Forward Kinematics

EE366/CE366/CS380: Introduction to Robotics

Dr. Basit Memon

Electrical and Computer Engineering
Habib University

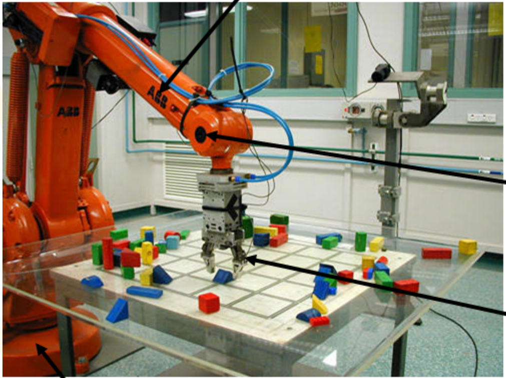
January 31, February, 7, 2024

Table of Contents

Table of Contents

General Terminology

Link: Rigid body. 6 DOF



Joint: Connects two links

End Effector: Interacts with environments

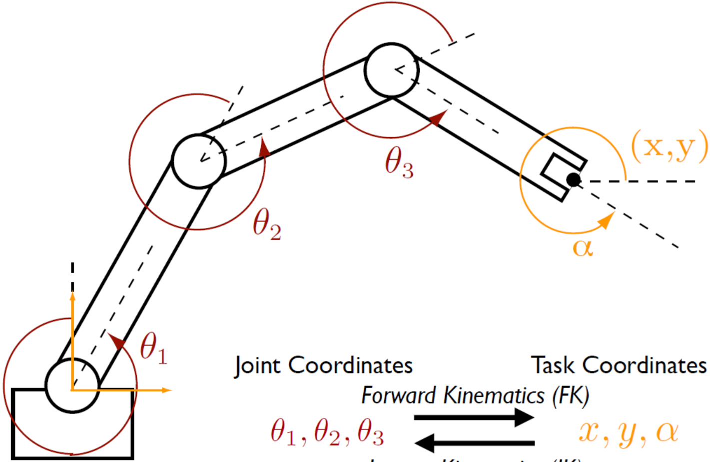
Base: Connected to ground.

Kinematics: Study of motion (position, velocity, acceleration, etc.) without regard to the forces, torques that cause it. Geometric description of motion.

Forward Kinematics: Calculation of position and orientation of end-effector from its joint coordinates.

Inverse Kinematics: Determine the values of the joint coordinates, given the end-effector's position and orientation.

Kinematics establishes link between joint and task variables.



Joint Coordinates — Task Coordinates

Forward Kinematics (FK)

$\theta_1, \theta_2, \theta_3 \rightleftarrows x, y, \alpha$

Inverse Kinematics (IK)

Given a robotic manipulator, what is the kinematic mapping?

- Joint Coordinates: q
- Task Coordinates: r
- Forward kinematic mapping: f such that $r = f(q)$
- Inverse kinematic mapping: f such that $q = f^{-1}(r)$
- How to find f?

Kinematic mapping f is a homogeneous transformation.

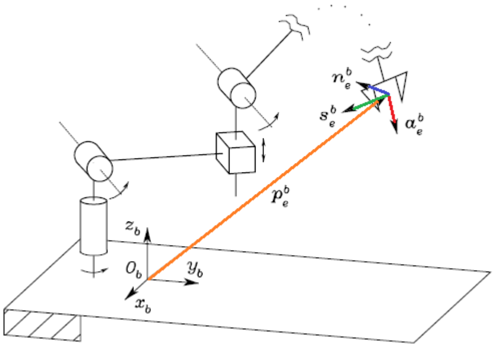


Figure: Source: Robotics-Modeling, Planning and Control

$$
\begin{aligned}
f(q) &= {}^bT_e(q) \\
&= \begin{bmatrix} {}^bn_e(q) & {}^bs_e(q) & {}^ba_e(q) & {}^bp_e(q) \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}
$$

- How do you find this transform?

Table of Contents

1. Find f by inspection.

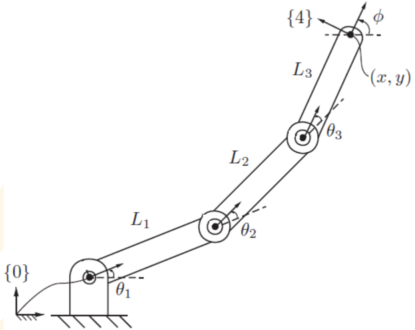


Figure: Source: Modern Robotics

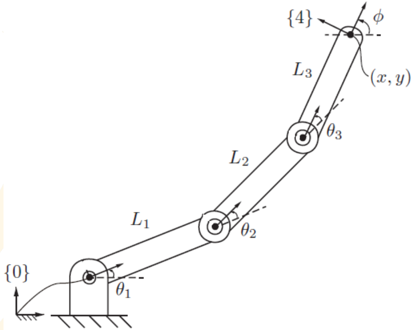


Figure: Source: Modern Robotics

- $x =$
  $L_1 \cos\theta_1 + L_2 \cos(\theta_1 + \theta_2) + L_3 \cos(\theta_1 + \theta_2 + \theta_3)$
- $y =$
  $L_1 \sin\theta_1 + L_2 \sin(\theta_1 + \theta_2) + L_3 \sin(\theta_1 + \theta_2 + \theta_3)$
- $\phi = \theta_1 + \theta_2 + \theta_3$

Transformation is not always obvious by inspection.

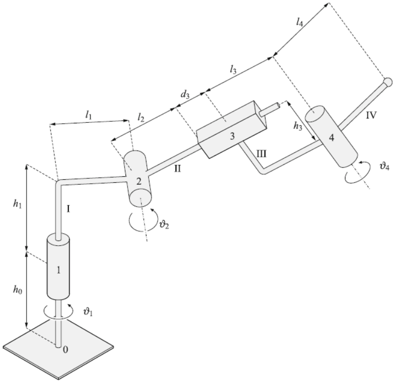


Figure: Source: Robotics-Bajd

- Tedious for more links and 3D geometries.

Assign frames and write homogeneous transformation sequentially.

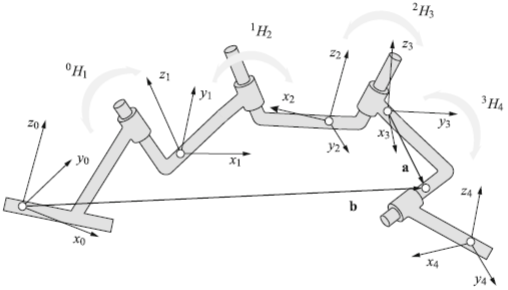


Figure: Source: Robotics-Bajd

- Attach reference frames to each link.

Assign frames and write homogeneous transformation sequentially.

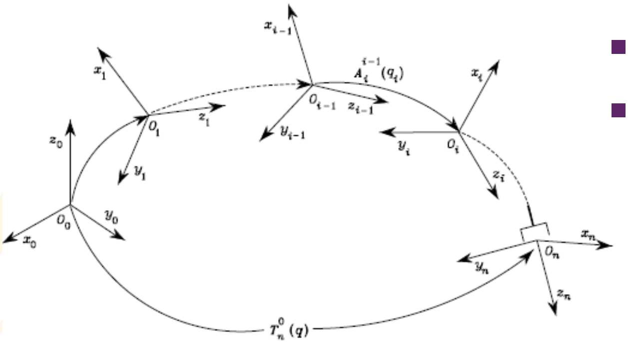


Figure: Source: Robotics-Modeling, Planning and Control

- Attach reference frames to each link.
- Find required homogeneous transformation as a compounded transformation.

$$
\begin{aligned}
{}^{b}T_{e} &= {}^{0}T_{n} \\
&= {}^{0}T_{1}\,{}^{1}T_{2}\cdots{}^{i-1}T_{i}\cdots{}^{n-1}T_{n}.
\end{aligned}
$$

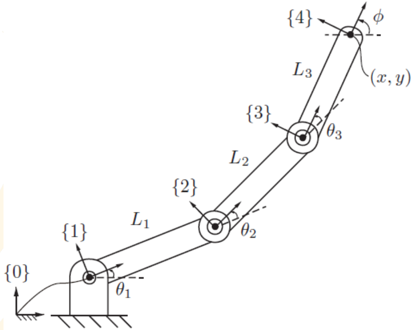


Figure: Source: Modern Robotics

- $${}^0T_1 = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
- $${}^1T_2 = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & L_1 \\ \sin\theta_2 & \cos\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

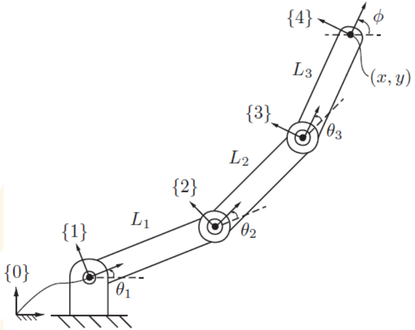


Figure: Source: Modern Robotics

- $${}^{2}_{3}T = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & L_2 \\ \sin\theta_3 & \cos\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
- $${}^{3}_{4}T = \begin{bmatrix} 1 & 0 & 0 & L_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
- MATLAB Live Script – FK3R

1. Find each ${}^{k}T_{k+1}$.
2. Find global transformation

$$ {}^{0}T_{n} = {}^{0}T_{1}\,{}^{1}T_{2}\cdots{}^{n-1}T_{n} $$

Computationally, to avoid unnecessary multiplications and additions:

$$ {}^{0}T_{n} = \begin{bmatrix} {}^{0}R_{n} & {}^{0}p_{n} \\ \mathbf{0} & 1 \end{bmatrix} $$

$$ {}^{0}R_{n} = {}^{0}R_{1}\,{}^{1}R_{2}\cdots{}^{n-1}R_{n} $$

$$ {}^{i}p_{j} = {}^{i}p_{(j-1)} + {}^{i}R_{j-1}\,{}^{j-1}p_{j} $$

${}^{0}p_{n}$ can be obtained by solving the above recursively.

1. Find each ${}^{k}T_{k+1}$.
2. Find global transformation

$$ {}^{0}T_{n} = {}^{0}T_{1}\,{}^{1}T_{2}\cdots{}^{n-1}T_{n} $$

- Step 1 is dependent on the frame assignment.
- Process is streamlined by using convention for frame assignment, specifically Denavit-Hartenberg convention.
  - This is not the only convention.

Table of Contents

Joints in Denavit-Hartenberg (DH) Convention [1]

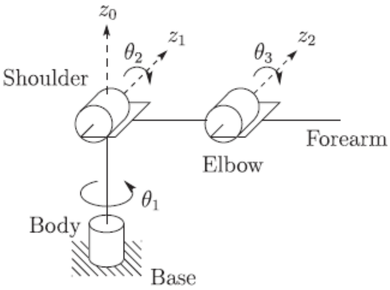


Figure: Source: Robot Modeling and Control

- Robot manipulator: Set of rigid links connected by joints.
- Joint is either revolute or prismatic, i.e. 1 dof joint.
- What about other joints?
  - Modeled as a sequence of single dof joints with zero length links in between.

Denavit-Hartenberg (DH) Naming Convention [1]

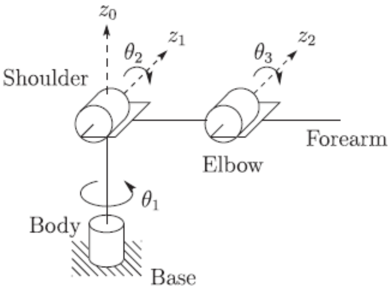


Figure: Source: Robot Modeling and Control

- Robot manipulator with n joints has $n+1$ links.
- Links numbered from 0 to n. Link 0 is base.
- Joints numbered from 1 to n.
- Joint i connects link $i-1$ to link i.
- Location of joint i is fixed wrt link $i-1$. When joint i moves, link i moves.

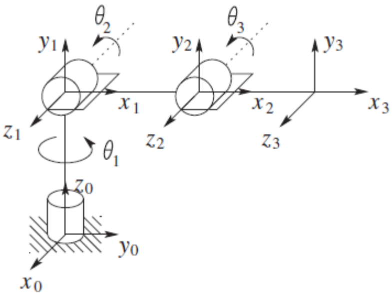


Figure: Source: Robot Modeling and Control

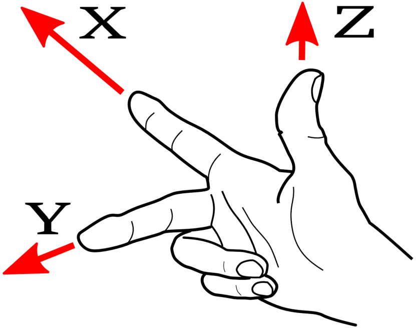
- Attach frame i rigidly to link i. Joint i moves, frame i moves.
- From frame assignment, find DH parameters, (a, α, d, θ), and compute A_i:

$$A_i = \begin{bmatrix} c\theta_i & -s\theta_i\, c\alpha_i & s\theta_i\, s\alpha_i & a_i\, c\theta_i \\ s\theta_i & c\theta_i\, c\alpha_i & -c\theta_i\, s\alpha_i & a_i\, s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Homogeneous transformation $A_i = {}^{i-1}T_i$ represents motion of frame i with respect to $i-1$.

Table of Contents

Assignment of base frame



- z_0 is along axis of rotation or translation of Joint 1.
- Choice of x_0 and y_0 is arbitrary, as long as right-hand rule is satisfied.

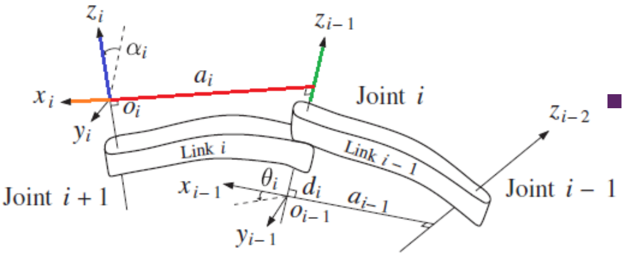


Figure: source: Robot Modeling and Control

- Choose axis z_i along the axis of joint $i+1$. For revolute, this is axis of rotation; for prismatic, the axis of translation.
- The origin O_i is at the intersection of axis z_i with the common normal between axes z_{i-1} and z_i.
  - Normal: **Shortest** line segment connecting z_{i-1} and z_i, and **perpendicular** to both.
  - If z_i and z_{i-1} are not coplanar, it is unique.

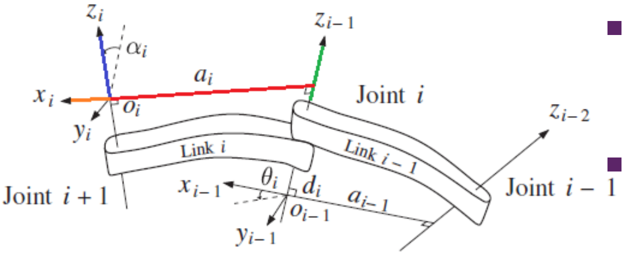


Figure: source: Robot Modeling and Control

- Axis x_i is along the direction of the common normal, pointing from Joint i to Joint $i+1$.
- Choose axis y_i, according to right-hand rule.
- Repeat the process iteratively.

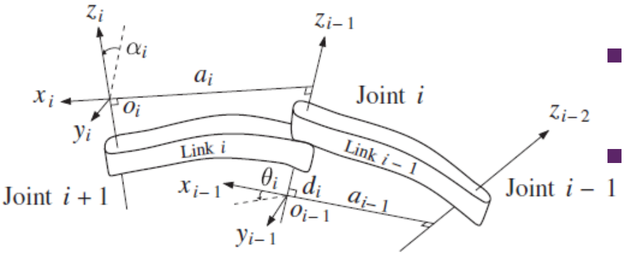


Figure: source: Robot Modeling and Control

- **Link Length (a_i)**: Distance between axes z_{i-1} and z_i and is measured along the axis x_i
- **Link Twist (α_i)**: Angle from z_{i-1} to z_i measured about x_i
- **Link Offset (d_i)**: Distance from the origin o_{i-1} to the intersection of x_i with z_{i-1}, measured along z_{i-1}
- **Joint Angle (θ_i)**: Angle from x_{i-1} to x_i measured about z_{i-1}

`https://youtu.be/rA9tm0gTln8`

Special Case-1: z_{i-1} is parallel to z_i

- Infinitely many common normals between z_{i-1} and z_i.
- Freedom to choose O_i
- Typically x_i chosen as normal passing through O_{i-1}.
- $\alpha_i = 0$.

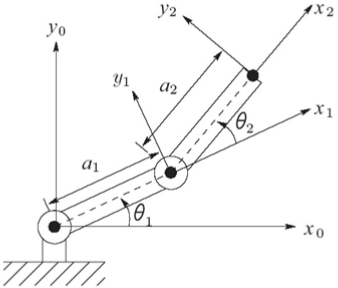


Figure: Source: Robot Modeling and Control

- Choose z_0 (axis of rotation of joint 1, base frame)
- x_0 and y_0 are chosen as shown.
- z_1 is axis of rotation of joint 2.
- z_0 and z_1 are parallel.
  - Choose x_1 as the perpendicular through O_0.
- End-effector frame is arbitrary.

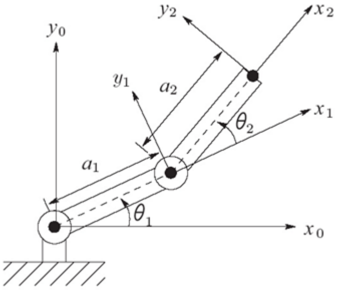


Figure: Source: Robot Modeling and Control

- DH Parameters:

Link	a_i	α_i	d_i	θ_i
1	a_1	0	0	θ_1
2	a_2	0	0	θ_2

-

$$A_i = \begin{bmatrix} c\theta_i & -s\theta_i \, c\alpha_i & s\theta_i \, s\alpha_i & a_i \, c\theta_i \\ s\theta_i & c\theta_i \, c\alpha_i & -c\theta_i \, s\alpha_i & a_i \, s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Link	a_i	α_i	d_i	θ_i
1	a_1	0	0	θ_1
2	a_2	0	0	θ_2

$$A_i = \begin{bmatrix} c\theta_i & -s\theta_i\, c\alpha_i & s\theta_i\, s\alpha_i & a_i\, c\theta_i \\ s\theta_i & c\theta_i\, c\alpha_i & -c\theta_i\, s\alpha_i & a_i\, s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_1 = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & a_1\, c\theta_1 \\ s\theta_1 & c\theta_1 & 0 & a_1\, s\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & a_2\, c\theta_2 \\ s\theta_2 & c\theta_2 & 0 & a_2\, s\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Example-1: Two link planar manipulator

$$A_1 = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & a_1\, c\theta_1 \\ s\theta_1 & c\theta_1 & 0 & a_1\, s\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & a_2\, c\theta_2 \\ s\theta_2 & c\theta_2 & 0 & a_2\, s\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned} {}^0_2T &= A_1\, A_2 \\ &= \begin{bmatrix} c_{12} & -s_{12} & 0 & a_1 c_1 + a_2 c_{12} \\ s_{12} & c_{12} & 0 & a_1 s_1 + a_2 s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

Frame n – End-effector or Tool Frame

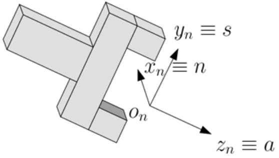


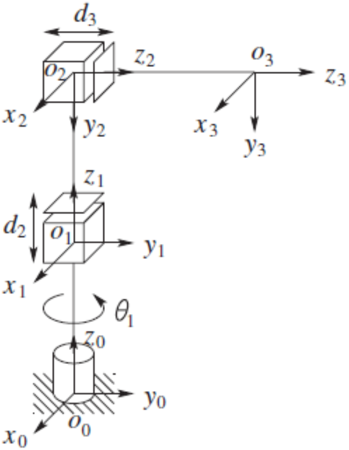
Figure: Source: Robot Modeling and Control

- Freedom of choice.
- O_n is often placed symmetrically between fingers.
- z_n is typically labeled a and called "approach" direction.
- y_n is labeled s and is "sliding" direction.
- x_n is labeled n and is "normal" direction, normal to the plane formed by a and s.

Special Case-2: z_{i-1} intersects z_i

- x_i is chosen normal to plane formed by z_i and z_{i-1}.
- Positive direction of x_i is arbitrary.
- Origin O_i is at intersection of z_i and z_{i-1}.
- $a_i = 0$.

Example-2: Three-Link Cylindrical Robot



- Choose z_0 (axis of rotation of joint 1, base frame)
- x_0 and y_0 are chosen as shown.
- z_1 is axis of translation of joint 2.
- z_0 and z_1 intersect.
  - O_1 is chosen at joint 1.
  - x_1 is orthogonal to both z_0 and z_1

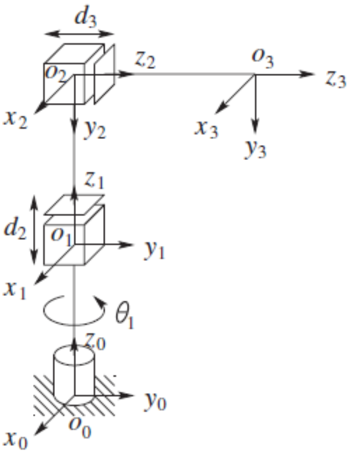


Figure: Source: Robot Modeling and Control

- z_2 is axis of translation of joint 3.
- z_1 and z_2 intersect.
  - O_2 is chosen at point of intersection.
  - x_2 is orthogonal to plane formed by z_1 and z_2
- Tool frame is chosen at end of link 3.

Example-2: Three-Link Cylindrical Robot

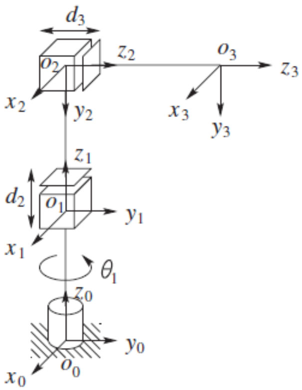


Figure: Source: Robot Modeling and Control

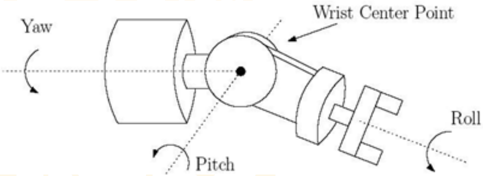
- DH Parameters:

Link	a_i	α_i	d_i	θ_i
1	0	0	L_1	θ_1
2	0	$-90°$	d_2	0
3	0	0	d_3	0

-

$$
{}^0_3T = A_1\,A_2\,A_3
$$

$$
= \begin{bmatrix} c_1 & 0 & -s_1 & -s_1\,d_3 \\ s_1 & 0 & c_1 & c_1\,d_3 \\ 0 & -1 & 0 & L_1 + d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$



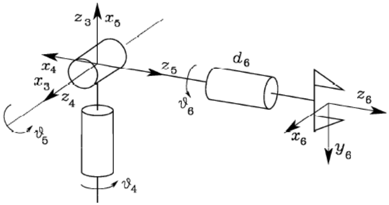


Figure: Source: Robotics–Modeling, Planning and Control

Link	a_i	α_i	d_i	θ_i
1	0	$-90°$	0	θ_4
2	0	$90°$	0	θ_5
3	0	0	d_6	θ_6

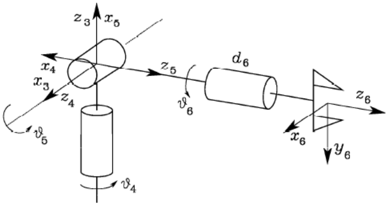


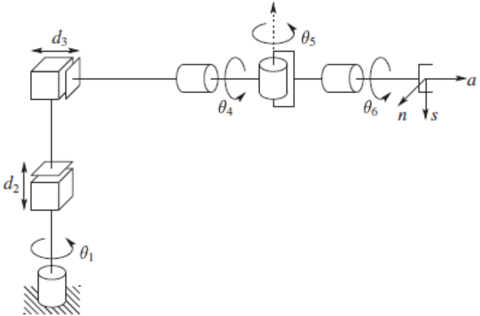
Figure: Source: Robotics–Modeling, Planning and Control

Link	a_i	α_i	d_i	θ_i
1	0	$-90°$	0	θ_4
2	0	$90°$	0	θ_5
3	0	0	d_6	θ_6

$$
{}^{3}_{6}T = \begin{bmatrix} c_4c_5c_6 - s_4s_6 & -c_4c_5s_6 - s_4c_6 & c_4s_5 & c_4s_5d_6 \\ s_4c_5c_6 + c_4s_6 & -s_4c_5s_6 + c_4c_6 & s_4s_5 & s_4s_5d_6 \\ -s_5c_6 & s_5s_6 & c_5 & c_5d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

- Rotation matrix is same as ZYZ Euler angles rotation, wrt $\{3\}$ frame.

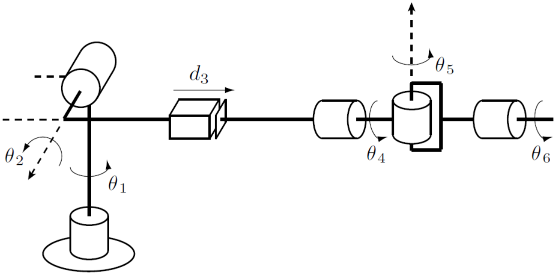
Example-4: Cylindrical Arm with Spherical Wrist



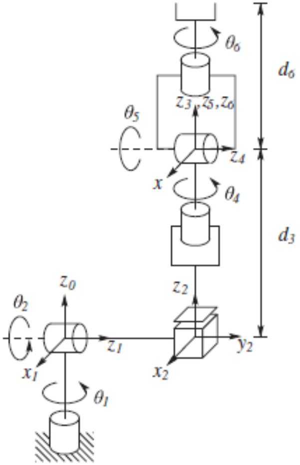
Link	a_i	α_i	d_i	θ_i
1	0	0	L_1	θ_1
2	0	$-90°$	d_2	0
3	0	0	d_3	0
4	0	$-90°$	0	θ_4
5	0	$90°$	0	θ_5
6	0	0	d_6	θ_6

- How many dof?
- 6: θ_1, d_2, d_3, θ_4, θ_5, θ_6

Example-5: Stanford Arm



Example-5: Stanford Arm



Link	a_i	α_i	d_i	θ_i
1	0	$-90°$	0	θ_1
2	0	$90°$	d_2	θ_2
3	0	0	d_3	0
4	0	$-90°$	0	θ_4
5	0	$90°$	0	θ_5
6	0	0	d_6	θ_6

Example-5: Stanford Arm

$$A_1 = \begin{bmatrix} c_1 & 0 & -s_1 & 0 \\ s_1 & 0 & c_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_2 = \begin{bmatrix} c_2 & 0 & s_2 & 0 \\ s_2 & 0 & -c_2 & 0 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} c_4 & 0 & -s_4 & 0 \\ s_4 & 0 & c_4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_5 = \begin{bmatrix} c_5 & 0 & s_5 & 0 \\ s_5 & 0 & -c_5 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_6 = \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Example-5: Stanford Arm

$$
\begin{aligned}
r_{11} &= c_1\left[c_2\left(c_4c_5c_6 - s_4s_6\right) - s_2s_5c_6\right] - d_2\left(s_4c_5c_6 + c_4s_6\right)\\
r_{21} &= s_1\left[c_2\left(c_4c_5c_6 - s_4s_6\right) - s_2s_5c_6\right] + c_1\left(s_4c_5c_6 + c_4s_6\right)\\
r_{31} &= -s_2\left(c_4c_5c_6 - s_4s_6\right) - c_2s_5c_6\\
r_{12} &= c_1\left[-c_2\left(c_4c_5s_6 + s_4c_6\right) + s_2s_5s_6\right] - s_1\left(-s_4c_5s_6 + c_4c_6\right)\\
r_{22} &= -s_1\left[-c_2\left(c_4c_5s_6 - s_4c_6\right) - s_2s_5s_6\right] + c_1\left(-s_4c_5s_6 + c_4s_6\right)\\
r_{32} &= s_2\left(c_4c_5s_6 + s_4c_6\right) + c_2s_5s_6\\
r_{13} &= c_1\left(c_2c_4s_5 + s_2c_5\right) - s_1s_4s_5\\
r_{23} &= s_1\left(c_2c_4s_5 + s_2c_5\right) + c_1s_4s_5\\
r_{33} &= -s_2c_4s_5 + c_2c_5\\
d_x &= c_1s_2d_3 - s_1d_2 + d_6\left(c_1c_2c_4s_5 + c_1c_5s_2 - s_1s_4s_5\right)\\
d_y &= s_1s_2d_3 + c_1d_2 + d_6\left(c_1s_4s_5 + c_2c_4s_1s_5 + c_5s_1s_2\right)\\
d_z &= c_2d_3 + d_6\left(c_2c_5 - c_4s_2s_5\right)
\end{aligned}
$$

Table of Contents

Claim: Each homogeneous transformation A_i in a serial chain can be represented as a product of four basic transformations:

$$A_i = {}^{i-1}T_i = R_z(\theta_i)\,\text{Trans}_z(d_i)\,\text{Trans}_x(a_i)\,R_x(\alpha_i).$$

- Why do we care?
- We're representing transformation with four numbers $(\theta_i, d_i, a_i, \alpha_i)$ instead of six.
- Out of the four, three are constant called joint parameters and one is joint variable. For revolute joint, variable is θ and for prismatic, it is d.

$$q_i = \begin{cases} \theta_i & \text{if joint } i \text{ is revolute,} \\ d_i & \text{if joint } i \text{ is prismatic.} \end{cases}$$

DH Convention: What's the catch?

- Should we be able to represent a homogeneous transformation with just 4 numbers?
- The coordinate axes of the frames attached to the links have to be defined in the prescribed manner.
- The origin of a frame, in some case, would not even be physically on the link.

Constraints on the assignment of frames

For two frames $\{0\}$ and $\{1\}$, the homogeneous transformation, A, that takes coordinates from frame $\{1\}$ to those of frame $\{0\}$ can be represented as

$$A = {}^{0}T_1 = R_z(\theta)\,\mathrm{Trans}_z(d)\,\mathrm{Trans}_x(a)\,R_x(\alpha),$$

such that a, d, θ, α are unique as long as the following conditions are satisfied:

- (**DH1**) The axis x_1 is perpendicular to the axis z_0;
- (**DH2**) The axis x_1 intersects the axis z_0.

Constraints on the assignment of frames

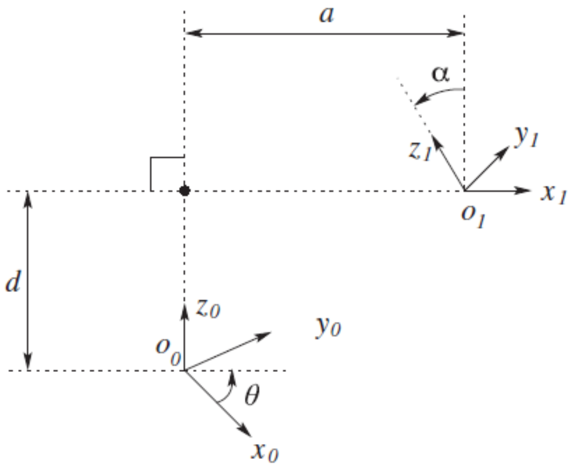


Figure: Source: Robot Modeling and Control

Homogeneous Transformation between links

$$
\begin{aligned}
A_i &= R_z(\theta_i)\,\mathrm{Trans}_z(d_i)\,\mathrm{Trans}_x(a_i)\,R_x(\alpha_i) \\
&= \begin{bmatrix} c\theta_i & -s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
&\quad \times \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_i & -s\alpha_i & 0 \\ 0 & s\alpha_i & c\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} c\theta_i & -s\theta_i\, c\alpha_i & s\theta_i\, s\alpha_i & a_i\, c\theta_i \\ s\theta_i & c\theta_i\, c\alpha_i & -c\theta_i\, s\alpha_i & a_i\, s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}
$$

Proof of the claim

$$A_1 = \begin{bmatrix} {}^0R_1 & {}^0p_1 \\ \mathbf{0} & 1 \end{bmatrix}$$

- We'll show that if the assumptions are true then this A matrix can be written in the form shown on previous slide.
- By (DH1) x_1 is perpendicular to z_0.

$$ {}^0x_1^T \; {}^0z_0 = 0$$

$$\begin{bmatrix} r_{11} & r_{21} & r_{31} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = 0$$

$$\Rightarrow r_{31} = 0$$

Proof: (DH1) gives required rotation matrix.

So,

$$ {}^0R_1 = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ 0 & r_{32} & r_{33} \end{bmatrix} $$

- Since ${}^0R_1$ is rotation matrix,

$$ r_{11}^2 + r_{21}^2 = 1 $$
$$ r_{32}^2 + r_{33}^2 = 1 $$

- Hence, there exist unique θ and α such that

$$ (r_{11}, r_{21}) = (c\theta, s\theta) \text{ and } (r_{33}, r_{32}) = (c\alpha, s\alpha). $$

- Using properties of rotation matrix, it can be shown that the matrix has requisite form.

Proof: (DH1) gives required rotation matrix - Steps

So,

$$
{}^0R_1 = \begin{bmatrix} c\theta & r_{12} & r_{13} \\ s\theta & r_{22} & r_{23} \\ 0 & s\alpha & c\alpha \end{bmatrix}
$$

$$
\begin{align}
r_{12}^2 + r_{22}^2 &= c^2\alpha \tag{1}\\
r_{13}^2 + r_{23}^2 &= s^2\alpha \tag{2}\\
r_{12}c\theta + r_{22}s\theta &= 0 \tag{3}\\
r_{13}c\theta + r_{23}s\theta &= 0 \tag{4}\\
r_{12}r_{13} + r_{22}r_{23} + c\alpha s\alpha &= 0 \tag{5}
\end{align}
$$

From (1) and (3),

$$
(r_{12}, r_{22}) = (-s\theta c\alpha, c\theta c\alpha) \text{ or } (s\theta c\alpha, -c\theta c\alpha)
$$

Similarly, from (2), (4), and (5),

$$
(r_{13}, r_{23}) = (s\theta s\alpha, -c\theta s\alpha) \text{ or } (-s\theta s\alpha, c\theta s\alpha)
$$

Proof: (DH1) gives required rotation matrix - Steps

Two possibilities are:

$$\begin{bmatrix} c\theta & s\theta c\alpha & -s\theta s\alpha \\ s\theta & -c\theta c\alpha & c\theta s\alpha \\ 0 & s\alpha & c\alpha \end{bmatrix} \text{ or } \begin{bmatrix} c\theta & -s\theta c\alpha & s\theta s\alpha \\ s\theta & c\theta c\alpha & -c\theta s\alpha \\ 0 & s\alpha & c\alpha \end{bmatrix}$$

The determinant of left one is -1 where as the one on right is $+1$.

$$\begin{aligned} \begin{vmatrix} c\theta & -s\theta c\alpha & s\theta s\alpha \\ s\theta & c\theta c\alpha & -c\theta s\alpha \\ 0 & s\alpha & c\alpha \end{vmatrix} &= c\theta(c\theta c^2\alpha + c\theta s^2\alpha) \\ &\quad - s\theta(-s\theta c^2\alpha - s\theta s^2\alpha) \\ &= 1 \end{aligned}$$

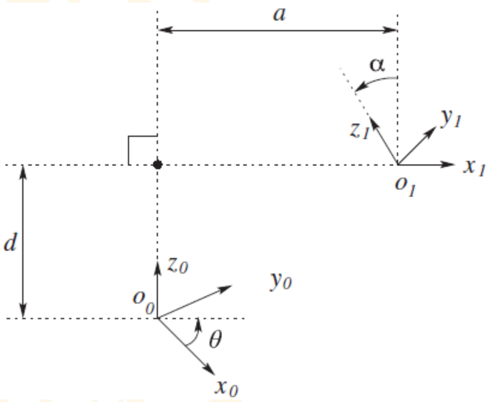


Figure: Source: Robot Modeling and Control

- (DH2) implies

$$
\begin{aligned}
{}^0p_1 &= d_i\,{}^0z_0 + a_i\,{}^0x_1 \\
&= d_i \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + a_i \begin{bmatrix} c\theta_i \\ s\theta_i \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} a_i\, c\theta_i \\ a_i\, s\theta_i \\ d_i \end{bmatrix}
\end{aligned}
$$

Table of Contents

[1] **Mark W Spong, Seth Hutchinson, and Mathukumalli Vidyasagar.**
Robot modeling and control.
John Wiley & Sons, 2020.