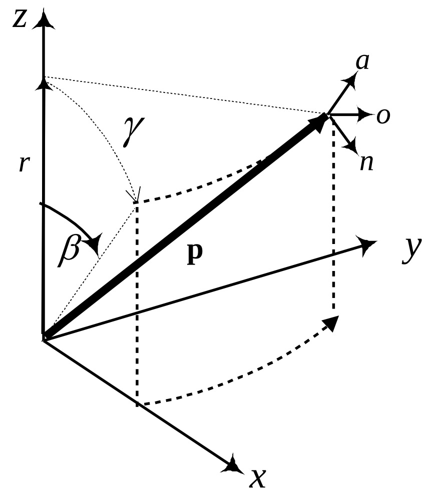
- Use the position equations to find the joint values.
- The application of ATAN2 function for correct determination of angles.

Spherical Coordinates

- Two revolute and one linear joints

$$ {}^{R}T_{P} = T_{sph}(r,\beta,\gamma) = Rot(z,\gamma)Rot(y,\beta)Trans(0,0,r) $$

$$ {}^{R}T_{P} = T_{sph}(r,\beta,\gamma) = \begin{bmatrix} C\beta C\gamma & -S\gamma & S\beta C\gamma & rS\beta C\gamma \\ C\beta S\gamma & C\gamma & S\beta S\gamma & rS\beta S\gamma \\ -S\beta & 0 & C\beta & rC\beta \\ 0 & 0 & 0 & 1 \end{bmatrix} $$



Inverse Solution

- Use the position equations to determine the joint values.
- Check your answers for correct values.

Articulated Coordinates

- We will study later with the Denavit–Hartenberg methodology

Forward and Inverse Kinematics for Orientation

- Three possibilities are common:

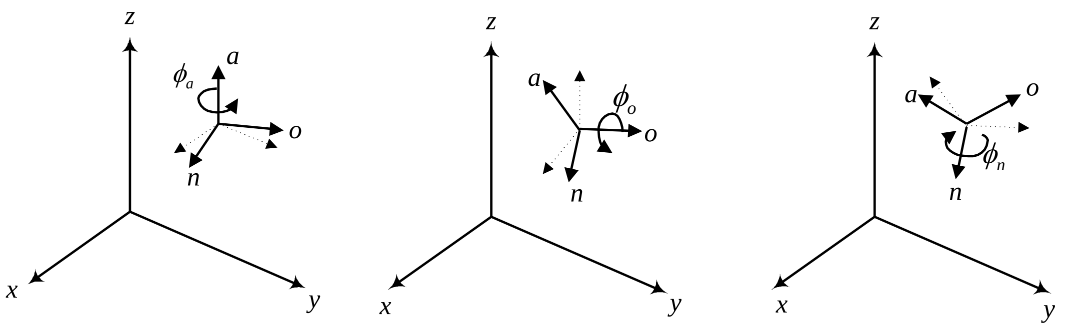
  a. Roll-pitch-yaw (RPY) angles

  b. Euler angles

  c. Articulated coordinates

RPY Angles

- Rotations relative to the current z-, y-, and x-axes



$$\mathrm{RPY}(\phi_a,\phi_o,\phi_n) = Rot(a,\phi_a)Rot(o,\phi_o)Rot(n,\phi_n) =$$

$$\begin{bmatrix} C\phi_a C\phi_o & C\phi_a S\phi_o S\phi_n - S\phi_a C\phi_n & C\phi_a S\phi_o C\phi_n + S\phi_a S\phi_n & 0 \\ S\phi_a C\phi_o & S\phi_a S\phi_o S\phi_n + C\phi_a C\phi_n & S\phi_a S\phi_o C\phi_n - C\phi_a S\phi_n & 0 \\ -S\phi_o & C\phi_o S\phi_n & C\phi_o C\phi_n & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Inverse Solution

- Use:

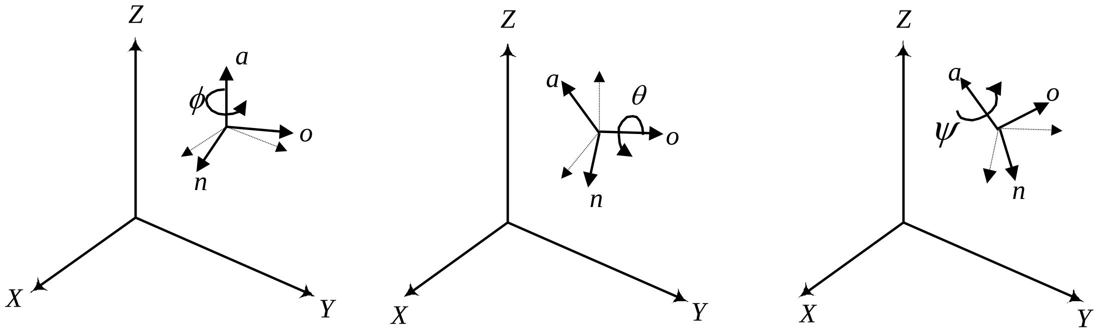
$$\phi_a = ATAN2(n_y, n_x) \text{ and } \phi_a = ATAN2(-n_y, -n_x)$$

$$\phi_o = ATAN2[-n_z, (n_x C\phi_a + n_y S\phi_a)]$$

$$\phi_n = ATAN2[(-a_y C\phi_a + a_x S\phi_a), (o_y C\phi_a - o_x S\phi_a)]$$

Euler Angles

- Rotations relative to the current z-, y-, and z-axes.



$$\mathrm{Euler}(\phi,\theta,\psi)=Rot(a,\phi)Rot(o,\theta),Rot(a,\psi)=$$

$$\begin{bmatrix} C\phi C\theta C\psi - S\phi S\psi & -C\phi C\theta S\psi - S\phi C\psi & C\phi S\theta & 0 \\ S\phi C\theta C\psi + C\phi S\psi & -S\phi C\theta S\psi + C\phi C\psi & S\phi S\theta & 0 \\ -S\theta C\psi & S\theta S\psi & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Inverse Solution

- Use:

$$\phi = ATAN2(a_y, a_x) \quad or \quad \phi = ATAN2(-a_y, -a_x)$$

$$\psi = ATAN2[(-n_x S\phi + n_y C\phi), (-o_x S\phi + o_y C\phi)]$$

$$\theta = ATAN2[(a_x C\phi + a_y S\phi), a_z)]$$

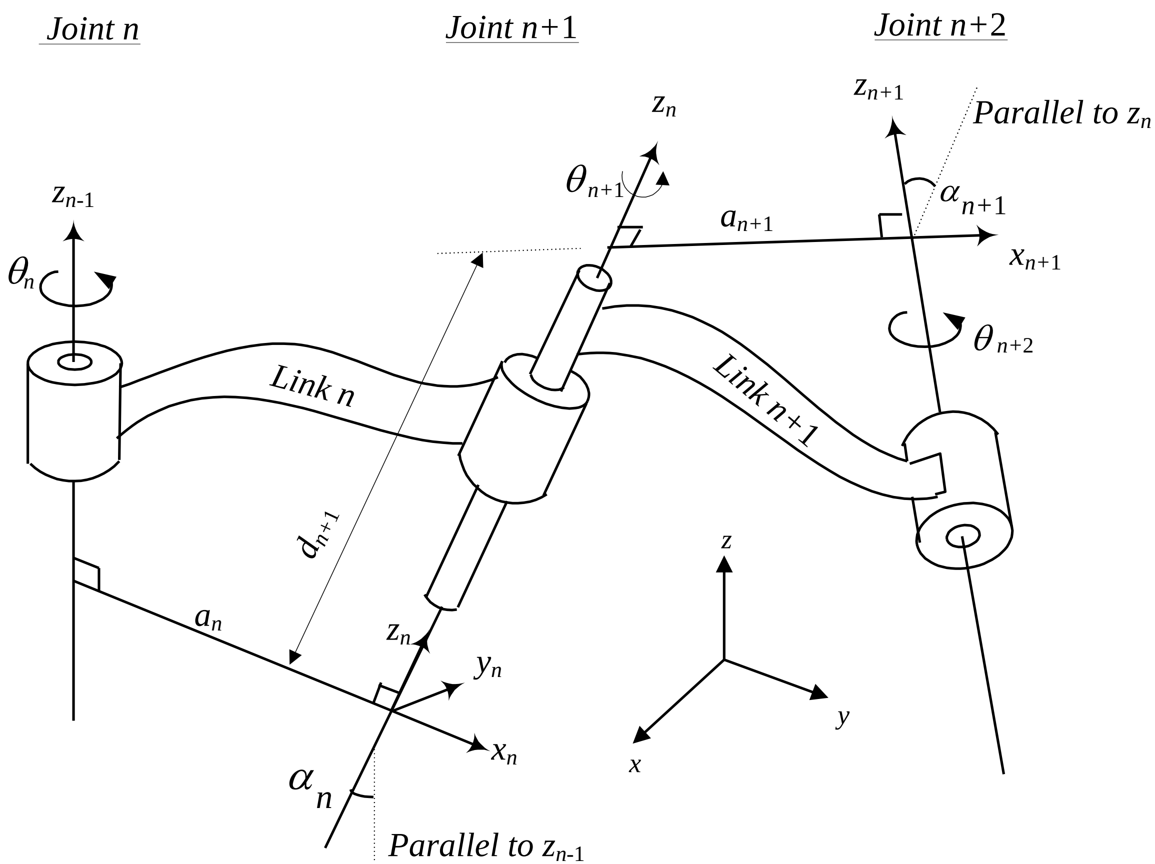
Articulated Angles

- We will study later with the Denavit–Hartenberg methodology

Denavit-Hartenberg (DH) Representation of Forward Kinematic Equations of Robots

- May be used for any configuration, whether specific coordinates or not.
- Can include joint offset, twist angles, multi-variable joints, and so on.
- Very common.
- Many other equations are based on this methodology

D-H Representation



(a)

D-H Representation

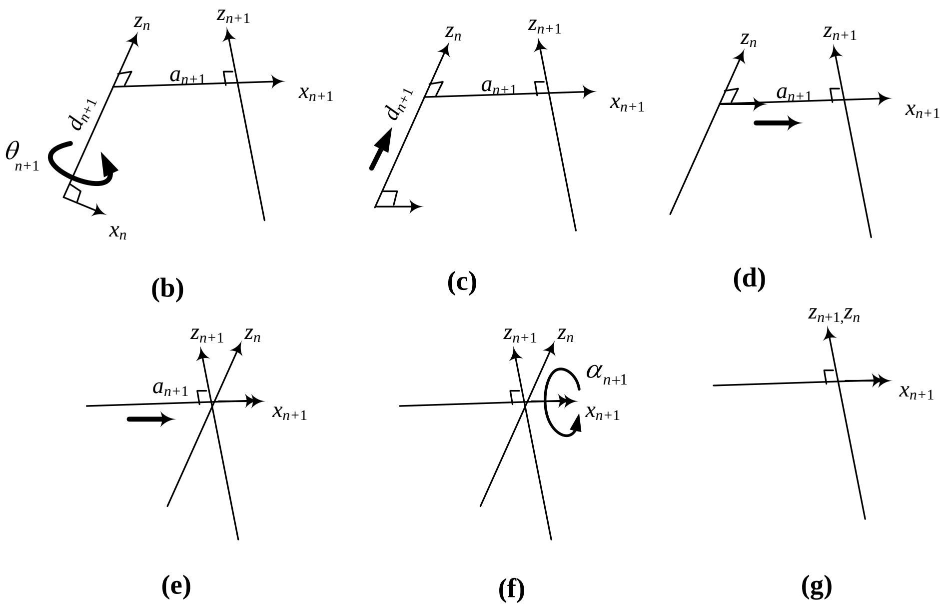
- Z-axes along the joint motion. θ *represents joint rotation.*
- *D is joint linear displacement or distance between common normals.*
- *α is the twist angle between z-axes.*
- *a is the length of the common normal.*

D-H Representation

- Assign z-axes to each joint along linear motion or revolute axis.
- Assign x-axes along the common normal between successive z-axes.
- No need for y-axes.
- If z-axes coincide, x-axis is perpendicular to both.
- If z-axes are parallel, x-axes can be anywhere.

D-H Representation

- Four transformations are necessary to go from one frame to the next:



D-H Representation

- Rotate about the z_n-axis an angle of θ_{n+1}. This will make x_n and x_{n+1} parallel to each other. This is true because a_n and a_{n+1} are both perpendicular to z_n, and rotating z_n an angle of θ_{n+1} will make them parallel (and thus, coplanar).
- Translate along the z_n-axis a distance of d_{n+1} to make x_n and x_{n+1} colinear. Since x_n and x_{n+1} were already parallel and normal to z_n, moving along z_n will lay them over each other.
- Translate along the (already rotated) x_n-axis a distance of a_{n+1} to bring the origins of x_n and $x_{n+}1$ together. At this point, the origins of the two reference frames will be at the same location.
- Rotate z_n-axis about x_{n+1}-axis an angle of $\alpha_{n+1}$to align z_n-axis with z_{n+1}-axis. At this point, frames n and n+1 will be exactly the same, and we have transformed from one to the next.

D-H Representation

- A transformation matrix can be formed by:

$$ {}^{n}T_{n+1} = A_{n+1} = Rot(z,\theta_{n+1}) \times Trans(0,0,d_{n+1}) \times Trans(a_{n+1},0,0) \times Rot(x,\alpha_{n+1}) $$

$$ = \begin{bmatrix} C\theta_{n+1} & -S\theta_{n+1} & 0 & 0 \\ S\theta_{n+1} & C\theta_{n+1} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_{n+1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & a_{n+1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha_{n+1} & -S\alpha_{n+1} & 0 \\ 0 & S\alpha_{n+1} & C\alpha_{n+1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

D-H Representation

- An A-matrix is:

$$A_{n+1} = \begin{bmatrix} C\theta_{n+1} & -S\theta_{n+1}C\alpha_{n+1} & S\theta_{n+1}S\alpha_{n+1} & a_{n+1}C\theta_{n+1} \\ S\theta_{n+1} & C\theta_{n+1}C\alpha_{n+1} & -C\theta_{n+1}S\alpha_{n+1} & a_{n+1}S\theta_{n+1} \\ 0 & S\alpha_{n+1} & C\alpha_{n+1} & d_{n+1} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

D-H Representation

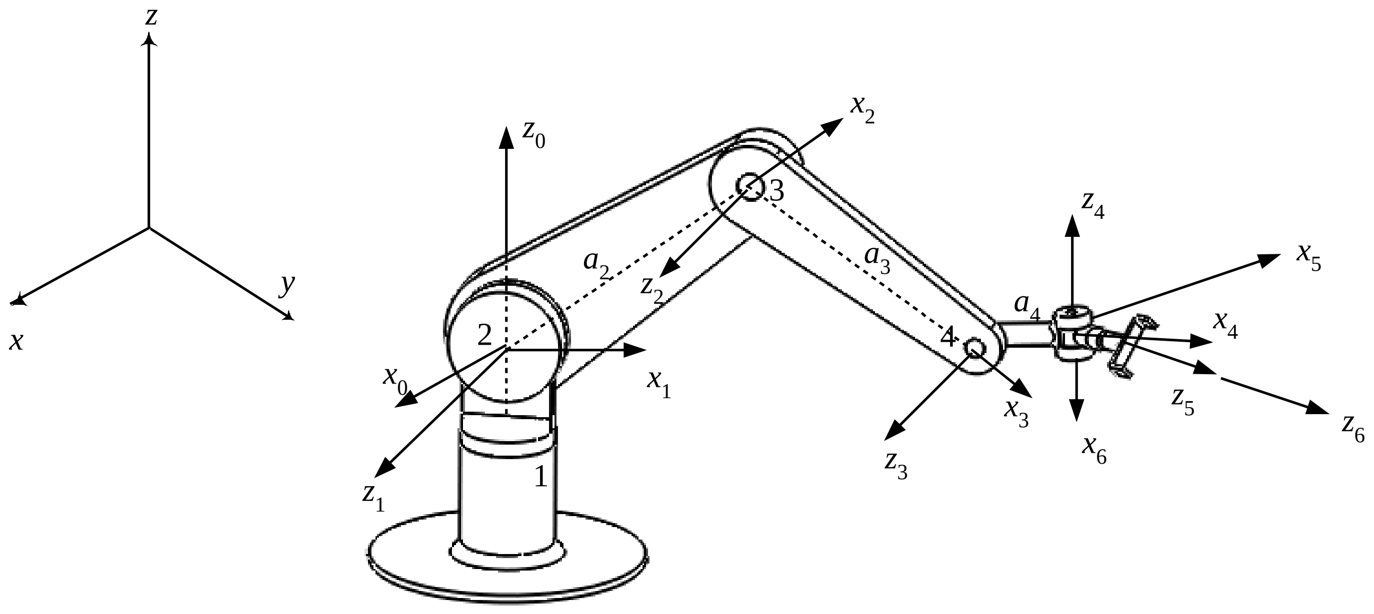
- A parameters table may look like:

Table 2.1: D-H Parameters Table

#	θ	d	a	α
0-1				
1-2				
2-3				
3-4				
4-5				
5-6				

D-H Representation

- Example: A simple 6-axis robot



Inverse Kinematic Equations

- Find a set of equations that allow determination of joint values from desired position and orientation information.
- Each robot has a different solution.
- It may be necessary to use different approaches for each robot.
- This usually requires pre-multiplication of transformation matrices by inverse of individual *A* matrices, squaring of terms, divisions, and so on.

Inverse Kinematic Equations

- For the shown example, the following may be found:

$$\theta_1 = \tan^{-1}\left(\frac{p_y}{p_x}\right) \quad and \quad \theta_1 = \theta_1 + 180°$$

$$C_3 = \frac{(p_x C_1 + p_y S_1 - C_{234} a_4)^2 + (p_z - S_{234} a_4)^2 - a_2^2 - a_3^2}{2 a_2 a_3}$$

where $S_3 = \pm\sqrt{1 - C_3^2}$ and $\theta_3 = \tan^{-1}\dfrac{S_3}{C_3}$

Inverse Kinematic Equations: cont.

$$\theta_{234} = \tan^{-1}\left(\frac{a_z}{C_1 a_x + S_1 a_y}\right) \text{ and } \theta_{234} = \theta_{234} + 180°$$

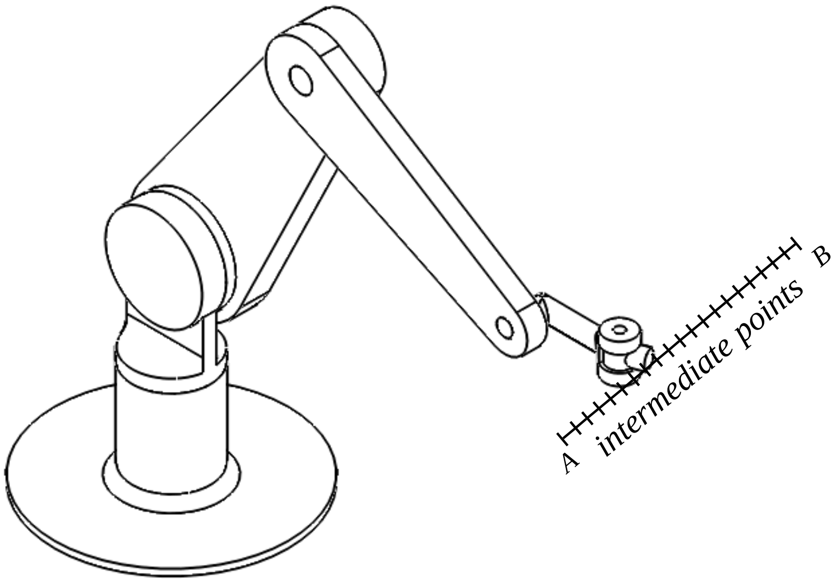
$$\theta_2 = \tan^{-1}\frac{(C_3 a_3 + a_2)(p_z - S_{234} a_4) - S_3 a_3 (p_x C_1 + p_y S_1 - C_{234} a_4)}{(C_3 a_3 + a_2)(p_x C_1 + p_y S_1 - C_{234} a_4) + S_3 a_3 (p_z - S_{234} a_4)}$$

$$\theta_4 = \theta_{234} - \theta_2 - \theta_3$$

$$\theta_5 = \tan^{-1}\frac{C_{234}(C_1 a_x + S_1 a_y) + S_{234} a_z}{S_1 a_x - C_1 a_y}$$

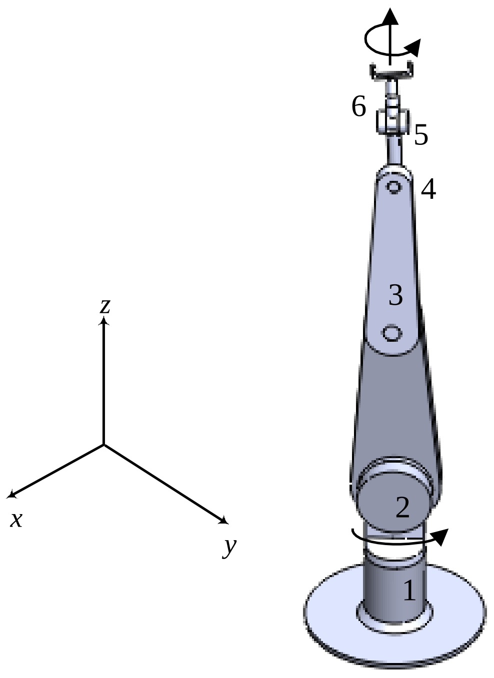
$$\theta_6 = \tan^{-1}\frac{-S_{234}(C_1 n_x + S_1 n_y) + C_{234} n_z}{-S_{234}(C_1 o_x + S_1 o_y) + C_{234} o_z}$$

Inverse Kinematic Programming of Robots



Degeneracy

- When a degree of freedom is lost.



Dexterity

- When a position within the work envelope can be specified, but orientation is limited. This usually happens near the boundaries or reach.

The Fundamental Problem with the Denavit-Hartenberg Representation

- No transformation along the *y*-axis is allowed.
- Joint manufacturing errors are usually in this direction.