Week 3 – Manipulator Kinematics: Forward & Inverse

Advanced Robotic Systems – MANU2453

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

RMIT UNIVERSITY

Lectures

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

Content

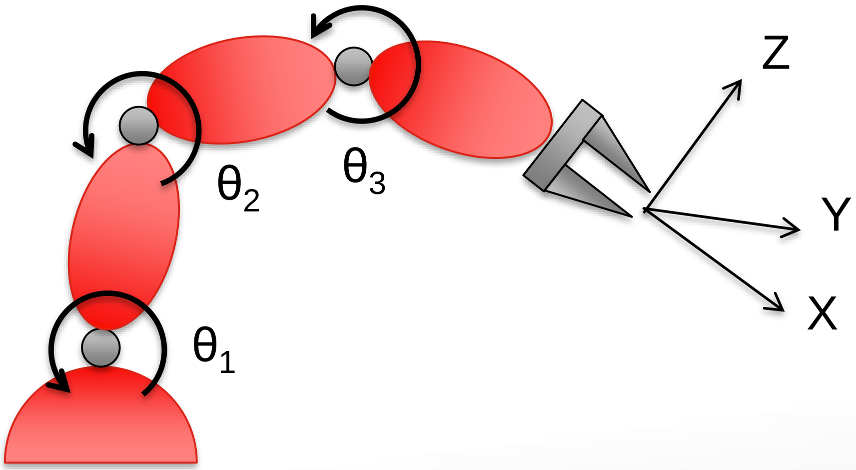
- Forward Kinematics
  - Introduction
  - Denavit-Hartenberg Parameters
  - Introduction to DH Parameters, Link Frame Attachment, Forward Kinematics through Example 1
  - More Examples
- Inverse Kinematics
  - Introduction
  - Algebraic Approach
  - Geometric Approach

Content

- Forward Kinematics
  - Introduction
  - Denavit-Hartenberg Parameters
  - Introduction to DH Parameters, Link Frame Attachment, Forward Kinematics through Example 1
  - More Examples
- Inverse Kinematics
  - Introduction
  - Algebraic Approach
  - Geometric Approach

Introduction

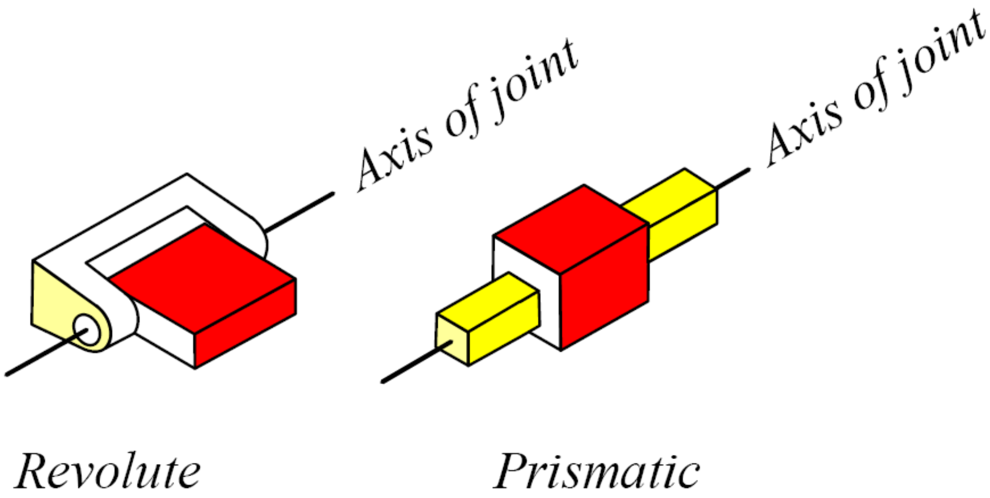
- Kinematics is the study of motion without regard to forces which causes it.
- Of interests are position, velocity, acceleration and higher order derivatives.



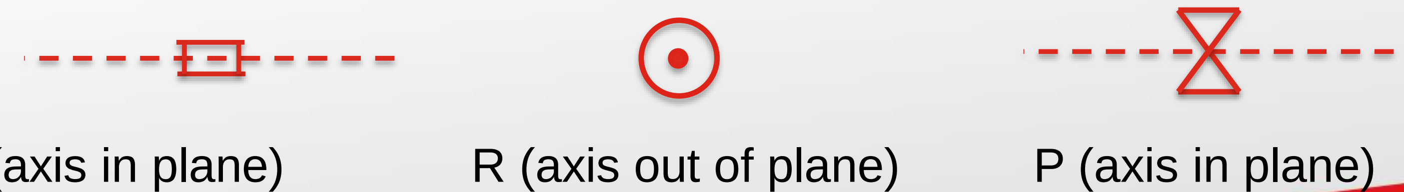
- In this lecture, position and orientation in static situations will be discussed:
  - Given the joint space parameters (angles for revolute joints, or offsets for prismatic joints), as well as the lengths of the links, what is the position and orientation of the end-effector in Cartesian space?

Introduction

- Industrial robots are mostly made by arms / links connected by joints.
  - The joints could be revolute (R) or prismatic (P).

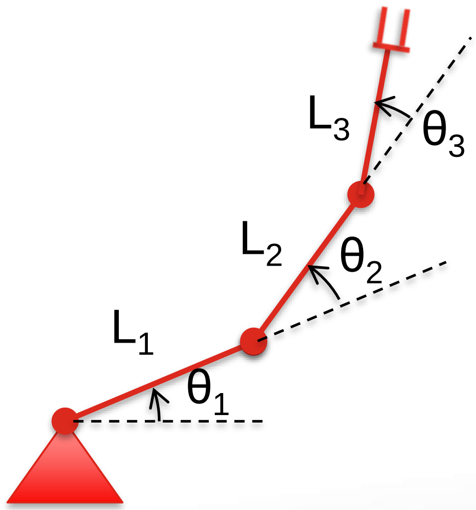


  - Schematics:



A Simple Example

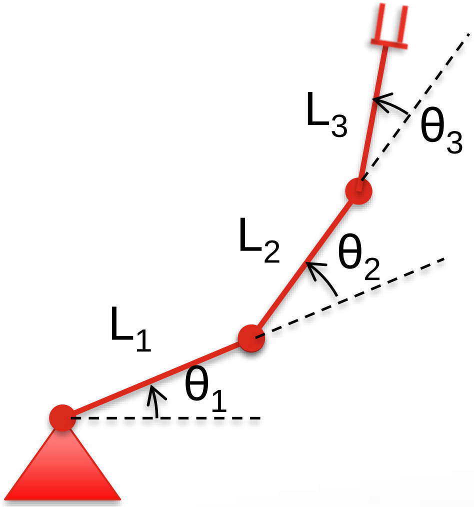
- Consider a 3-link planar robot as shown:



- The link lengths and the joint angles are known.
- What is the position and orientation of the end-effector?

A Simple Example

- In this simple example, the position can be calculated based on trigonometry:

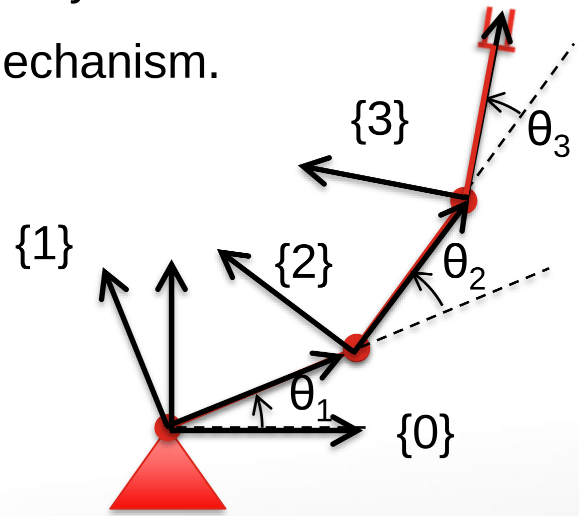


$$x = L_1 \cos\theta_1 + L_2 \cos(\theta_1 + \theta_2) + L_3 \cos(\theta_1 + \theta_2 + \theta_3)$$
$$y = L_1 \sin\theta_1 + L_2 \sin(\theta_1 + \theta_2) + L_3 \sin(\theta_1 + \theta_2 + \theta_3)$$

- The orientation is also straightforward: $\theta_{total} = \theta_1 + \theta_2 + \theta_3$

For More General Cases

- For more general cases (e.g. more links, prismatic joints, non-planar robots), it is important to do this in a more systematic way.
- Frames will be attached to various parts of mechanism.



- Finally, relationship between these frames will be described via Homogeneous Transform.
  - ${}^0_1T, {}^1_2T, {}^2_3T$ known individually.
  - Then calculate ${}^0_3T = {}^0_1T \cdot {}^1_2T \cdot {}^2_3T$ to get the position of frame {3}.

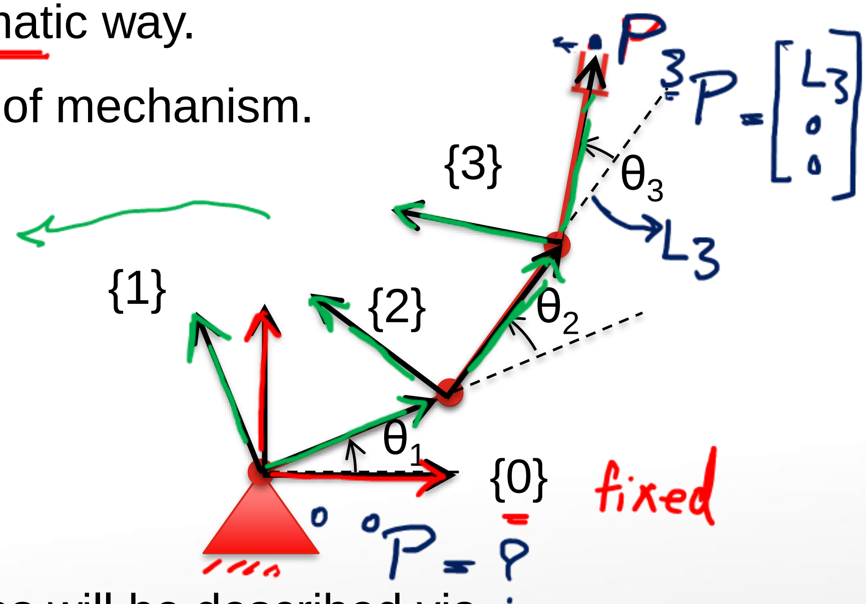
For More General Cases

- For more general cases (e.g. more links, prismatic joints, non-planar robots), it is important to do this in a more systematic way.
- Frames will be attached to various parts of mechanism.

Compound Transf.

→ known ${}^0_1T, {}^1_2T, {}^2_3T$

→ calculate ${}^0_3T = {}^0_1T \cdot {}^1_2T \cdot {}^2_3T$



- Finally, relationship between these frames will be described via Homogeneous Transform.
  - ${}^0_1T, {}^1_2T, {}^2_3T$ known individually.
  - Then calculate ${}^0_3T = {}^0_1T \cdot {}^1_2T \cdot {}^2_3T$ to get the position of frame {3}.

→ ${}^3P = [L_3 \; 0 \; 0]^T$ → ${}^0P = {}^0_3T \cdot {}^3P$

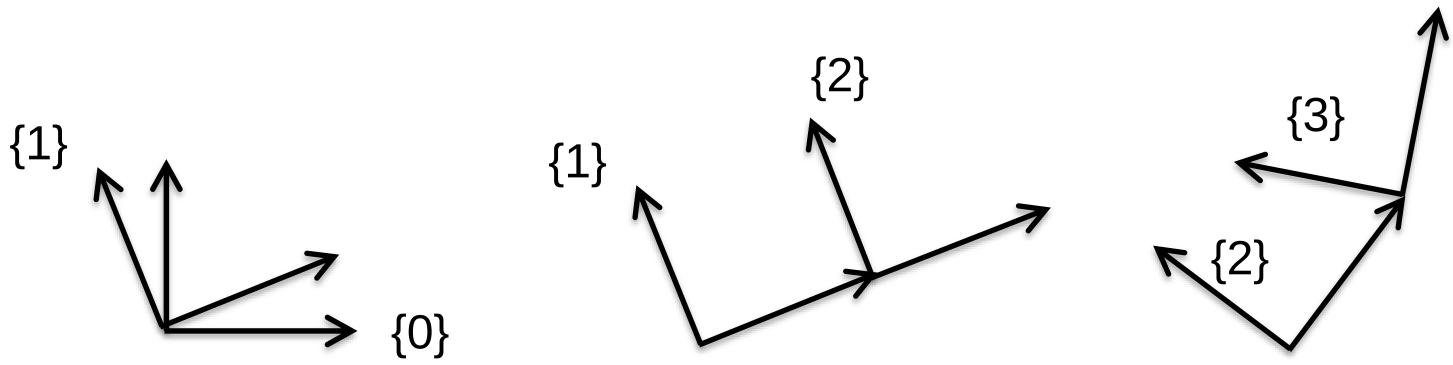
Content

- Forward Kinematics
  - Introduction
  - Denavit-Hartenberg Parameters
  - Introduction to DH Parameters, Link Frame Attachment, Forward Kinematics through Example 1
  - More Examples
- Inverse Kinematics
  - Introduction
  - Algebraic Approach
  - Geometric Approach

RMIT UNIVERSITY

Denavit-Hartenberg Parameters

- We mentioned that ${}^{0}_{1}T, {}^{1}_{2}T, {}^{2}_{3}T$ can be obtained individually.



- Question: How many parameters are needed to describe each transformation? In general transformations, six: three for position and three for orientation.
- However, in robots, certain values are fixed.
  - E.g. the distance between origins of {1} and {2} – constrained by the link length.
  - Also, each joint is constrained to only 1 DOF.

Denavit-Hartenberg Parameters

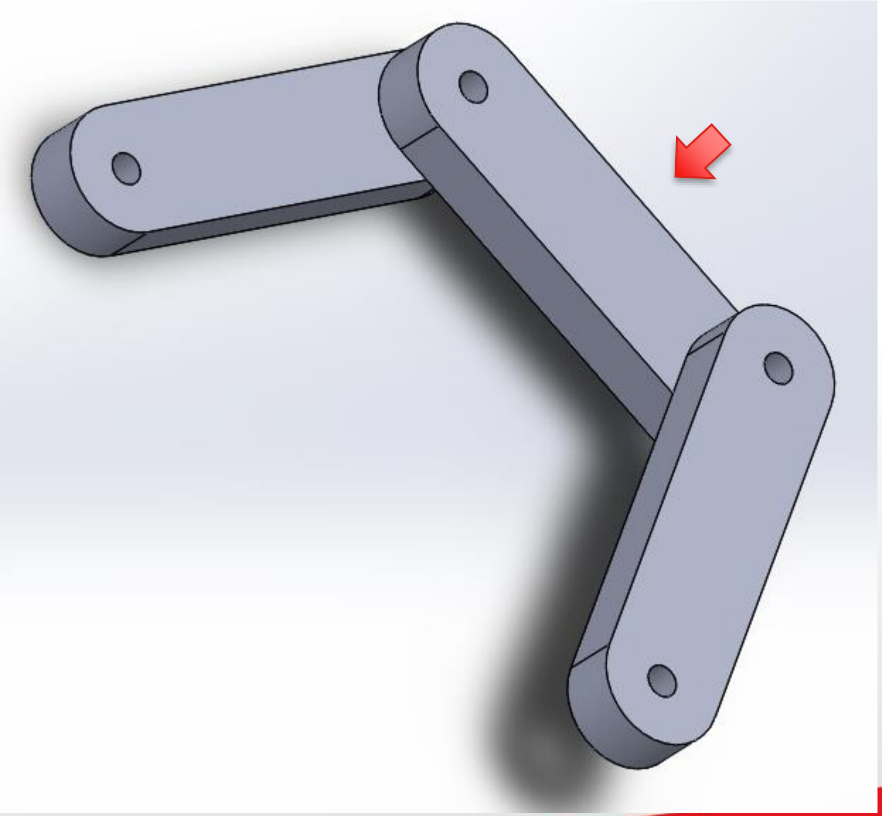
- Therefore, for robotic manipulators, we only need 4 parameters to describe the transformation between each links.
- They are:
  - Link lengths $a_{i\text{-}1}$
  - Link twists $\alpha_{i\text{-}1}$
  - Joint angles θ_i
  - Link offsets d_i
- The above four parameters are called Denavit-Hartenberg Parameters, or in short "DH Parameters".
- We will learn about the DH Parameters, how to attach the frames, and how to get the Cartesian description of the last link, through examples in the next sections.

Content

- Forward Kinematics
  - Introduction
  - Denavit-Hartenberg Parameters
  - Introduction to DH Parameters, Link Frame Attachment, Forward Kinematics through Example 1
  - More Examples
- Inverse Kinematics
  - Introduction
  - Algebraic Approach
  - Geometric Approach

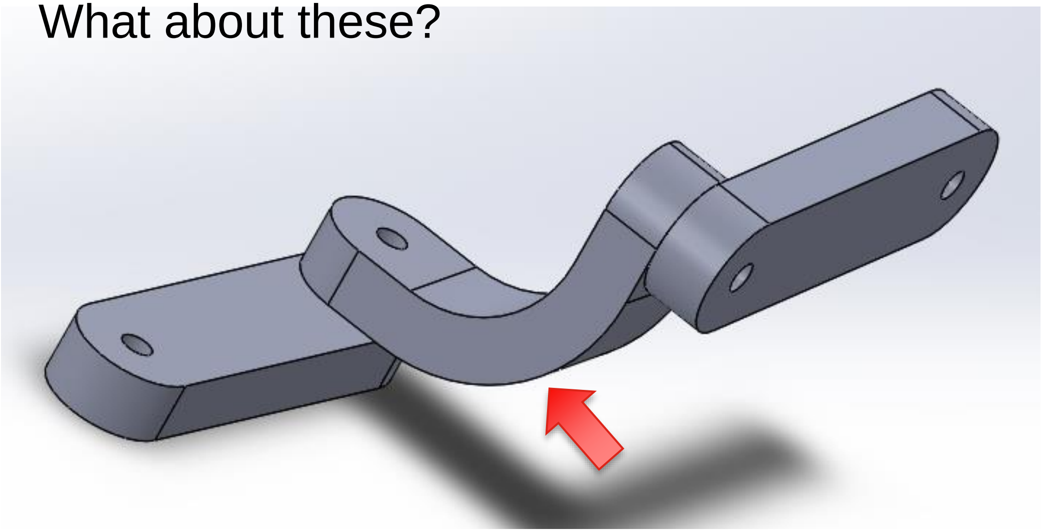
Link Length

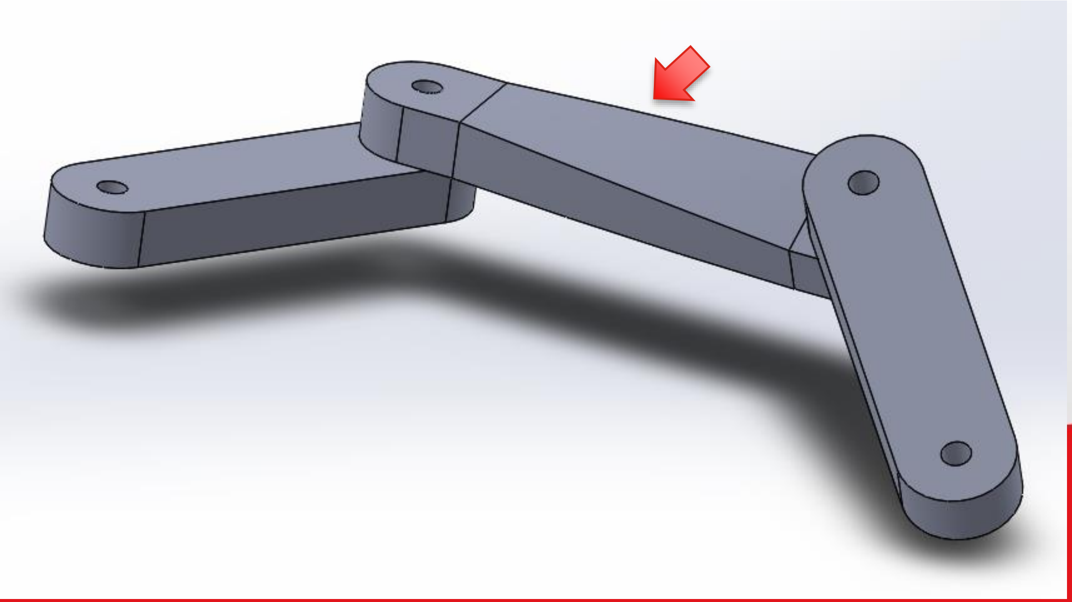
- A link is basically just a rigid body between two joint axes.
- It turns out that a link can be fully described by its length and its twist.
- What is its length?



Link Length

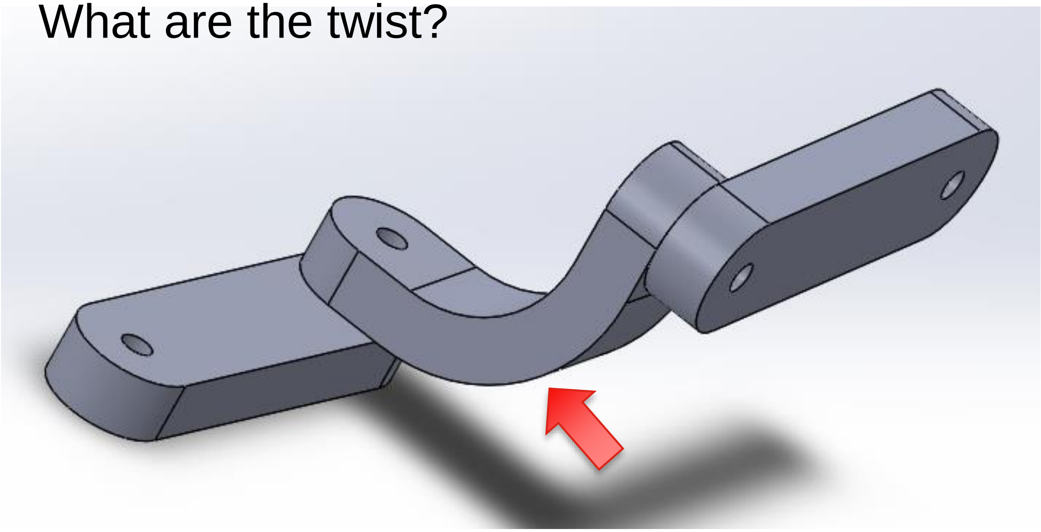
- What about these?

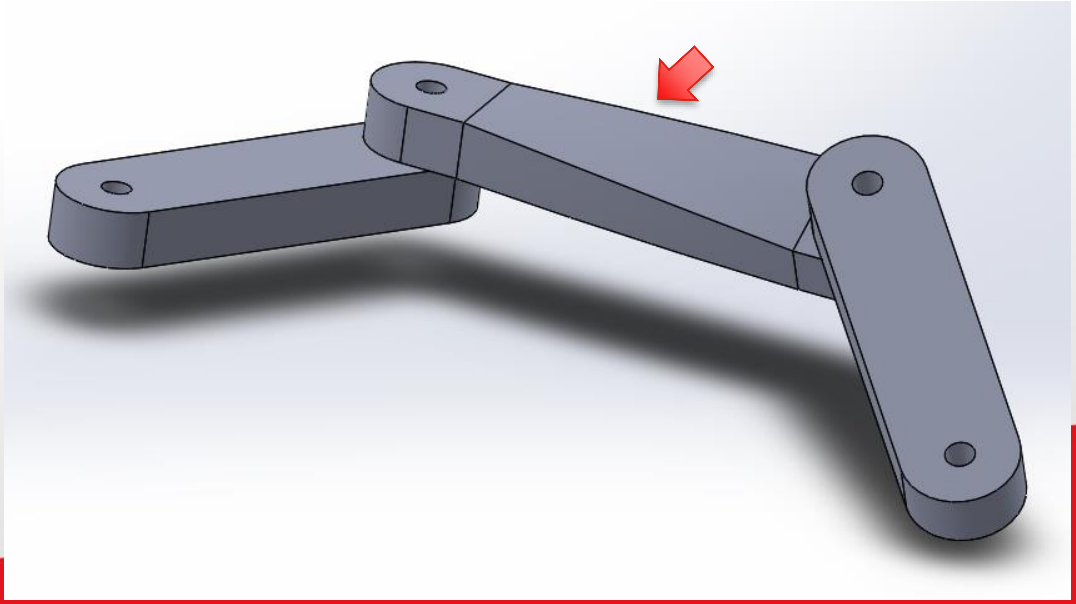




Link Twist

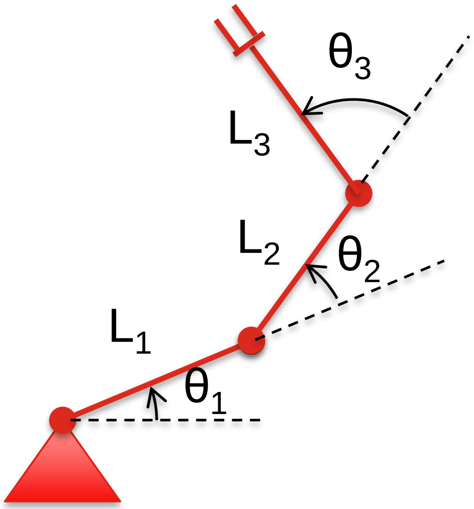
- What are the twist?

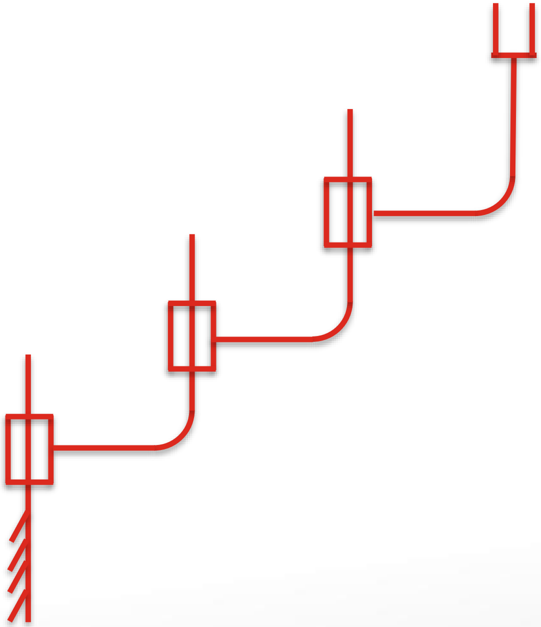




Example 1

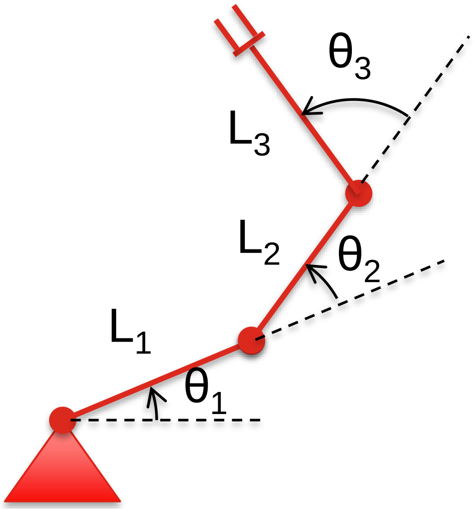
- 3-link RRR planar robot:

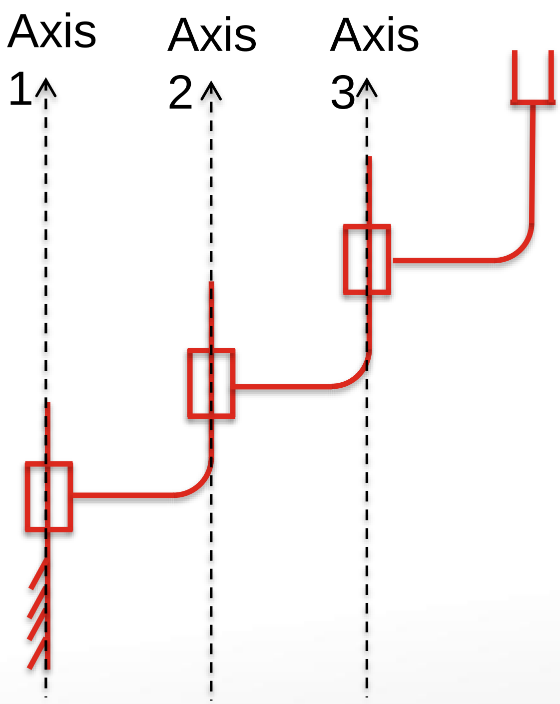




Example 1

- 3-link RRR planar robot:

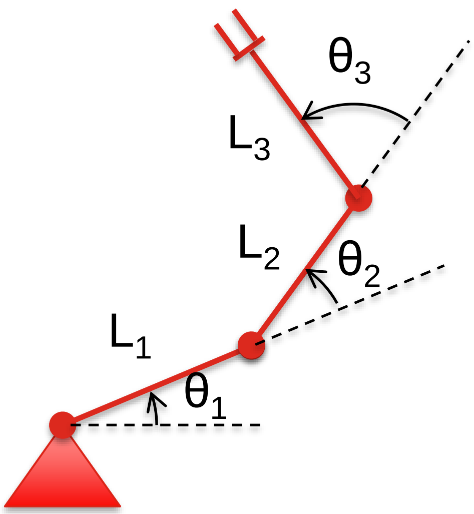


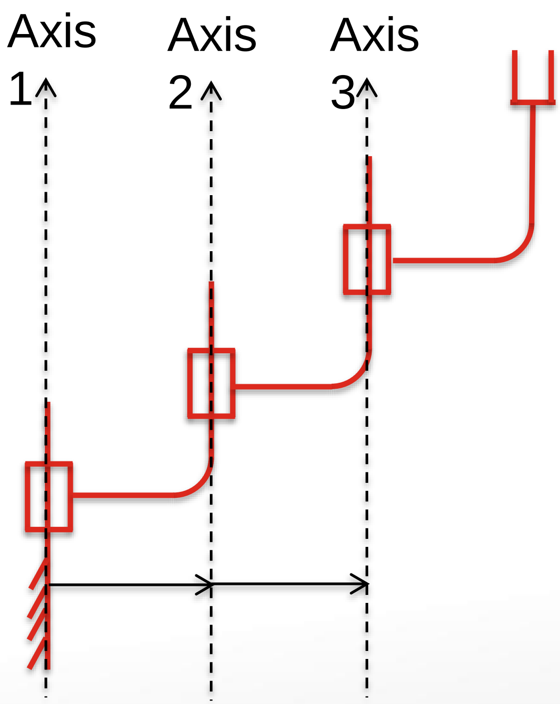


- Step 1, draw the axes.
  - For rotary joint: About the rotation
  - For prismatic joint: Along the translation

Example 1

- 3-link RRR planar robot:

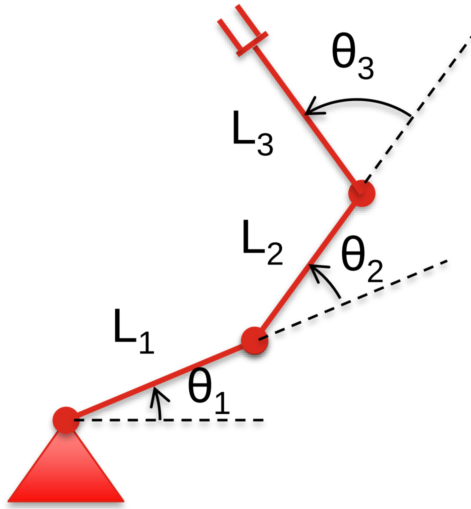


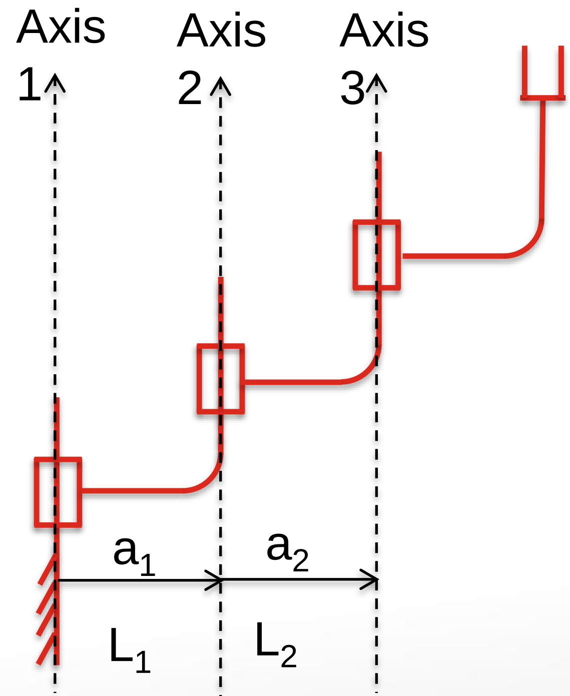


- Step 2, we draw the mutually perpendicular lines between axes.
  - Draw from lower axis to higher axis, e.g. 1 to 2, 2 to 3…
  - Because the axes are parallel, the mutual perpendicular can be placed in an arbitrary position.
  - We will put them in the same plane.

Example 1

- 3-link RRR planar robot:



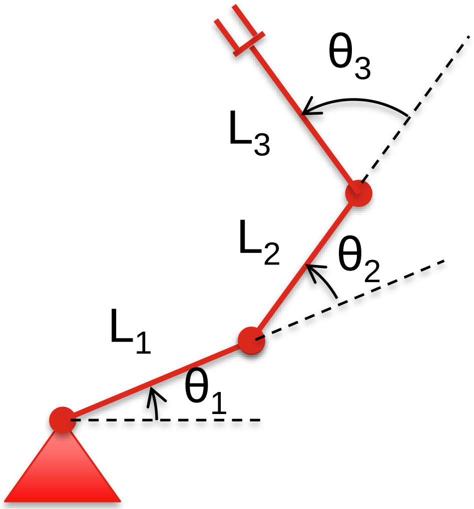


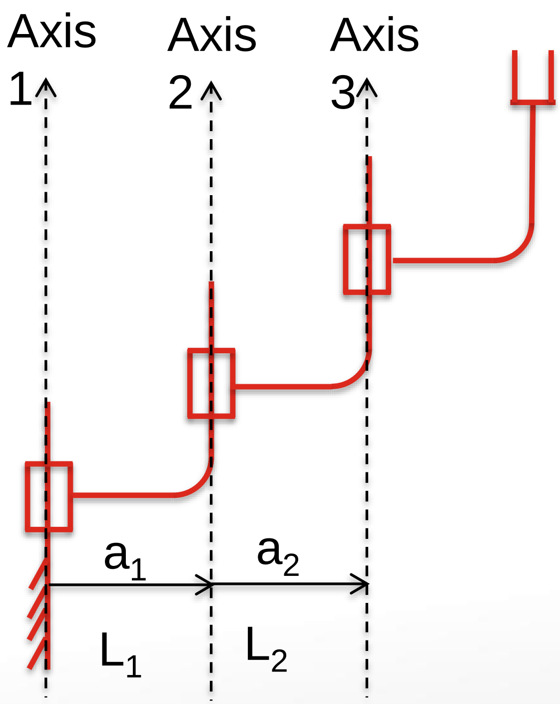
- Step 3, put in the link lengths a_{i-1}.
- Definition: a_{i-1} = length of mutual perpendicular, from axis i-1 to axis i.

Do 1 to n-1
- a_1 = length of mutual perpendicular from axis 1 to 2 = L_1.
- a_2 = length of mutual perpendicular from axis 2 to 3 = L_2.

Example 1

- 3-link RRR planar robot:

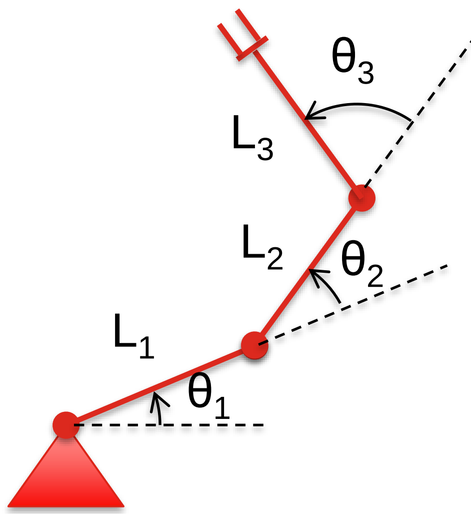




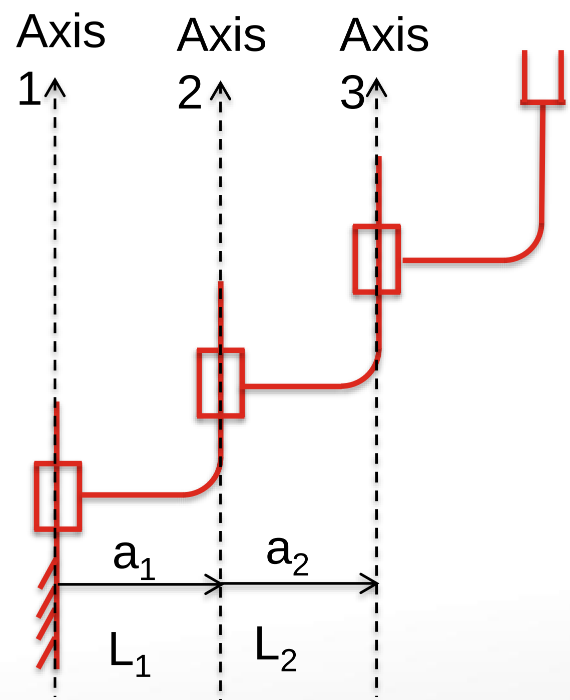
- What about a_0?
  - a_0 = length of mutual perpendicular from axis 0 to 1. However, axis 0 is not known yet.

Example 1

- 3-link RRR planar robot:



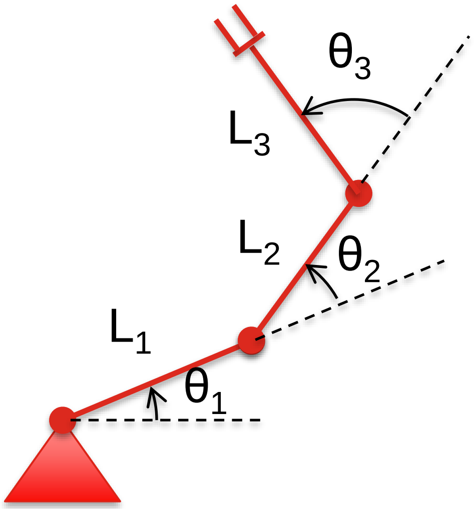
Axis 0 intersecting with

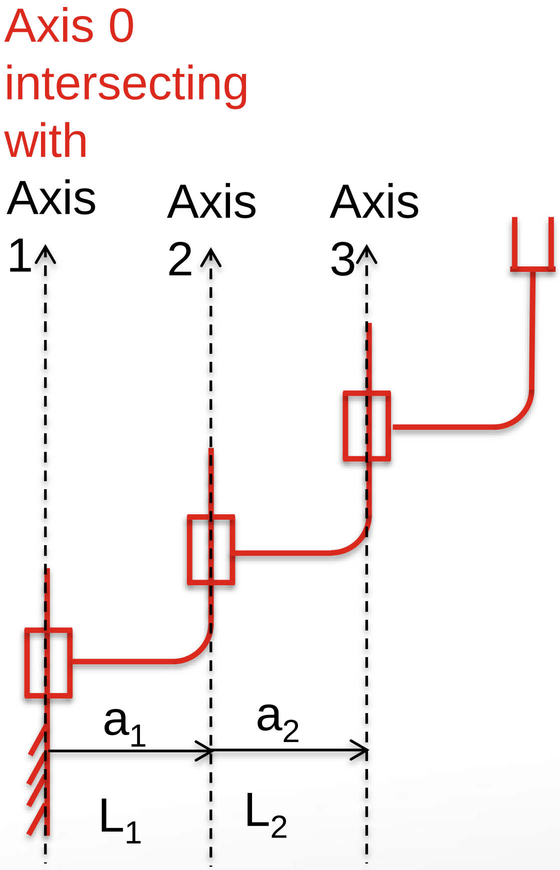


- What about a_0?
  - a_0 = length of mutual perpendicular from axis 0 to 1. However, axis 0 is not known yet.
  - By convention, $a_0 = 0$.
  - This means: Axis 0 and Axis 1 intersect with each other.

Example 1

- 3-link RRR planar robot:





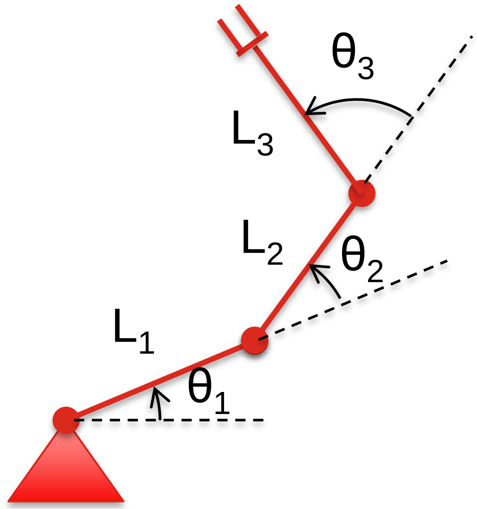
- Step 4, put in link twists α_{i-1}.
- Definition: α_{i-1} = angle between axis i-1 and axis i, in the right hand sense about a_{i-1}

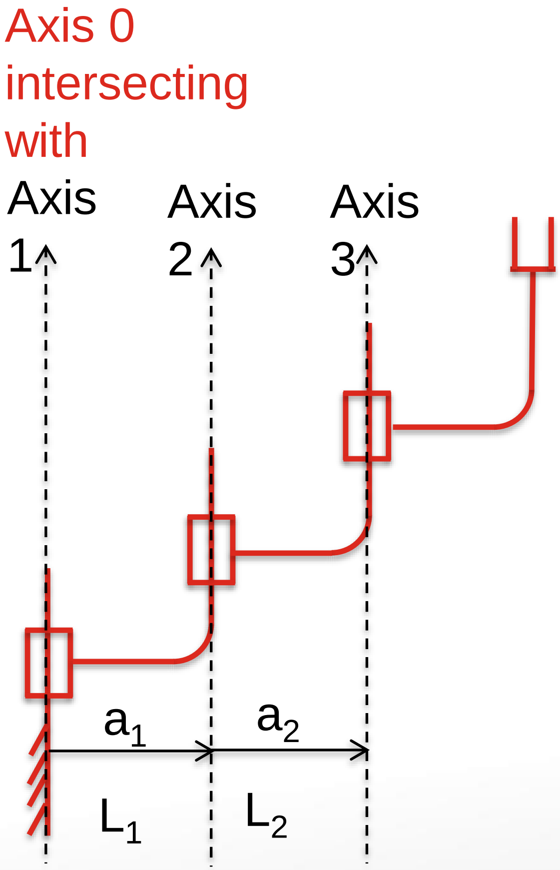
Do 1 to n-1

- α_1 = angle between axis 1 and axis 2, about a_1 = 0deg.
- α_2 = angle between axis 2 and axis 3, about a_2 = 0deg.

Example 1

- 3-link RRR planar robot:

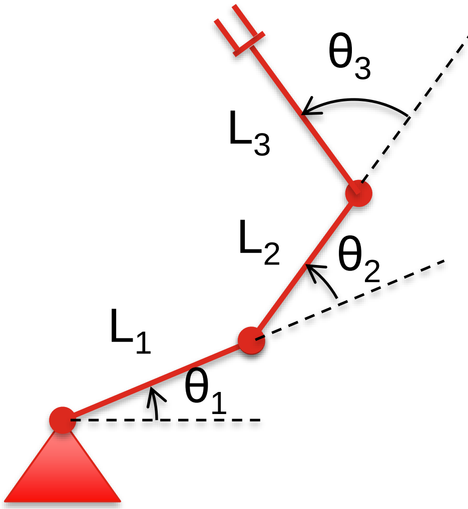


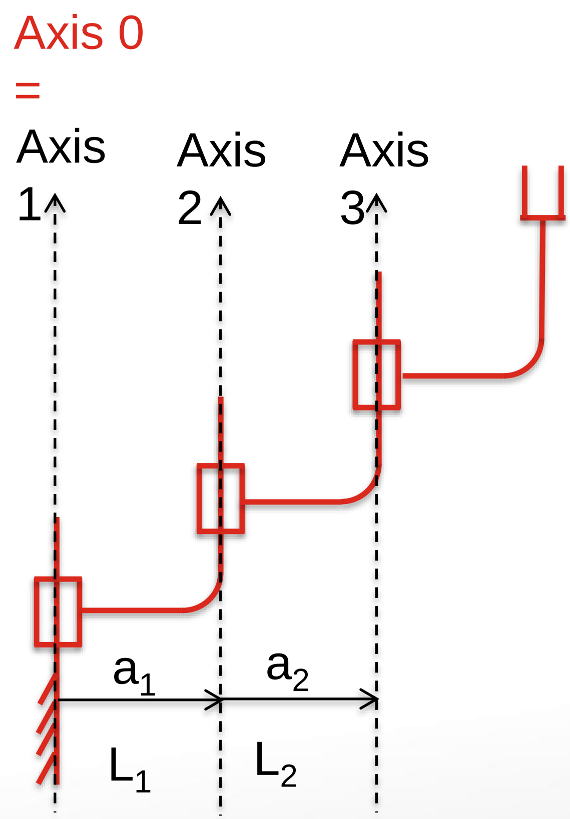


- What about α_0?
  - α_0 = angle between axis 0 and axis 1, about a_0 . However, axis 0 is not fully known yet.

Example 1

- 3-link RRR planar robot:

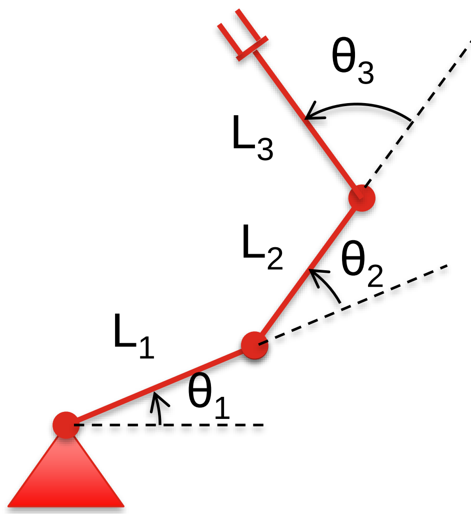


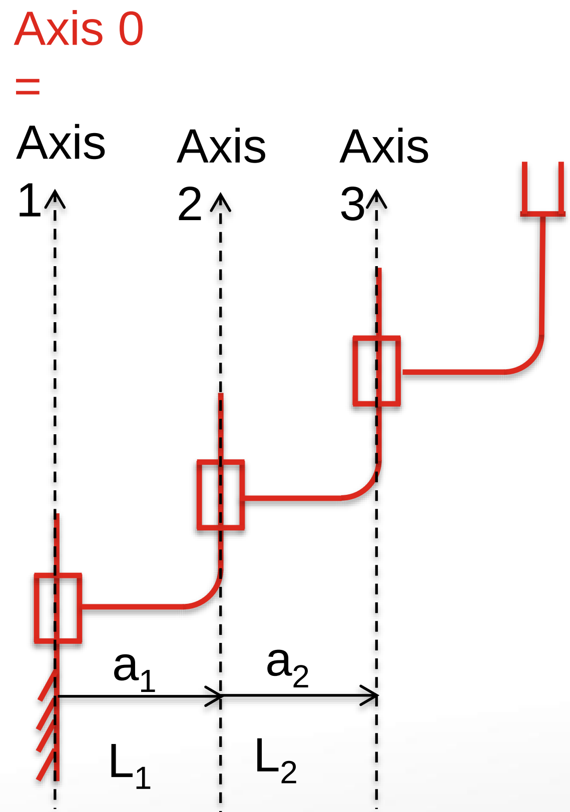


- What about α_0?
  - α_0 = angle between axis 0 and axis 1, about a_0 . However, axis 0 is not fully known yet.
  - By convention, $\alpha_0 = 0$.
  - This means: Axis 0 and Axis 1 are the same.

Example 1

- 3-link RRR planar robot:





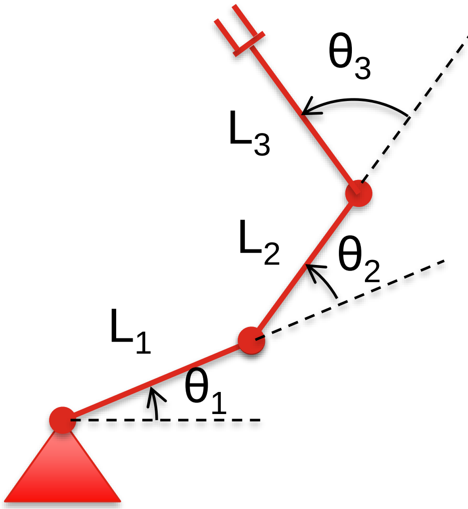
- Step 5, write down the link offsets d_i.
- Definition: d_i = distance from a_{i-1} to a_i, along axis i.

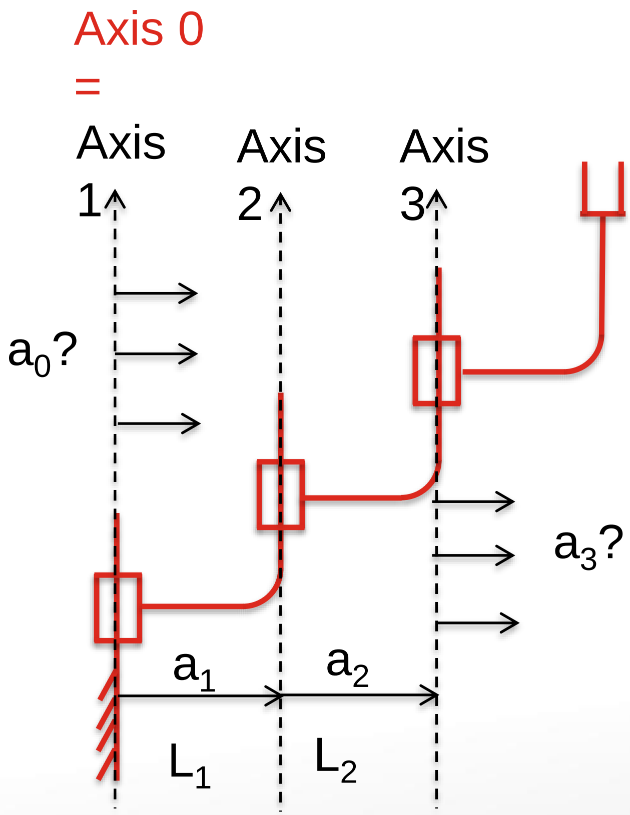
Do 2 to n-1

- d_2 = distance from a1 to a2, along axis 2 = 0.

Example 1

- 3-link RRR planar robot:

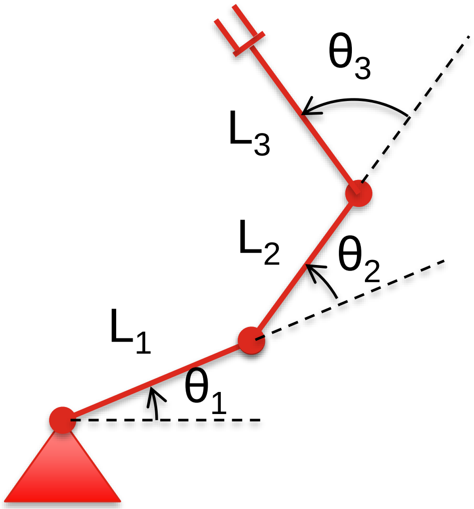


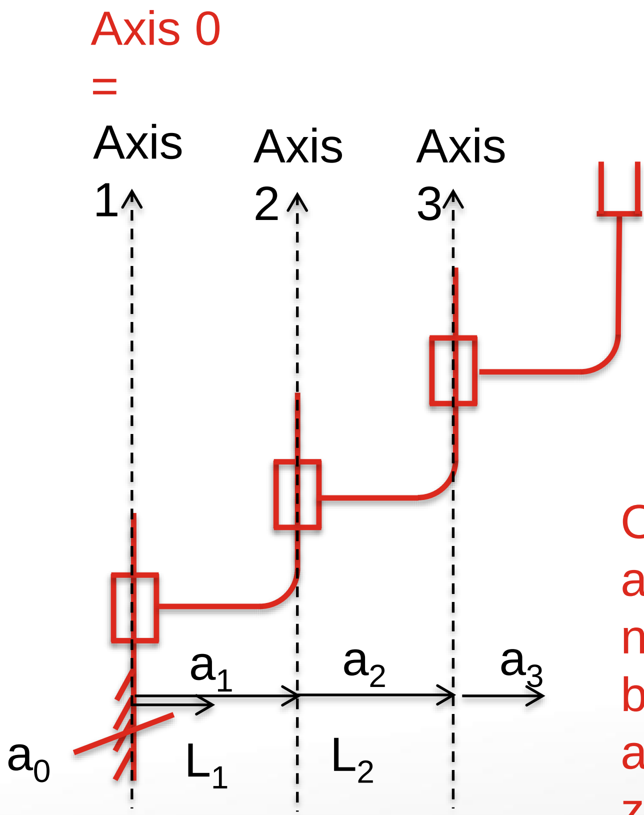


- What about d_1 and d_3?
  - d_1 = distance from a_0 to a_1, along axis 1.
  - d_3 = distance from a_2 to a_3, along axis 3.
  - But where exactly are a_0 and a_3?

Example 1

- 3-link RRR planar robot:



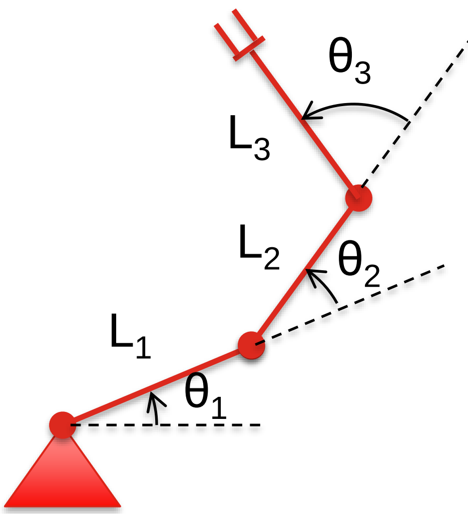


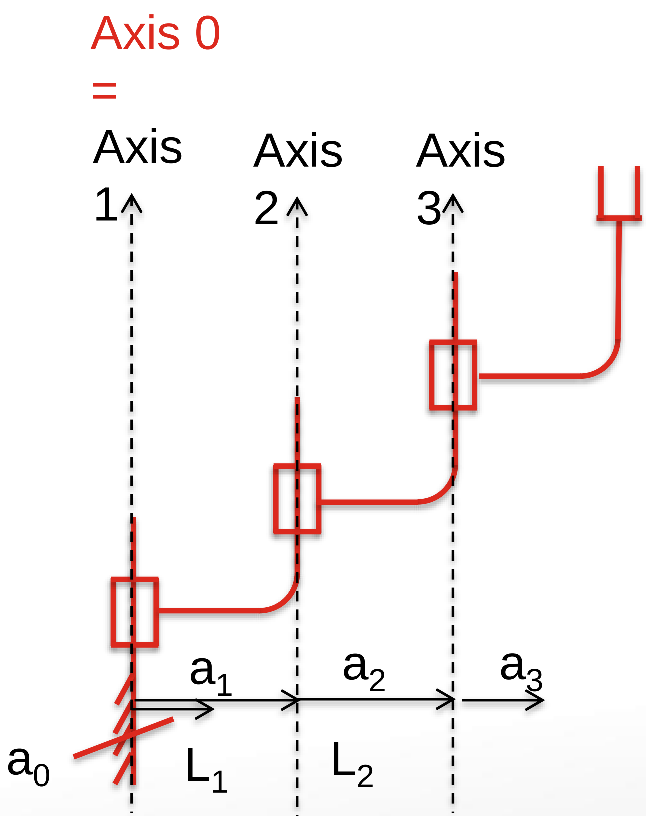
Origins of a_0 and a_3 are now known, because d_1 and d_3 are zero.

- What about d_1 and d_3?
  - d_1 = distance from a_0 to a_1, along axis 1.
  - d_3 = distance from a_2 to a_3, along axis 3.
  - By convention: Zero for revolute joint, variable for prismatic joint
  - So in this case, d_1 and d_3 are both zero.

Example 1

- 3-link RRR planar robot:





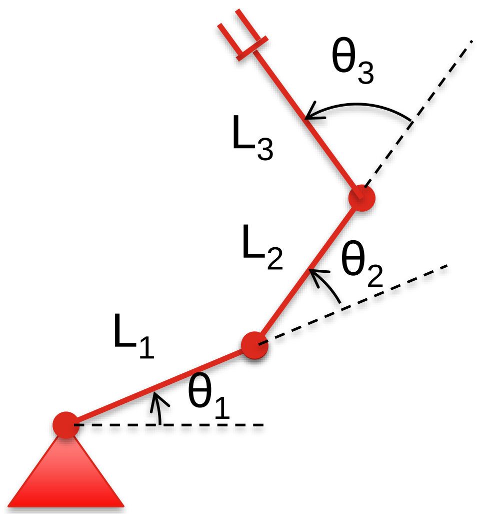
- Step 6, write down the joint angle θ_i.
- Definition: θ_i is the angle between the (extension of a_{i-1}) and a_i, measured about the axis i.

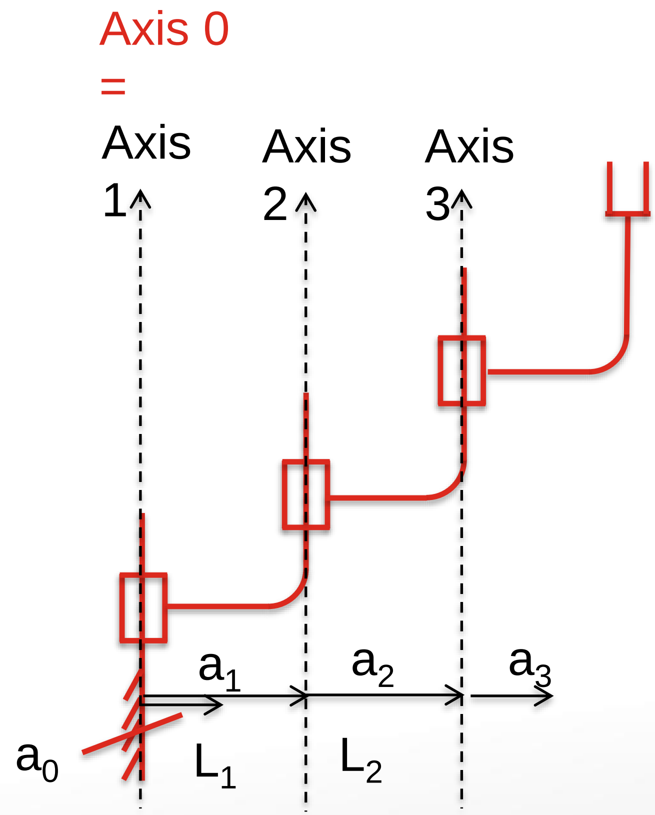
Do 2 to n-1

- θ_2 = angle between (extension of a1) and a2, about axis 2.
- It is a variable because the joint is revolute.

Example 1

- 3-link RRR planar robot:

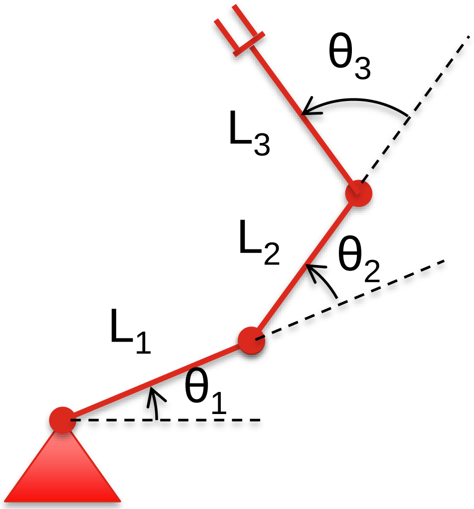


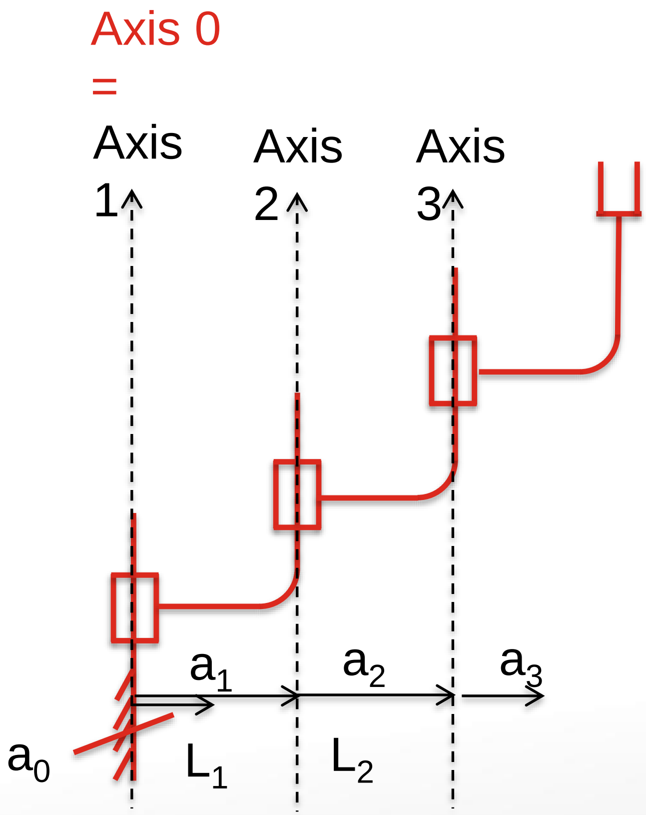


- What about θ_1 and θ_3?
  - θ_1 = angle between (extension of a0) and a1, about axis 1.
  - θ_3 = angle between (extension of a2) and a3, about axis 3.
  - By convention: Zero for prismatic joint, variable for revolute joint
  - So in this case, θ_1 and θ_3 are both variables.

Example 1

- 3-link RRR planar robot:

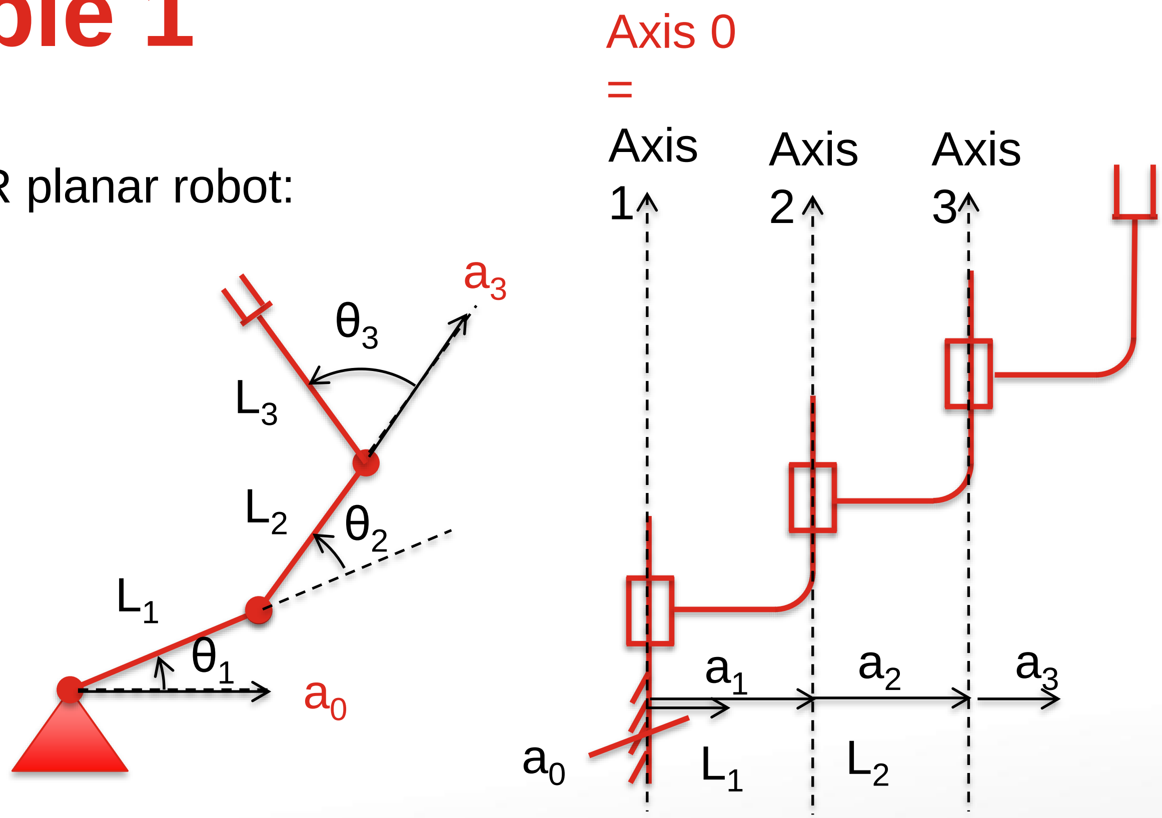




- We still have a problem. Since θ_1 and θ_3 are both variables, we need to determine their “zero”-angle position.

Example 1

- 3-link RRR planar robot:



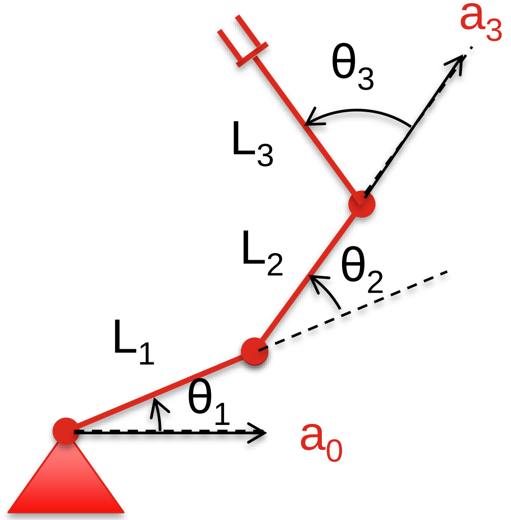
- For convenience, align a_0 with a_1 when the joint variable 1 is zero.
- As for joint n:
  - Revolute: align a_n with a_{n-1} when $\theta_n = 0$.
  - Prismatic: align a_n with a_{n-1} when $d_n = 0$.

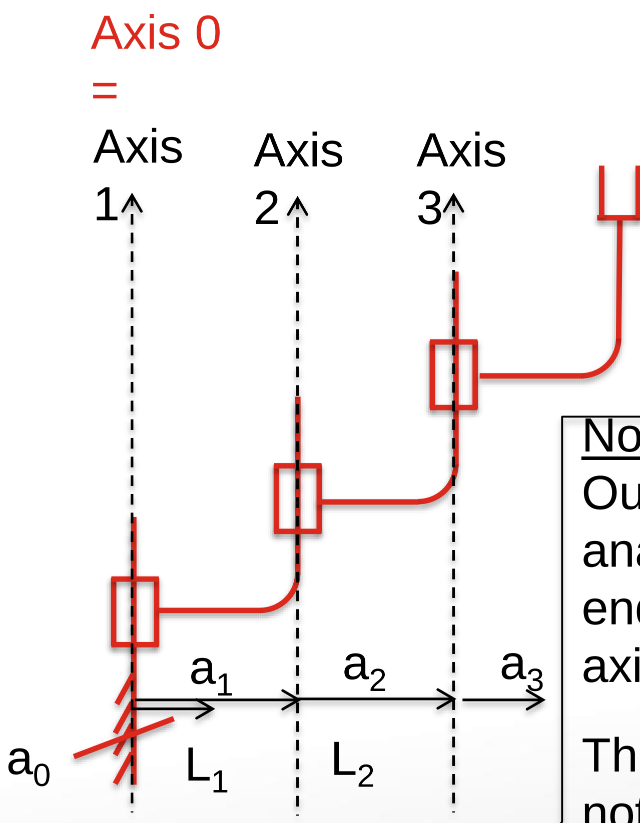
Aside: DH Parameters

- In summary, any robot can be described kinematically using the four parameters we have just learnt:
  - Link length a_{i-1}.
  - Link twist α_{i-1}.
  - Link offset d_i.
  - Joint angle θ_i.
- Important Notes:
  - a_{i-1}, α_{i-1} are always constants
  - Revolute joint:
    - d_i constants
    - θ_i variable.
  - Prismatic joint:
    - θ_i constants
    - d_i variable.

Example 1

- 3-link RRR planar robot:





Note:
Our kinematic analysis always ends at last joint axis.

Therefore L_3 does not appear in the table.

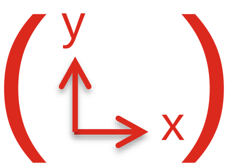
It is a final offset which can be added later.

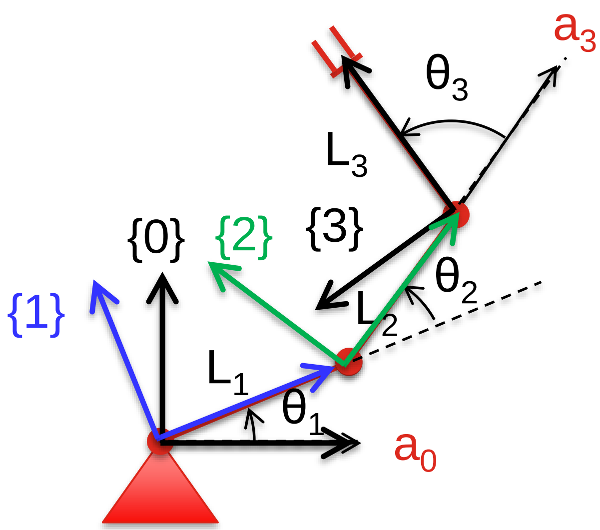
- Step 7, transfer to a DH-table:

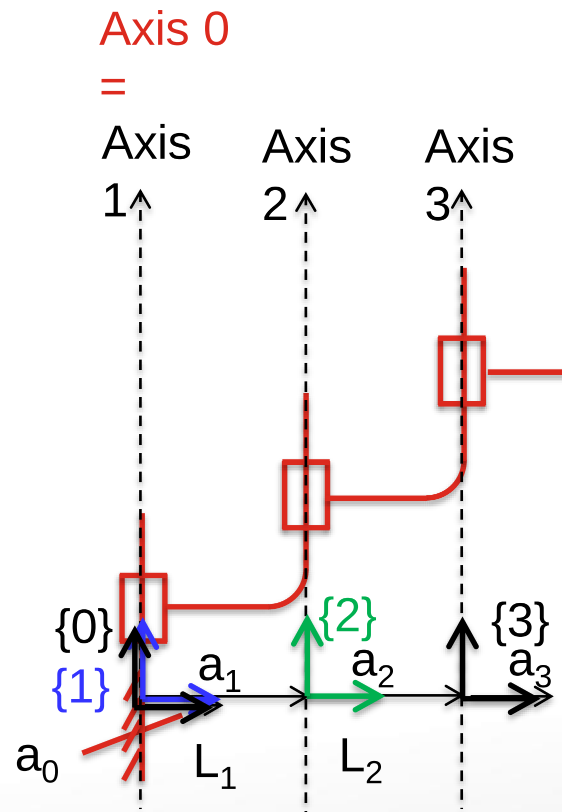
i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	0	L_1	0	θ_2
3	0	L_2	0	θ_3

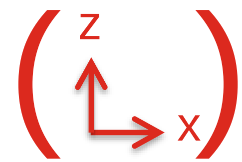
Example 1

- 3-link RRR planar robot:







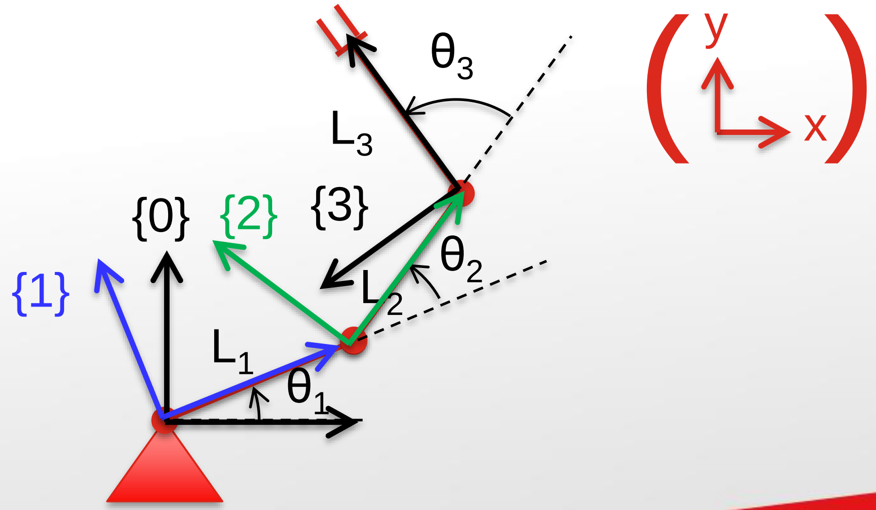


- Step 8, insert the frames. Rules:
  - Z-axis of frame {i}, i.e. Z_i, is coincident with joint axis i.
  - Origin of frame {i} is where the a_i intersects the joint i axis.
  - X-axis of frame {i}, i.e. X_i, is coincident with a_i.

Aside: Link Transformation

- It should be quite clear now why we attached frames to the links, and also why we determined the DH parameters for the links and joints.
  - We want to able to do transformation from frame {0} to {1}, then {1} to {2}, then {2} to {3} and so on.
  - Then by compound transformation, we can get the transformation from link the base to end-effector.

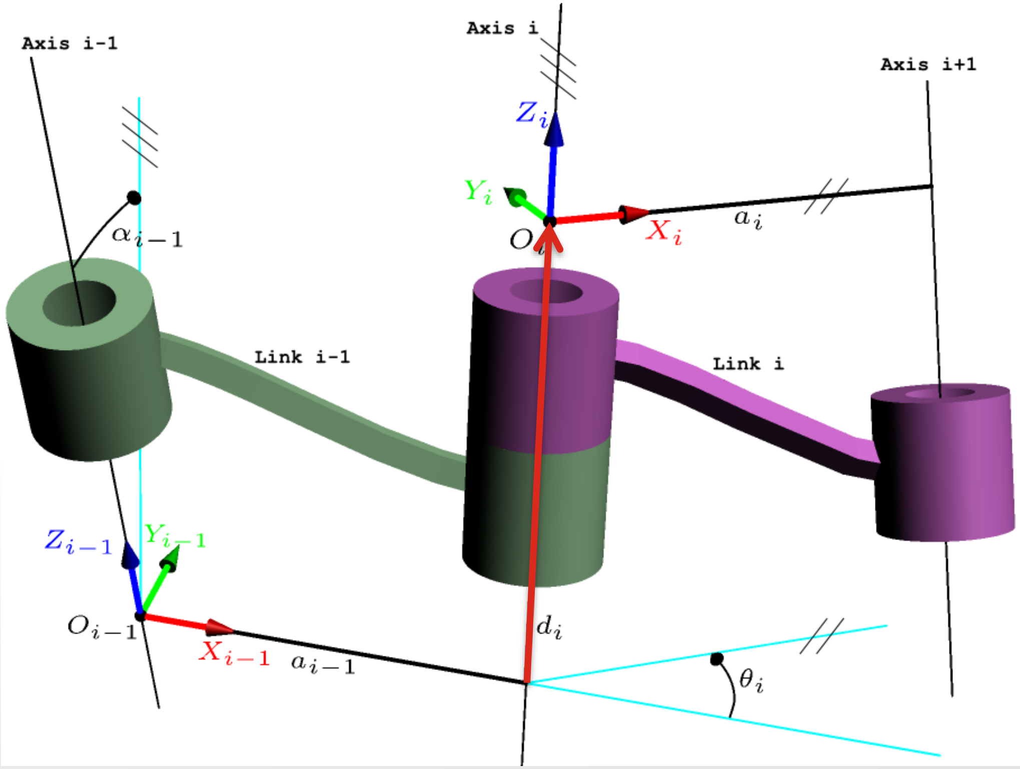
Forward Kinematics!



Aside: Link Transformation

- Each of the transformation can be simplified to 4 sub-problems:

$$ {}^{i-1}_{i}T = R_x(\alpha_{i-1}) \cdot D_x(a_{i-1}) \cdot R_z(\theta_i) \cdot D_z(d_i) $$



Aside: Link Transformation

$$
{}^{i-1}_{i}T = R_x(\alpha_{i-1}) \cdot D_x(a_{i-1}) \cdot R_z(\theta_i) \cdot D_z(d_i)
$$

$$
= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_{i-1} & -s\alpha_{i-1} & 0 \\ 0 & s\alpha_{i-1} & c\alpha_{i-1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_{i-1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\theta_i & -s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
= \begin{bmatrix} 1 & 0 & 0 & a_{i-1} \\ 0 & c\alpha_{i-1} & -s\alpha_{i-1} & 0 \\ 0 & s\alpha_{i-1} & c\alpha_{i-1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\theta_i & -s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ c\alpha_{i-1}s\theta_i & c\alpha_{i-1}c\theta_i & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\alpha_{i-1}s\theta_i & s\alpha_{i-1}c\theta_i & c\alpha_{i-1} & c\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

- Finally: $${}^{0}_{N}T = {}^{0}_{1}T \cdot {}^{1}_{2}T \cdot {}^{2}_{3}T \cdots {}^{N-1}_{N}T$$

Example 1

- Step 9 (Final step!), calculate the transformations.

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	0	L_1	0	θ_2
3	0	L_2	0	θ_3

$$ {}^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ c\alpha_{i-1}s\theta_i & c\alpha_{i-1}c\theta_i & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\alpha_{i-1}s\theta_i & s\alpha_{i-1}c\theta_i & c\alpha_{i-1} & c\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{0}_{1}T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{1}_{2}T = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & L_1 \\ s\theta_2 & c\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{2}_{3}T = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & L_2 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{0}_{3}T = {}^{0}_{1}T \cdot {}^{1}_{2}T \cdot {}^{2}_{3}T = \begin{bmatrix} c\theta_{123} & -s\theta_{123} & 0 & L_2c\theta_{12} + L_1c\theta_1 \\ s\theta_{123} & c\theta_{123} & 0 & L_2s\theta_{12} + L_1s\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

where

$$ \theta_{12} = \theta_1 + \theta_2 $$
$$ \theta_{123} = \theta_1 + \theta_2 + \theta_3 $$

Example 1

- Step 10.

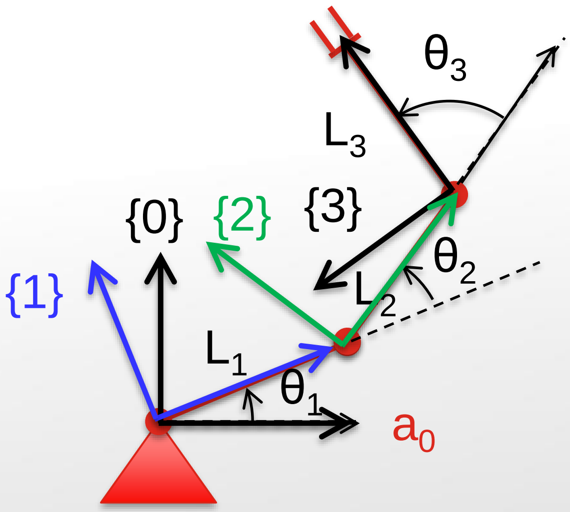
The end-effector, with reference to frame {3}, has position $[L_3, 0, 0]^T$, and same orientation as frame {3}. Therefore:

$$
{}^{0}P = {}^{0}_{3}T \cdot {}^{3}P = \begin{bmatrix} c\theta_{123} & -s\theta_{123} & 0 & L_2 c\theta_{12} + L_1 c\theta_1 \\ s\theta_{123} & c\theta_{123} & 0 & L_2 s\theta_{12} + L_1 s\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} L_3 \\ 0 \\ 0 \\ 1 \end{bmatrix}
$$

Orientation

$$
= \begin{bmatrix} L_3 c\theta_{123} + L_2 c\theta_{12} + L_1 c\theta_1 \\ L_3 s\theta_{123} + L_2 s\theta_{12} + L_1 s\theta_1 \\ 0 \\ 1 \end{bmatrix}
$$

Position



Example 1

- Verification: At the beginning of this lecture, we showed that the position and orientation of the end-effector are:

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

$$\theta_{total} = \theta_1 + \theta_2 + \theta_3$$

- Let's verify using the frame transformation method:

$$ {}^0P = {}^0_3T \cdot {}^3P = \begin{bmatrix} c\theta_{123} & -s\theta_{123} & 0 & L_2c\theta_{12} + L_1c\theta_1 \\ s\theta_{123} & c\theta_{123} & 0 & L_2s\theta_{12} + L_1s\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} L_3 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Orientation

$$= \begin{bmatrix} L_3c\theta_{123} + L_2c\theta_{12} + L_1c\theta_1 \\ L_3s\theta_{123} + L_2s\theta_{12} + L_1s\theta_1 \\ 0 \\ 1 \end{bmatrix}$$

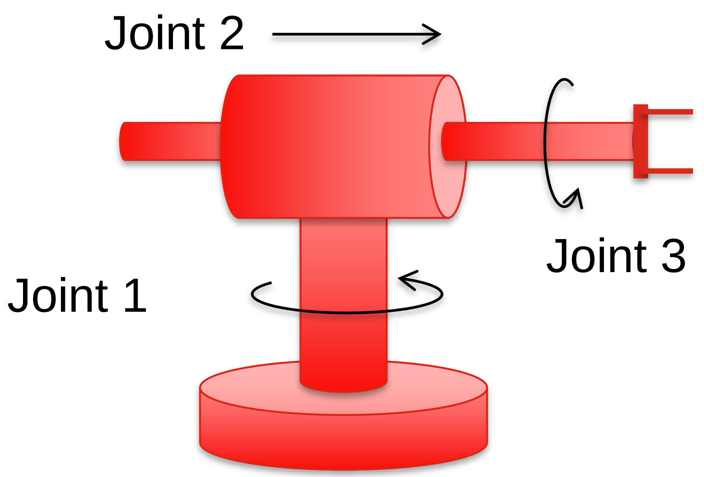
Position

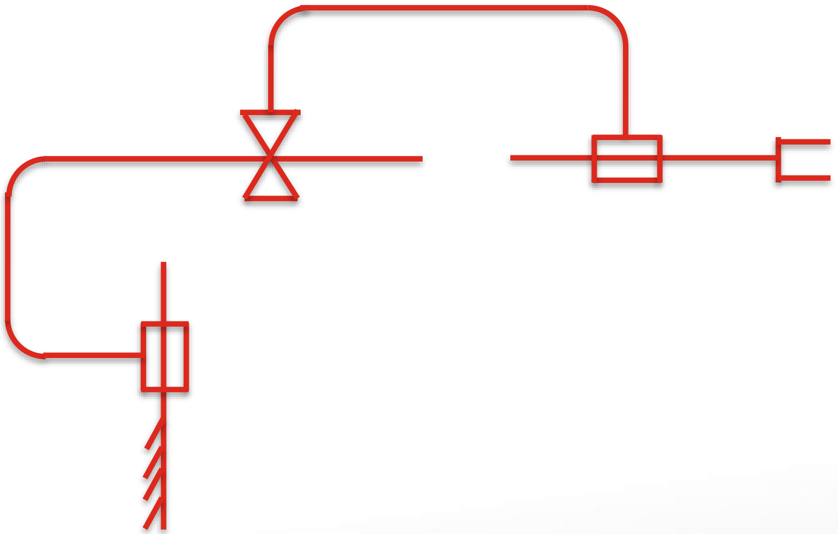
Content

- Forward Kinematics
  - Introduction
  - Denavit-Hartenberg Parameters
  - Introduction to DH Parameters, Link Frame Attachment, Forward Kinematics through Example 1
  - More Examples
- Inverse Kinematics
  - Introduction
  - Algebraic Approach
  - Geometric Approach

Example 2

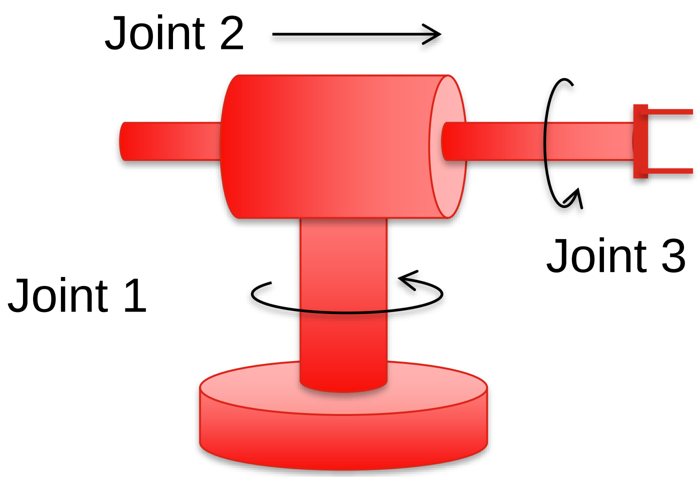
- 3-link RPR robot:

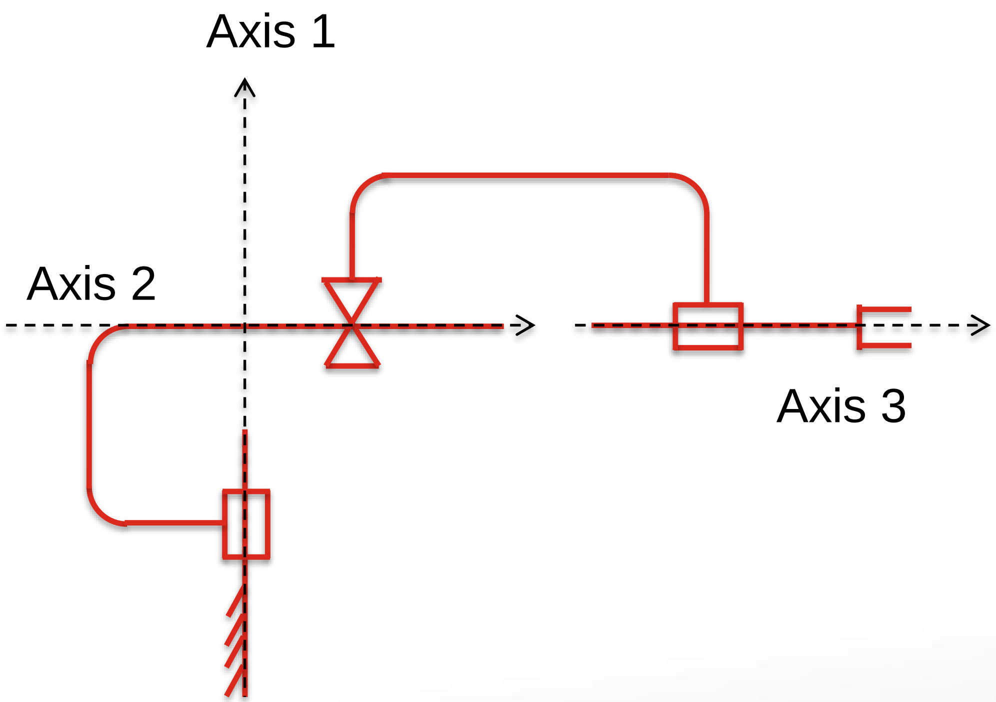




Example 2

- 3-link RPR robot:

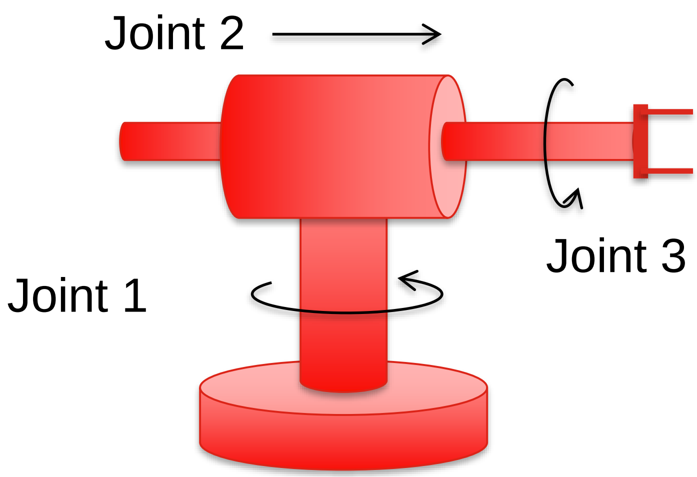


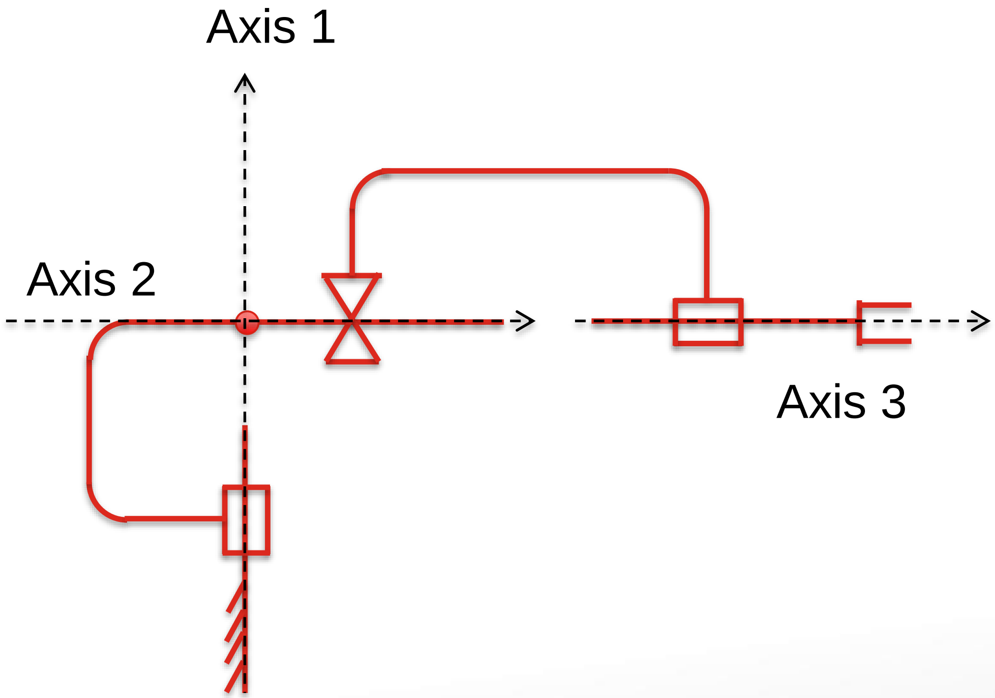


- Step 1, draw the axes.
    - For rotary joint: About the rotation
    - For prismatic joint: Along the translation

Example 2

- 3-link RPR robot:

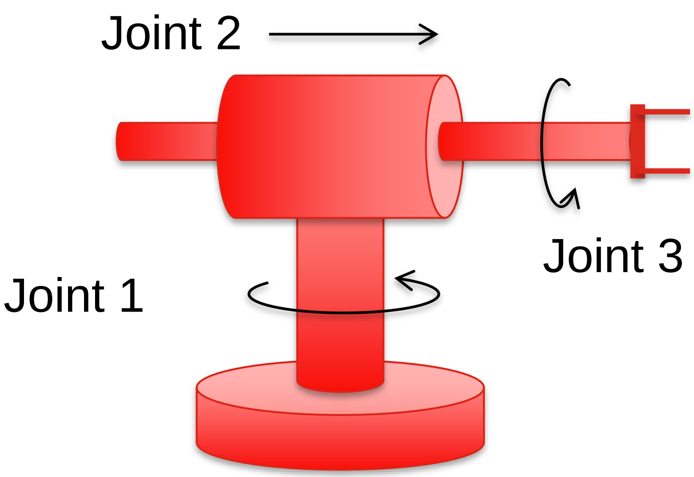


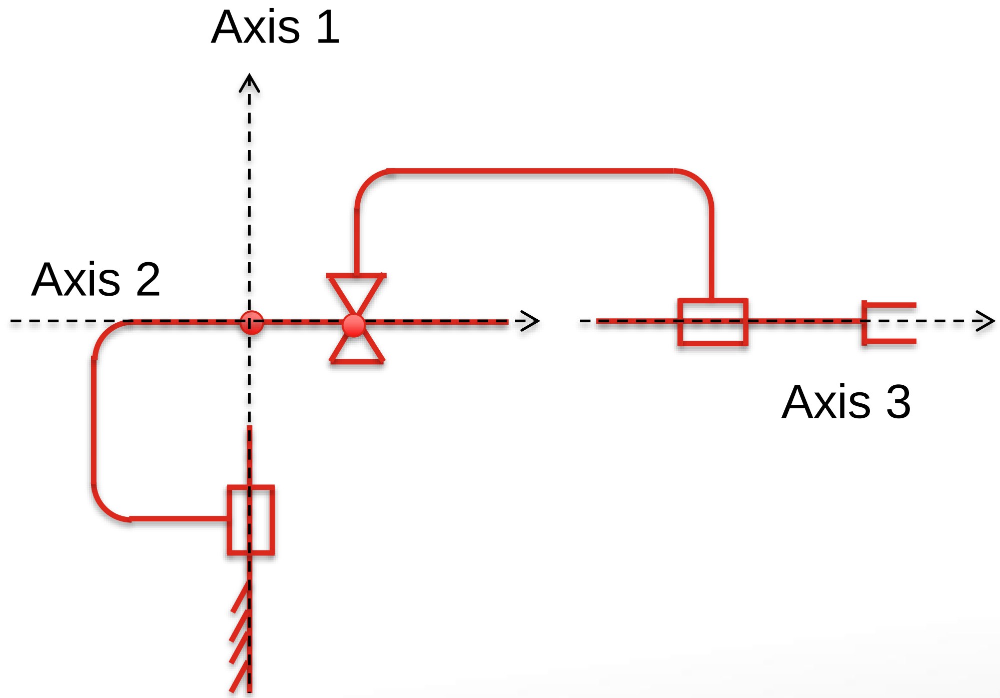


- Step 2, we draw the mutually perpendicular lines between axes.
  - Draw from lower axis to higher axis, e.g. 1 to 2, 2 to 3…
  - Axis 1 and Axis 2 intersect.
    - Line is perpendicular to the plane made by axes 1 and 2.
    - Because the axes intersect, the sense of direction is arbitrary (either into or out of the plane).

Example 2

- 3-link RPR robot:

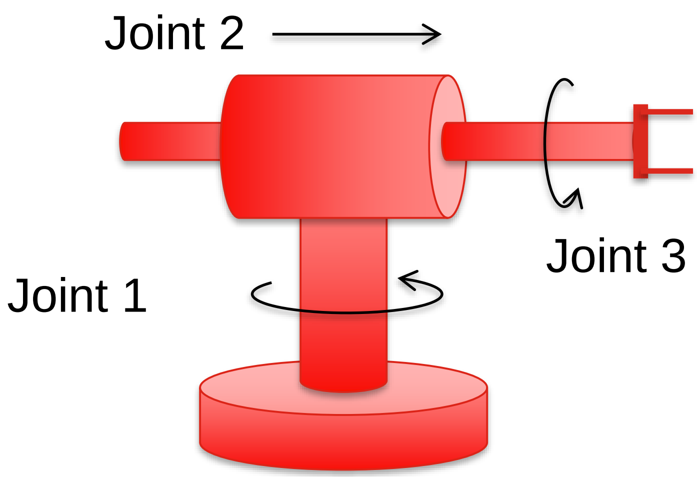


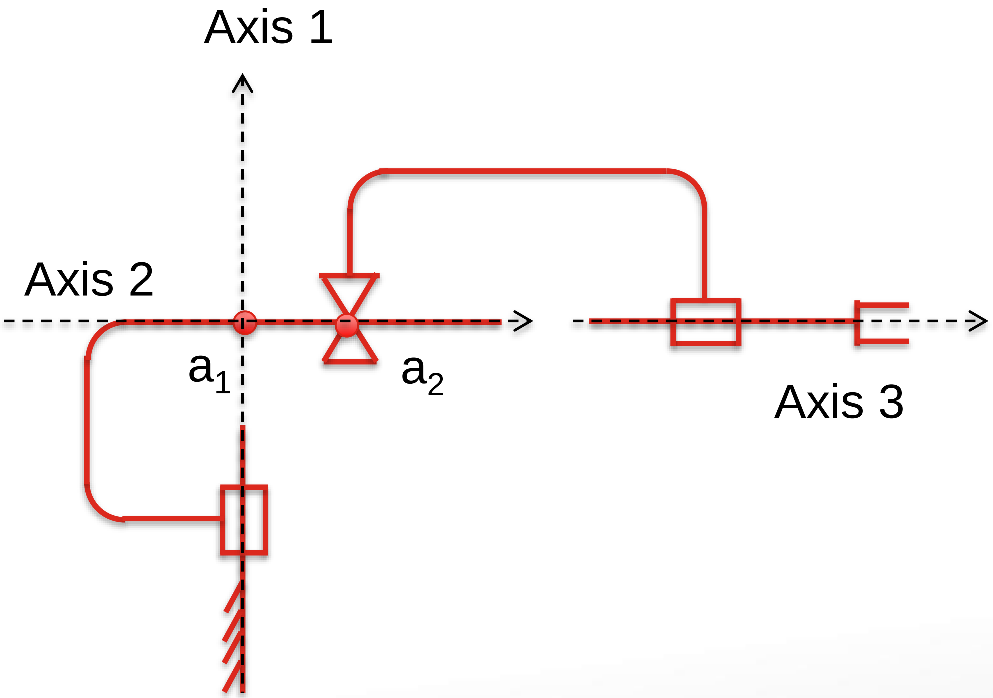


- Step 2, we draw the mutually perpendicular lines between axes.
  - Draw from lower axis to higher axis, e.g. 1 to 2, 2 to 3…
  - Axis 2 and Axis 3 intersect (and are the same).
    - Line is arbitrary. Put it at joint 2.

Example 2

- 3-link RPR robot:



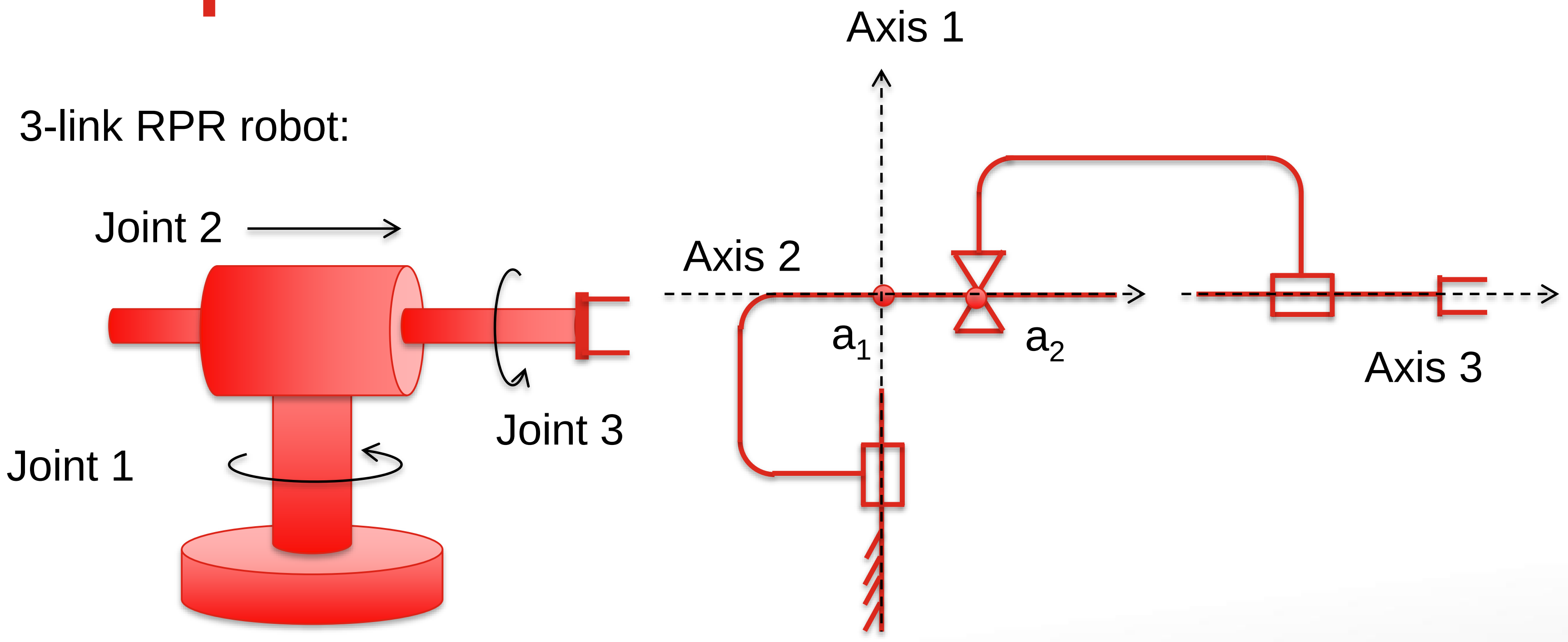


- Step 3, put in the link lengths a_{i-1}.
- $\boxed{\text{Definition: } a_{i-1} = \text{length of mutual perpendicular, from axis i-1 to axis i.}}$

Do 1 to n-1
- a_1 = length of mutual perpendicular from axis 1 to 2 = 0 (intersect).
- a_2 = length of mutual perpendicular from axis 2 to 3 = 0 (intersect).

Example 2

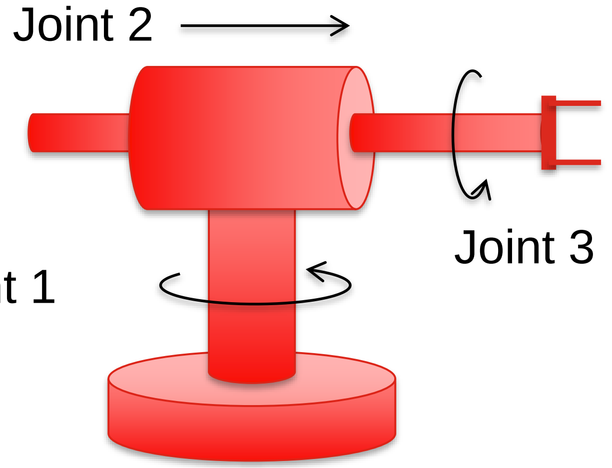
- 3-link RPR robot:

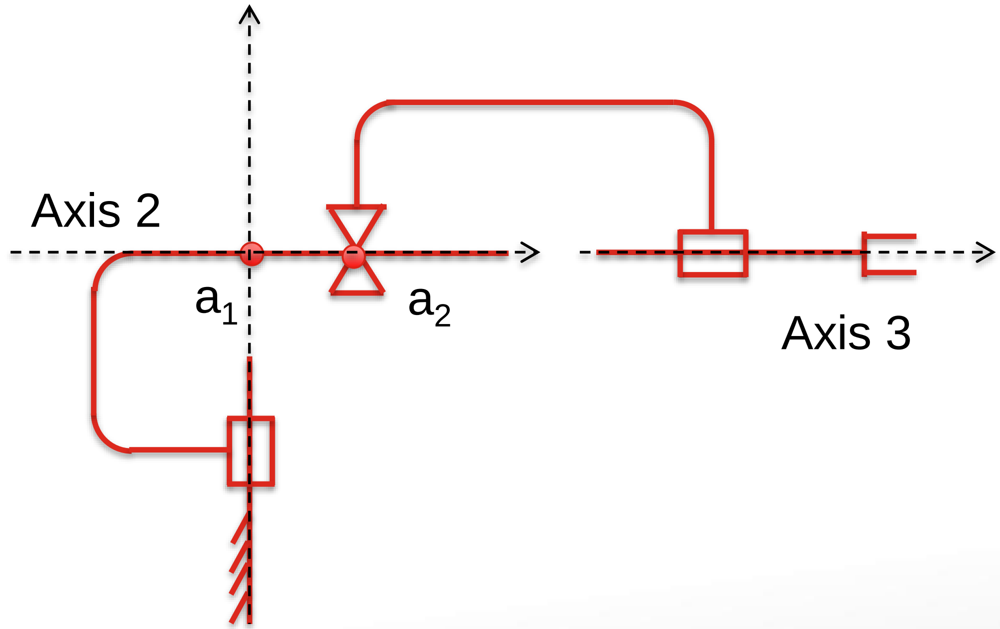


- What about a_0?
  - a_0 = length of mutual perpendicular from axis 0 to 1. However, axis 0 is not known yet.

Example 2

- 3-link RPR robot:

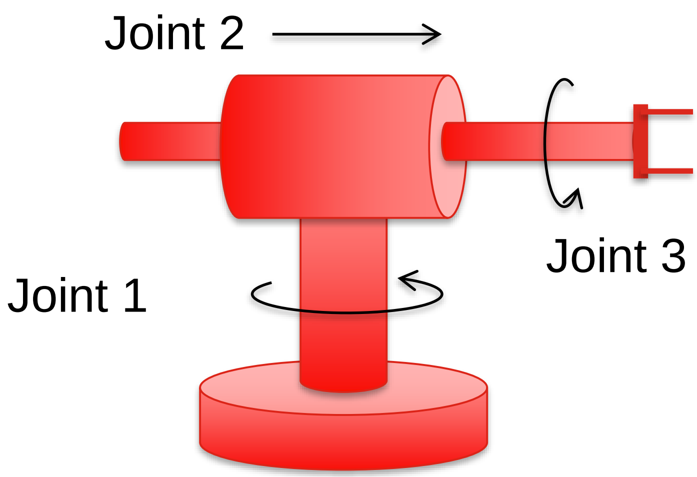


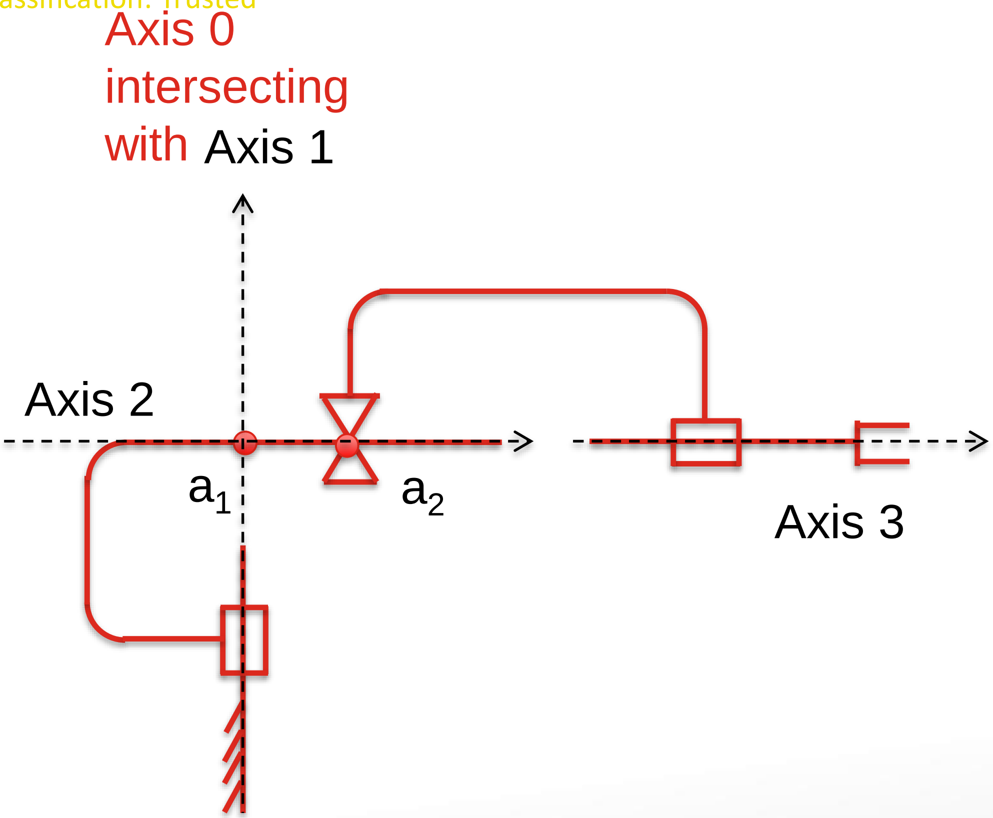


- What about a_0?
  - a_0 = length of mutual perpendicular from axis 0 to 1. However, axis 0 is not known yet.
  - By convention, $a_0 = 0$.
  - This means: Axis 0 and Axis 1 intersect with each other.

Example 2

- 3-link RPR robot:





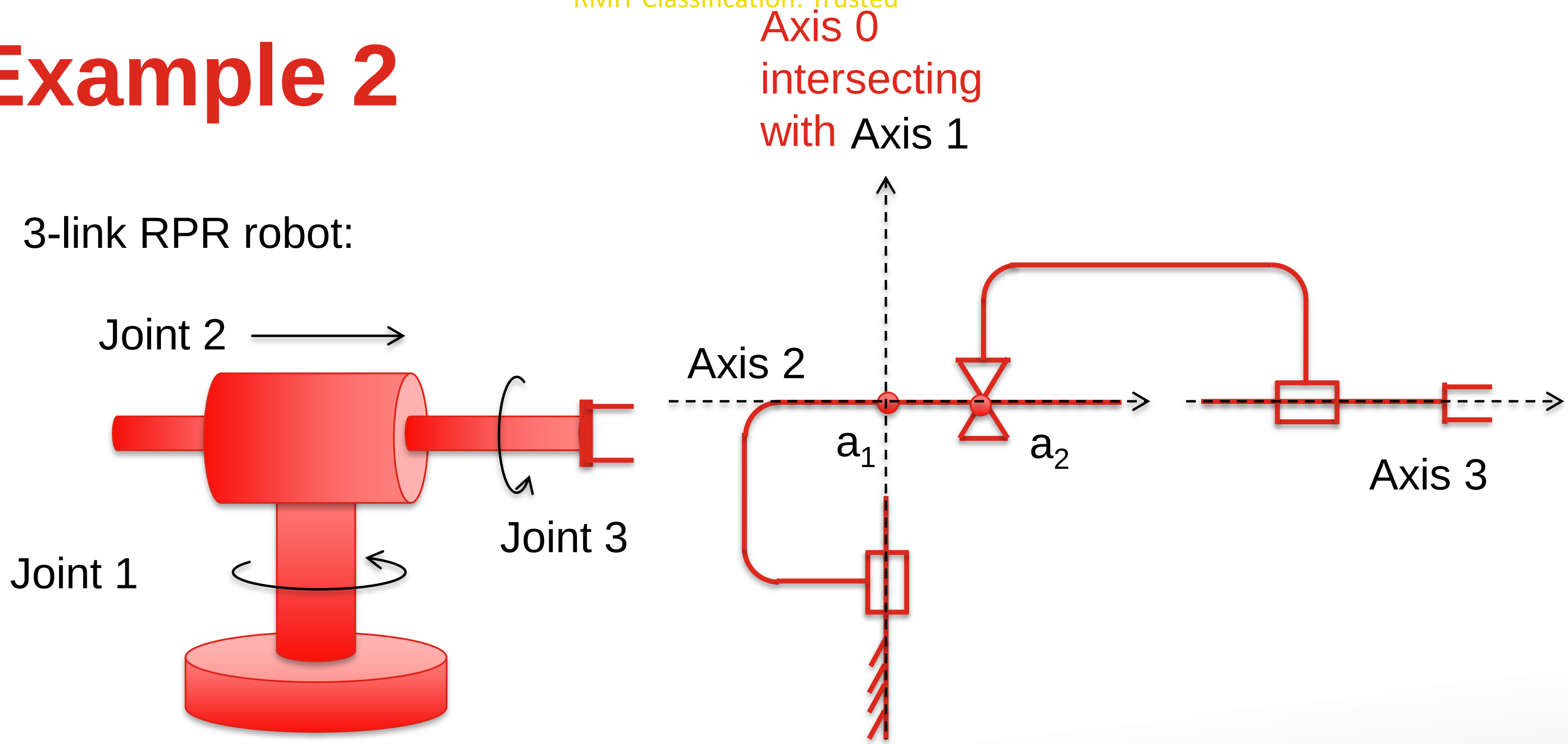
- Step 4, put in link twists α_{i-1}.
- Definition: α_{i-1} = angle between axis i-1 and axis i, in the right hand sense about a_{i-1}

Do 1 to n-1

- α_1 = angle between axis 1 and axis 2, about a_1 = -90deg.
- α_2 = angle between axis 2 and axis 3, about a_2 = 0deg.

Example 2

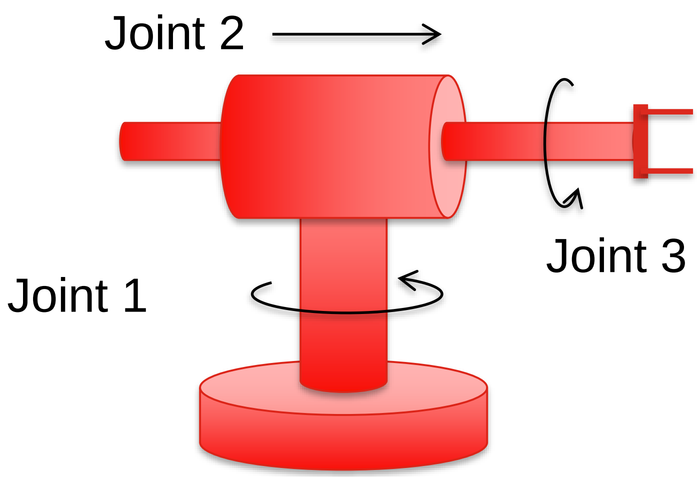
- 3-link RPR robot:

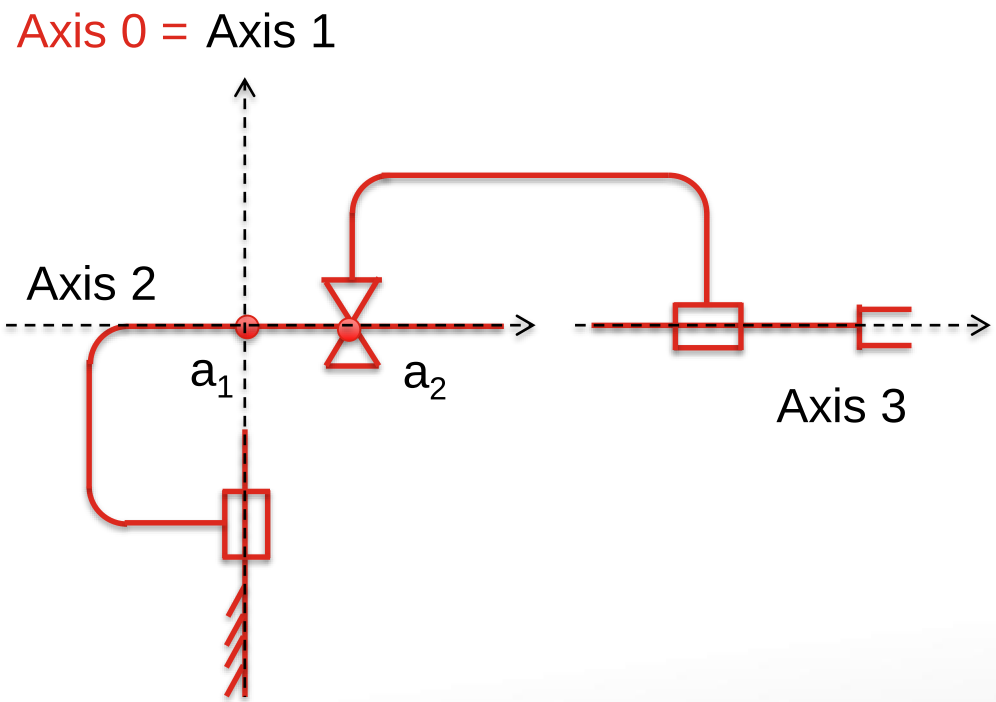


- What about α_0?
  - α_0 = angle between axis 0 and axis 1, about a_0 . However, axis 0 is not fully known yet.

Example 2

- 3-link RPR robot:

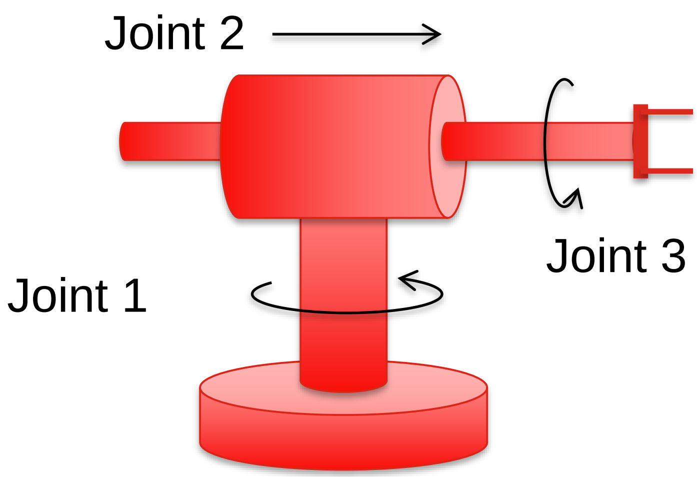


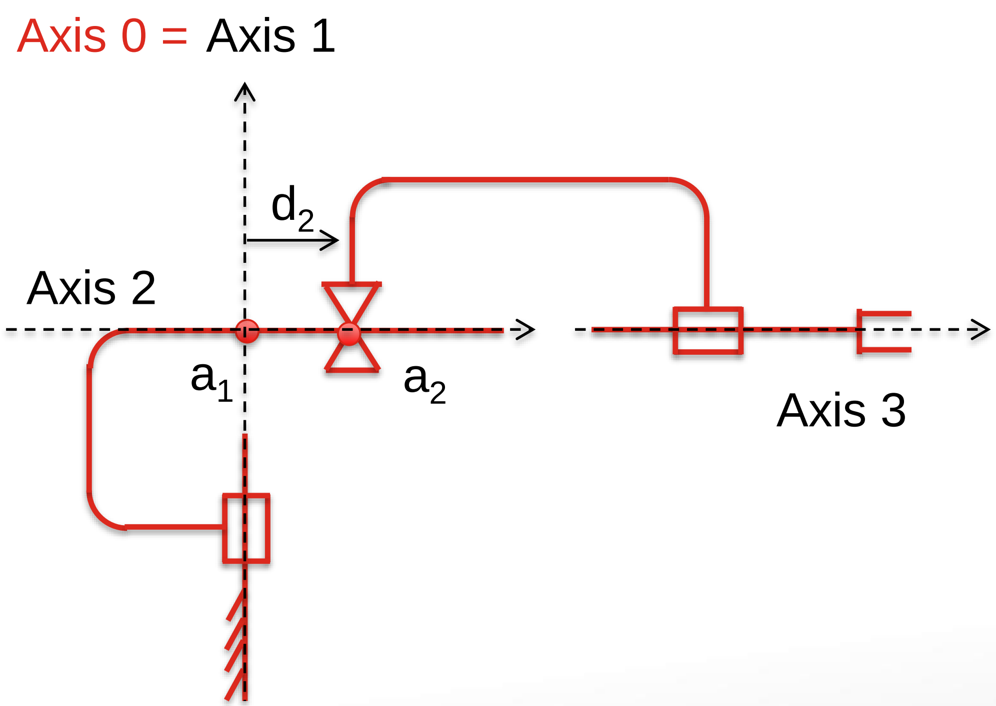


- What about α_0?
  - α_0 = angle between axis 0 and axis 1, about a_0 . However, axis 0 is not fully known yet.
  - By convention, $\alpha_0 = 0$.
  - This means: Axis 0 and Axis 1 are the same.

Example 2

- 3-link RPR robot:





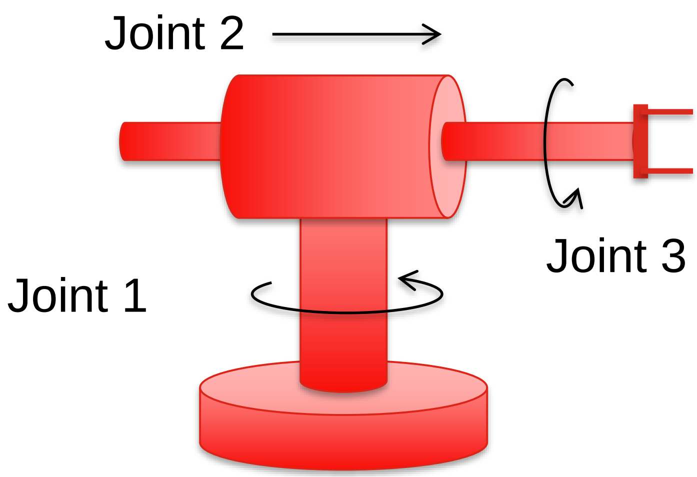
- Step 5, write down the link offsets d_i.
- $\boxed{\text{Definition: } d_i = \text{distance from } a_{i-1} \text{ to } a_i \text{, along axis i.}}$

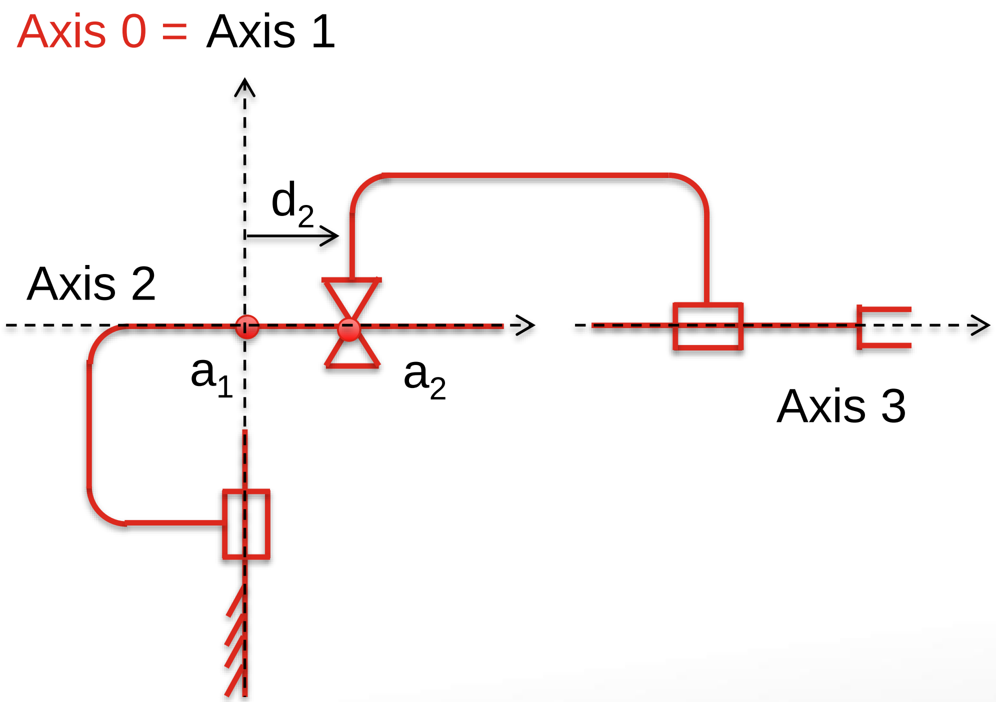
Do 2 to n-1

- d_2 = distance from a1 to a2, along axis 2, is a variable (prismatic)!

Example 2

- 3-link RPR robot:

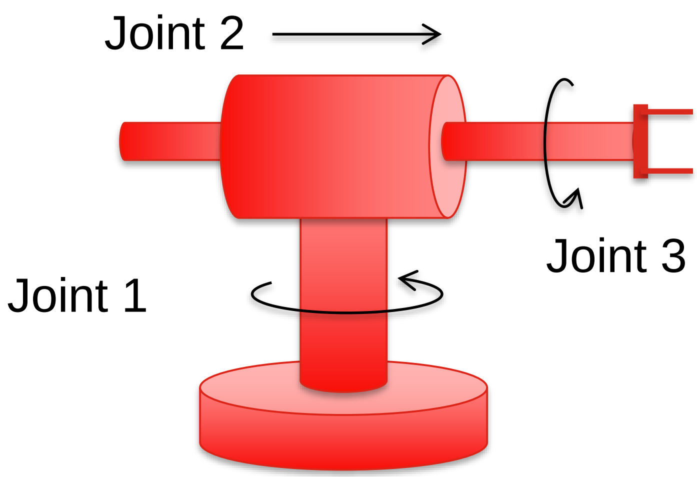


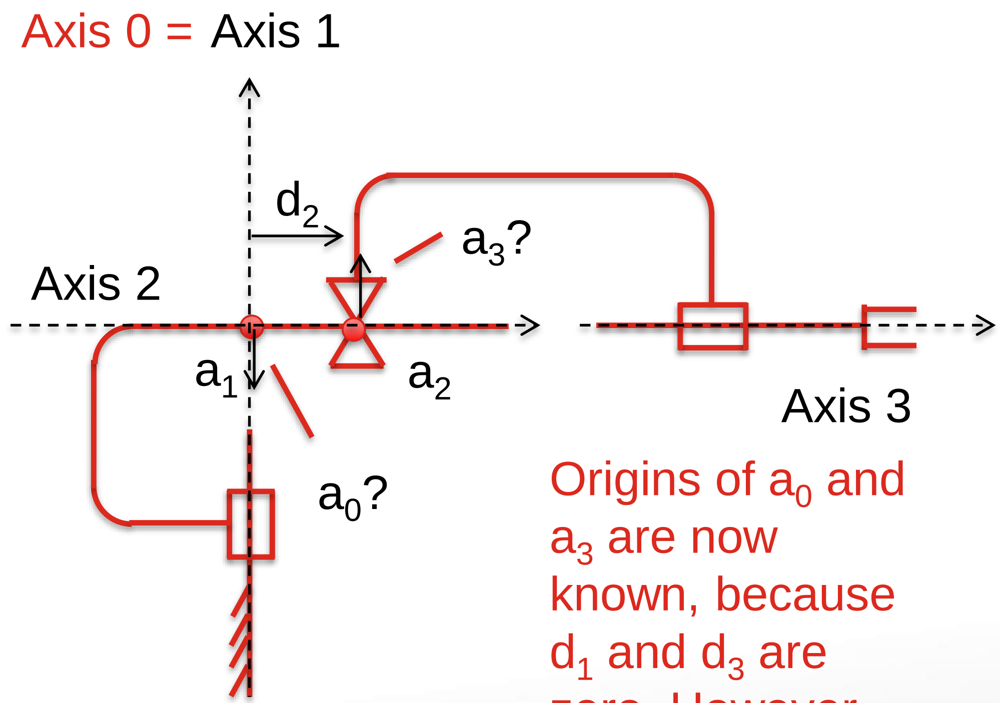


- What about d_1 and d_3?
  - d_1 = distance from a_0 to a_1, along axis 1.
  - d_3 = distance from a_2 to a_3, along axis 3.
  - But where exactly are a_0 and a_3?

Example 2

- 3-link RPR robot:



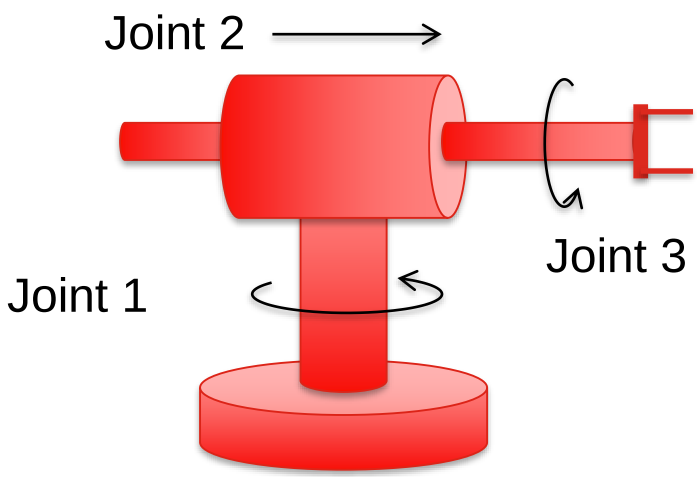


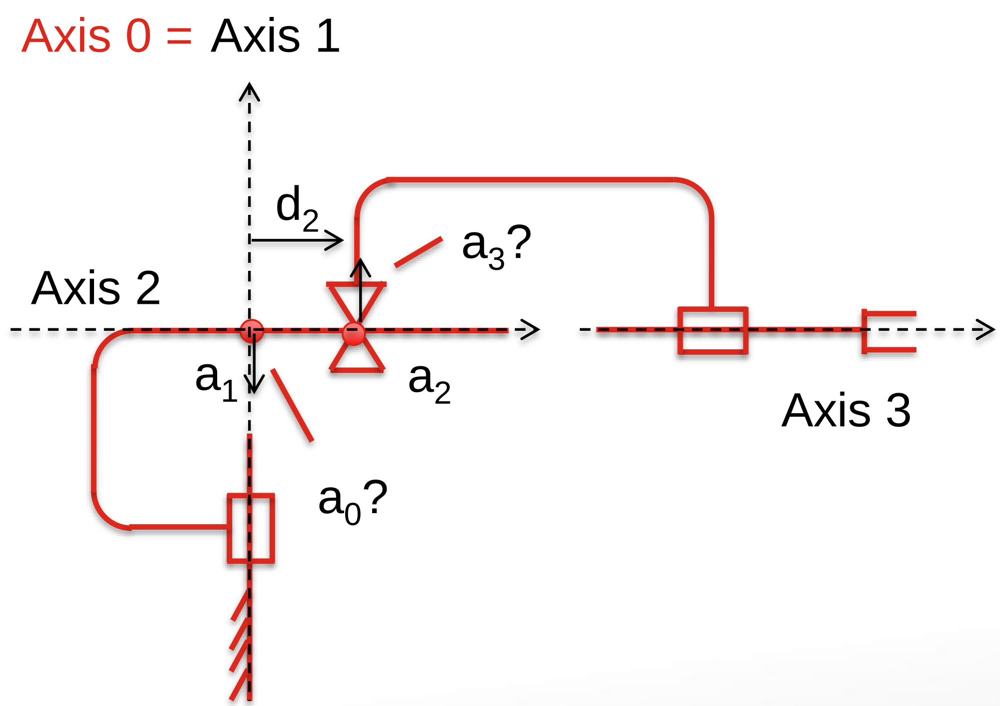
Origins of a_0 and a_3 are now known, because d_1 and d_3 are zero. However, the direction is not known.

- What about d_1 and d_3?
  - d_1 = distance from a_0 to a_1, along axis 1.
  - d_3 = distance from a_2 to a_3, along axis 3.
  - By convention: Zero for revolute joint, variable for prismatic joint
  - So in this case, d_1 and d_3 are both zero.

Example 2

- 3-link RPR robot:





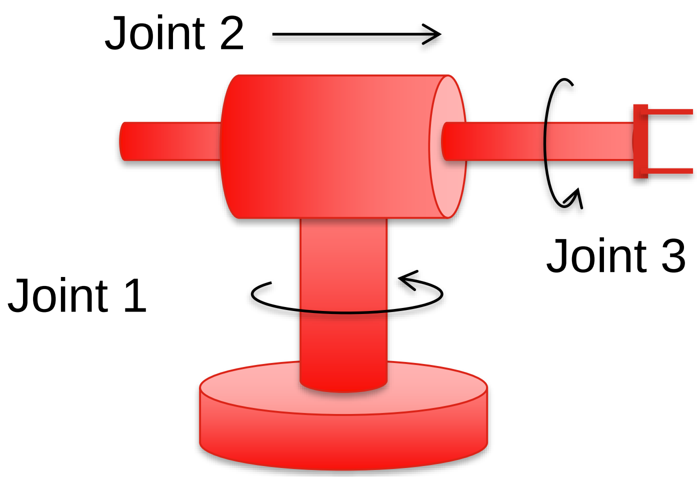
- Step 6, write down the joint angle θ_i.
- Definition: θ_i is the angle between the (extension of a_{i-1}) and a_i, measured about the axis i.

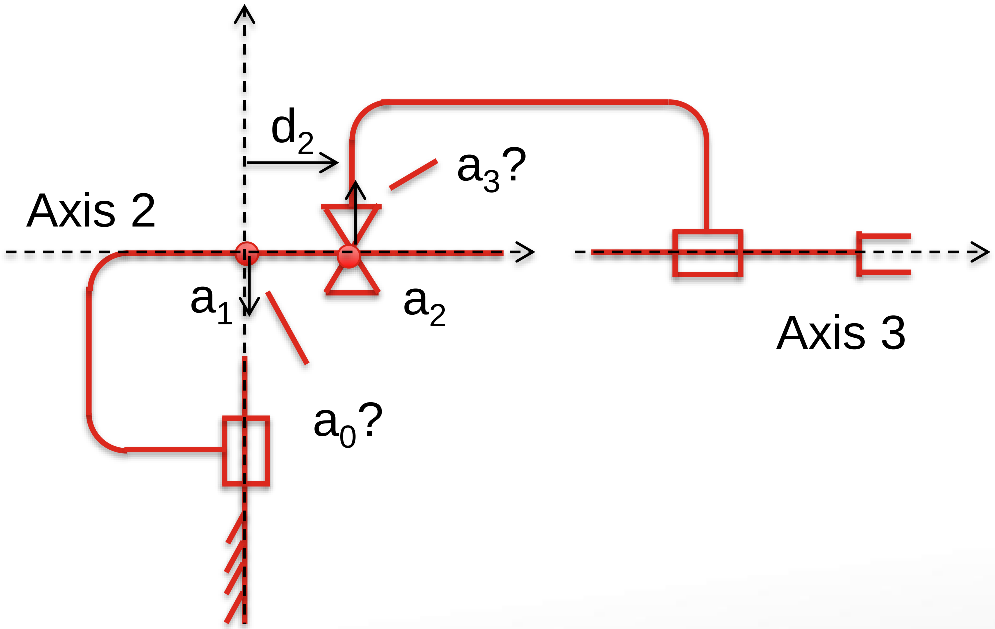
Do 2 to n-1

- θ_2 = angle between (extension of a1) and a2, about axis 2.
- Here, θ_2 is zero (a constant).

Example 2

- 3-link RPR robot:

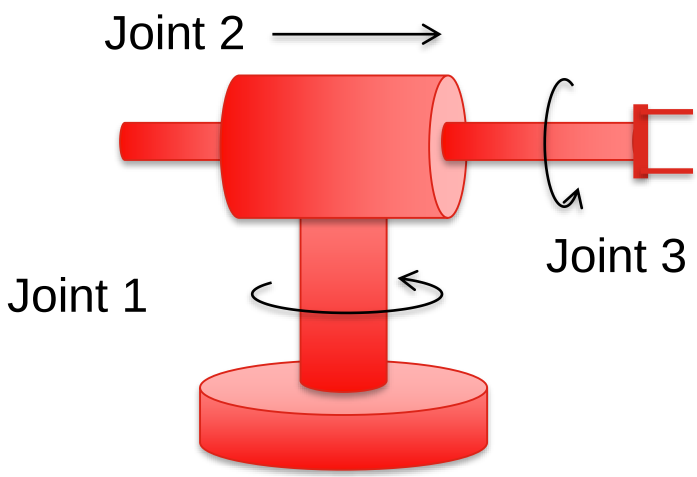


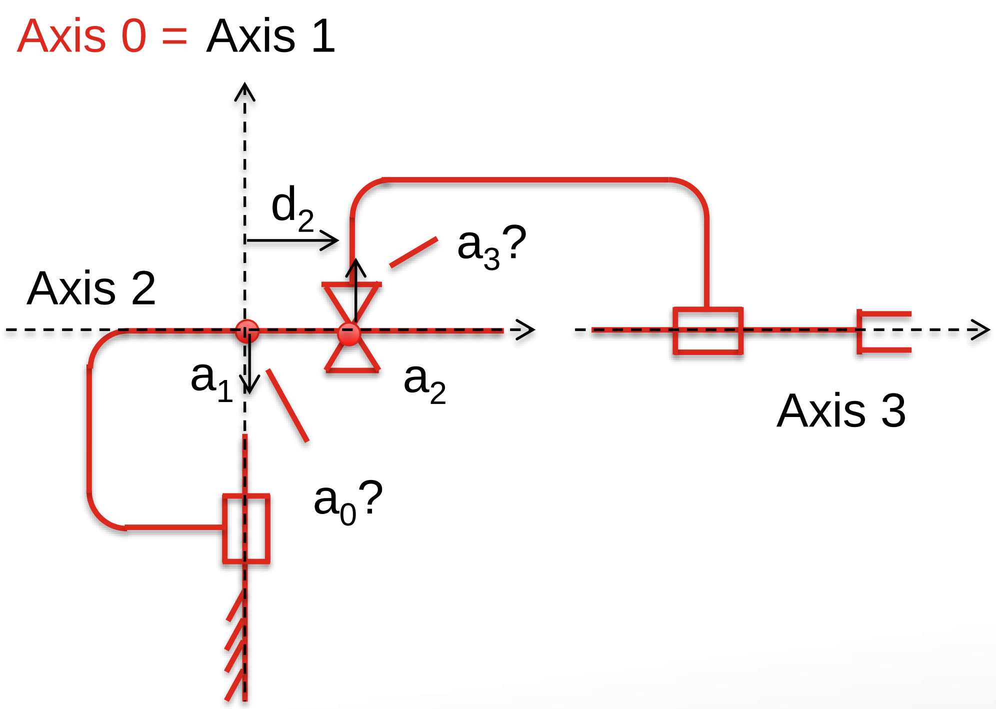


- What about θ_1 and θ_3?
  - θ_1 = angle between (extension of a0) and a1, about axis 1.
  - θ_3 = angle between (extension of a2) and a3, about axis 3.
  - By convention: Zero for prismatic joint, variable for revolute joint
  - So in this case, θ_1 and θ_3 are both variables.

Example 2

- 3-link RPR robot:

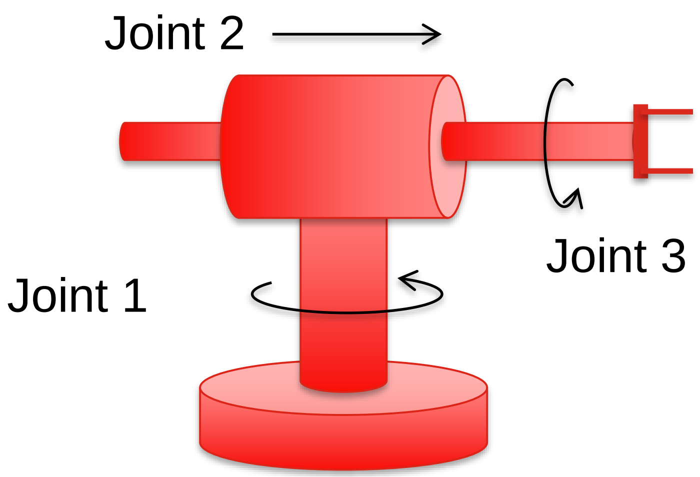


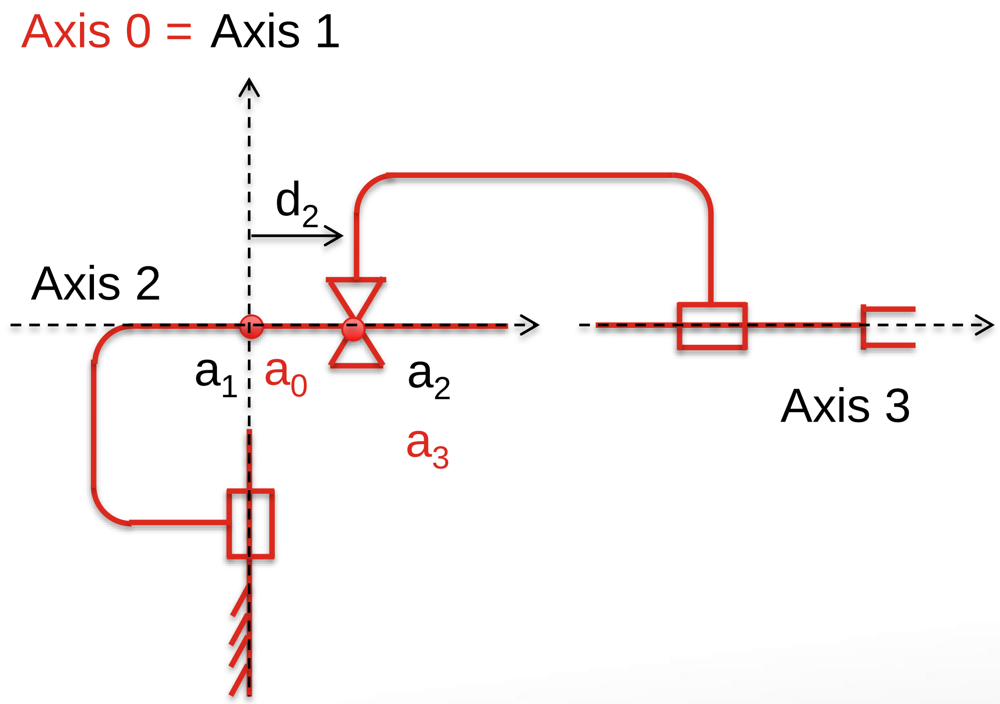


- We still have a problem. Since θ_1 and θ_3 are both variables, we need to determine their “zero”-angle position.

Example 2

- 3-link RPR robot:

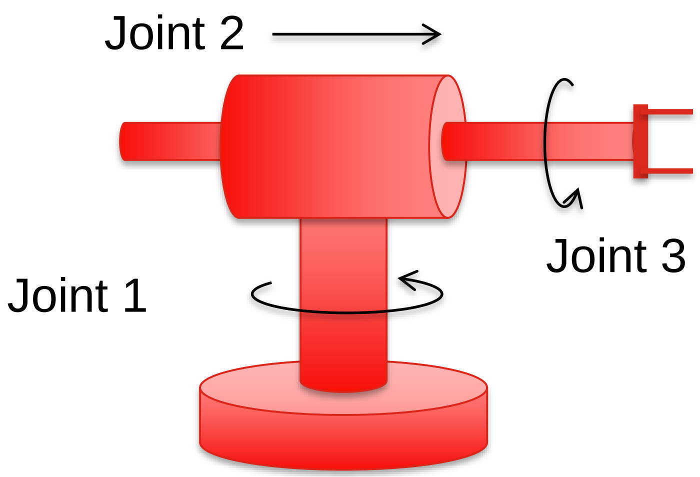


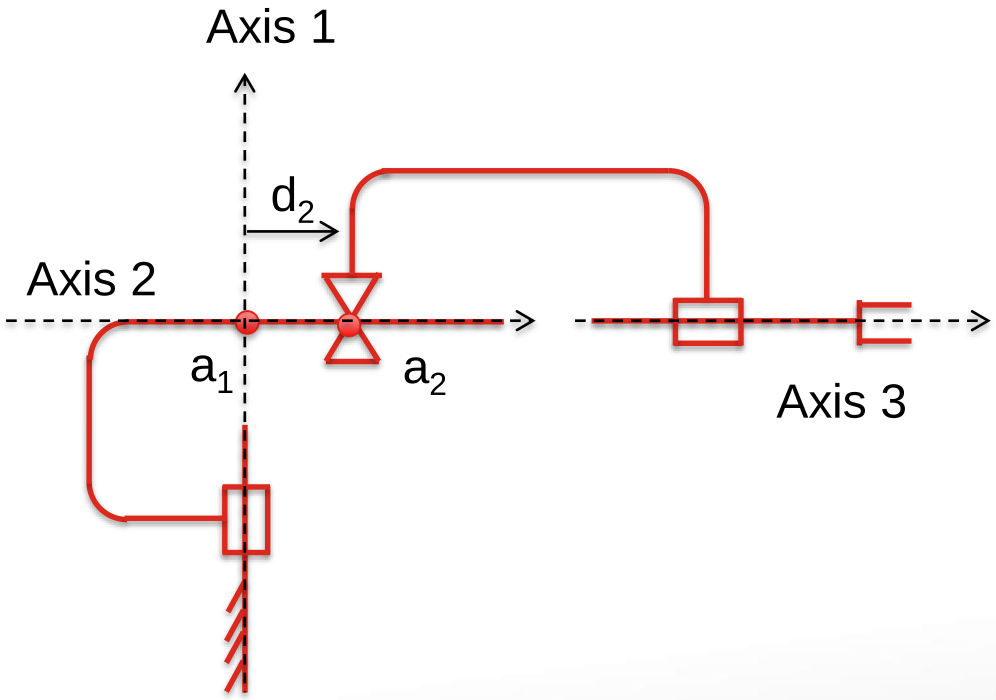


- For convenience, align a_0 with a_1 when the joint variable 1 is zero.
- As for joint n:
  - Revolute: align a_n with a_{n-1} when $\theta_n = 0$.
  - Prismatic: align a_n with a_{n-1} when $d_n = 0$.

Example 2

- 3-link RPR robot:



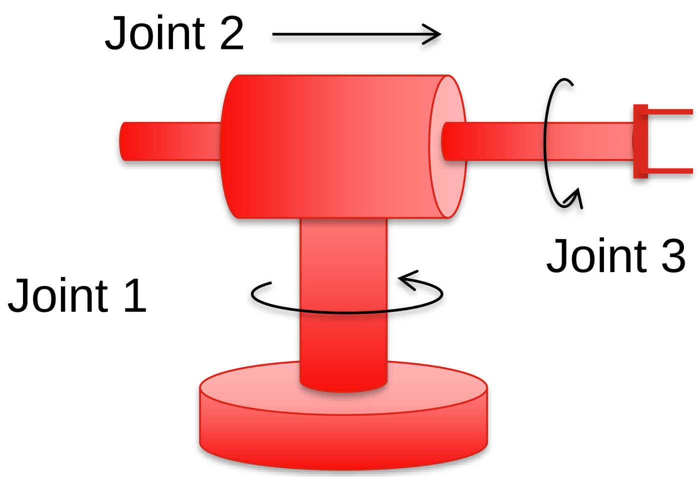


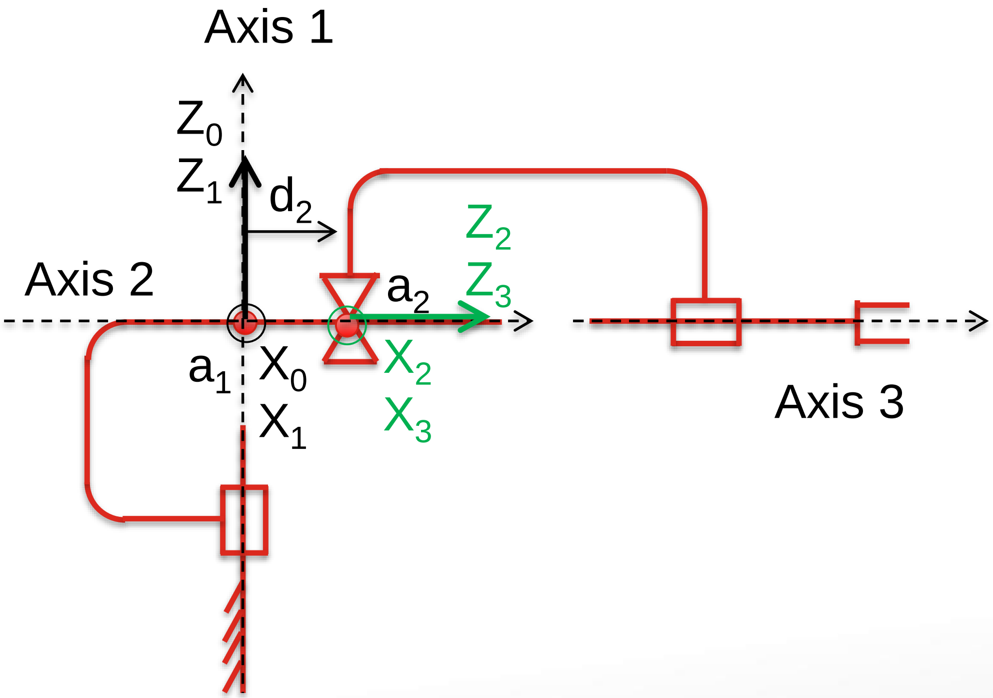
- Step 7, transfer to a DH-table:

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	-90	0	d_2	0
3	0	0	0	θ_3

Example 2

- 3-link RPR robot:





- Step 8, insert the frames. Rules:
  - Z-axis of frame {i}, i.e. Z_i, is coincident with joint axis i.
  - Origin of frame {i} is where the a_i intersects the joint i axis.
  - X-axis of frame {i}, i.e. X_i, is coincident with a_i.

Example 2

- Step 9 (Final step!), calculate the transformations.

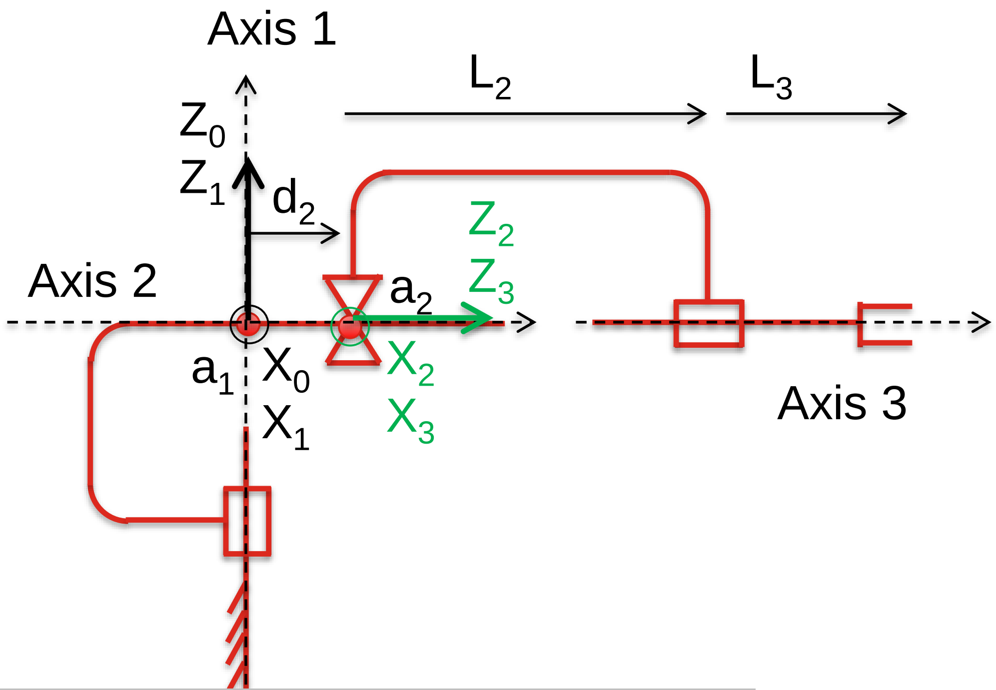
i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	-90	0	d_2	0
3	0	0	0	θ_3

$$ {}^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ c\alpha_{i-1}s\theta_i & c\alpha_{i-1}c\theta_i & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\alpha_{i-1}s\theta_i & s\alpha_{i-1}c\theta_i & c\alpha_{i-1} & c\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{0}_{1}T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{1}_{2}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{2}_{3}T = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & 0 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ \Rightarrow \quad {}^{0}_{3}T = {}^{0}_{1}T \cdot {}^{1}_{2}T \cdot {}^{2}_{3}T = \begin{bmatrix} c\theta_1 c\theta_3 & -c\theta_1 s\theta_3 & -s\theta_1 & -d_2 s\theta_1 \\ s\theta_1 c\theta_3 & -s\theta_1 s\theta_3 & c\theta_1 & d_2 c\theta_1 \\ -s\theta_3 & -c\theta_3 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Example 2



- Verification:
- The end-effector, with reference to frame {3}, has position $[0, 0, L_2+L_3]^T$, and same orientation as frame {3}. Therefore:

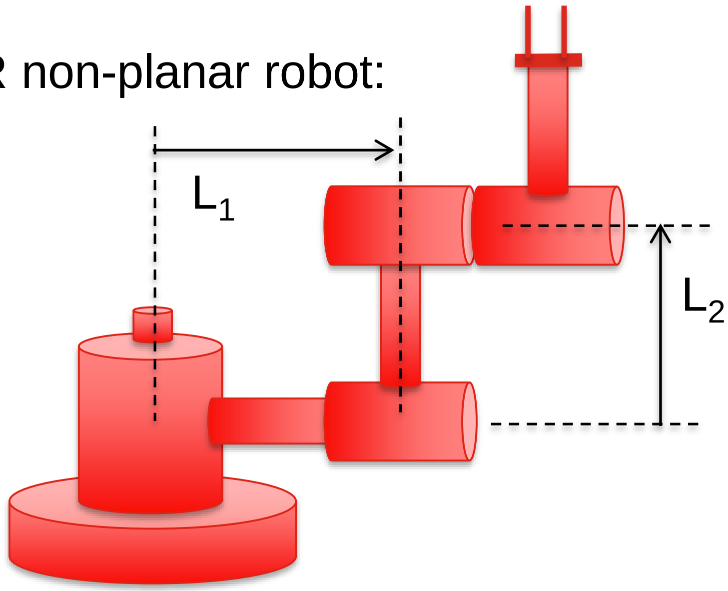
$$
{}^{0}P = {}^{0}_{3}T \cdot {}^{3}P
$$

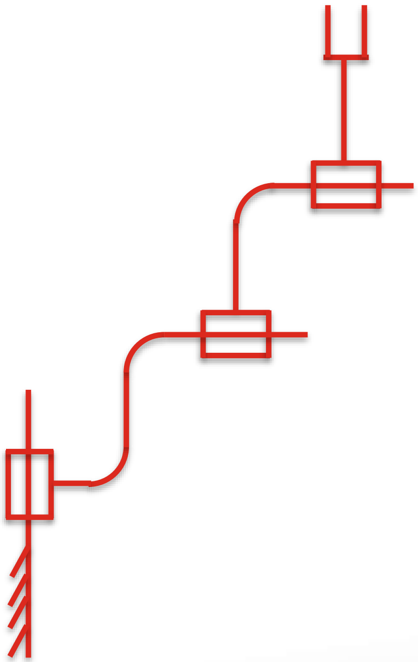
$$
= \begin{bmatrix} c\theta_1 c\theta_3 & -c\theta_1 s\theta_3 & -s\theta_1 & -d_2 s\theta_1 \\ s\theta_1 c\theta_3 & -s\theta_1 s\theta_3 & c\theta_1 & d_2 c\theta_1 \\ -s\theta_3 & -c\theta_3 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ L_2 + L_3 \\ 1 \end{bmatrix} = \begin{bmatrix} -(L_3 + L_2 + d_2) s\theta_1 \\ (L_3 + L_2 + d_2) c\theta_1 \\ 0 \\ 1 \end{bmatrix}
$$

- This makes sense!
  - e.g. $\theta_1 = 0$.

Example 3

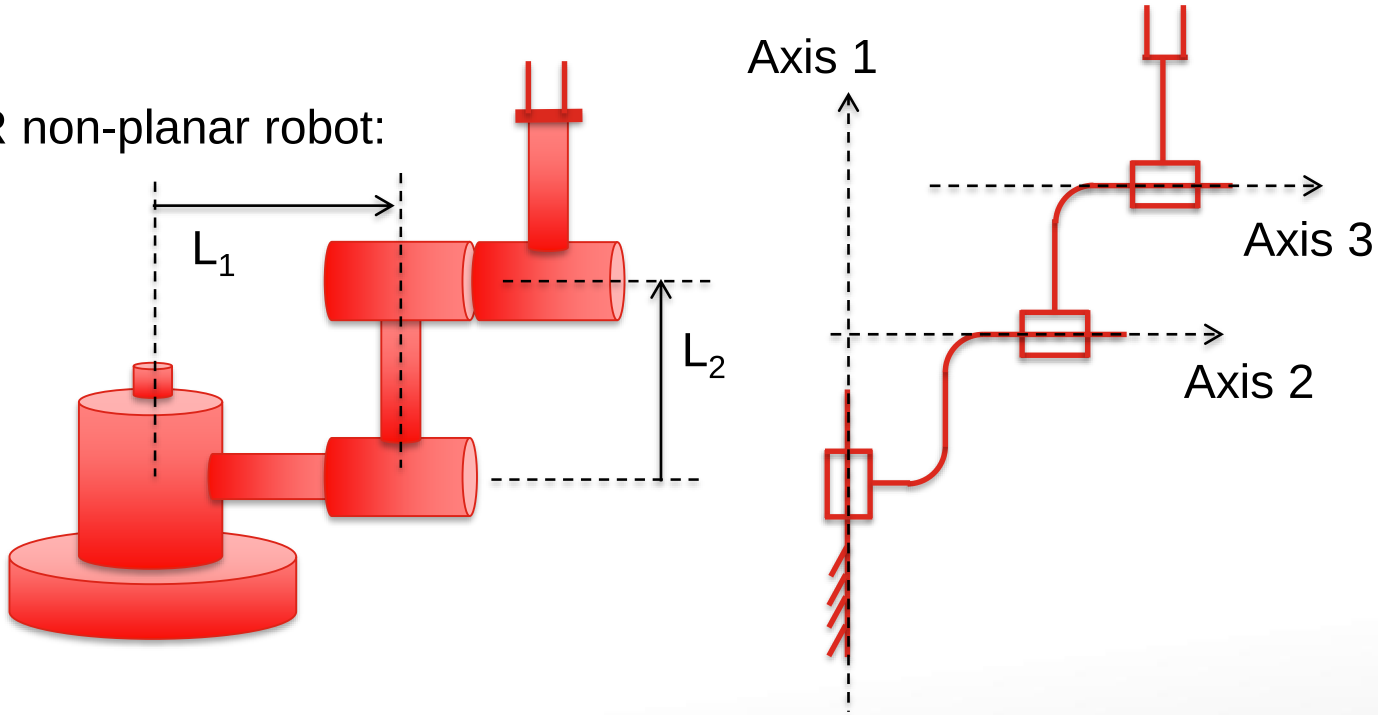
- 3-link RRR non-planar robot:





Example 3

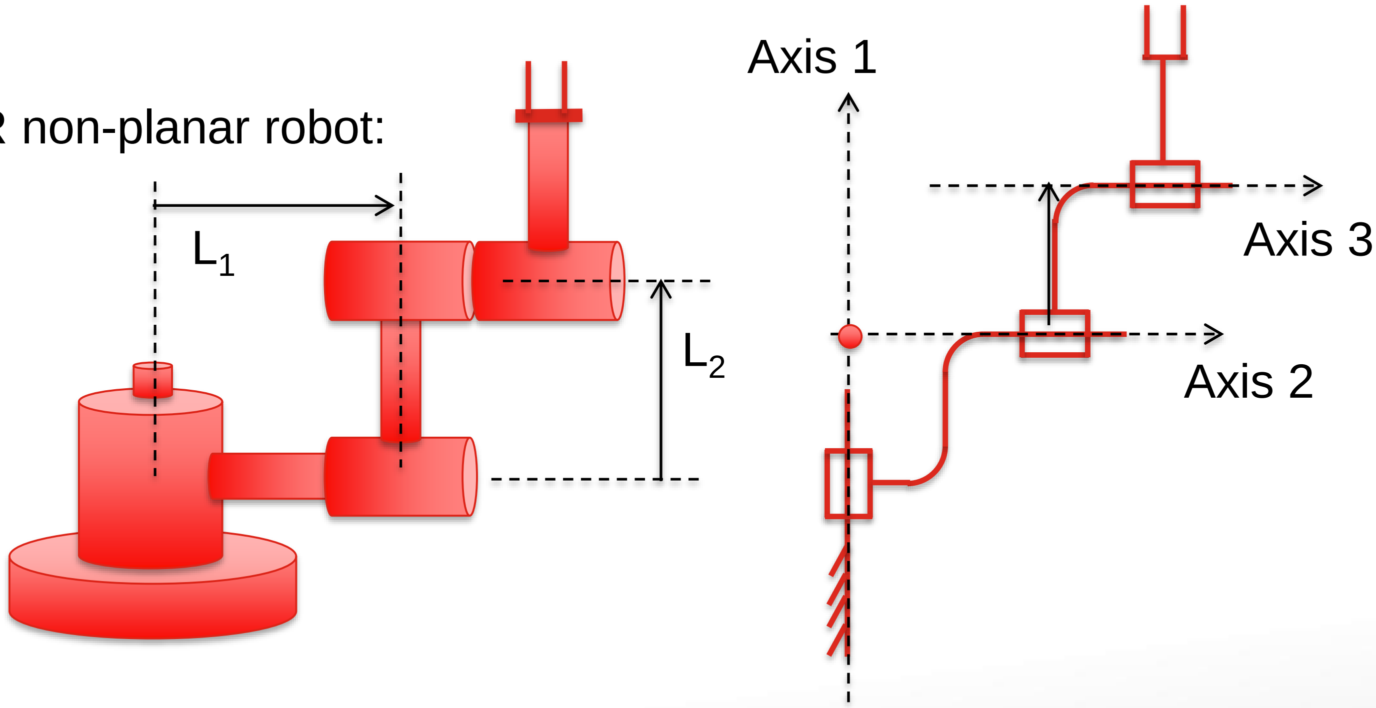
- 3-link RRR non-planar robot:



- Step 1, draw the axes.
  - For rotary joint: About the rotation
  - For prismatic joint: Along the translation

Example 3

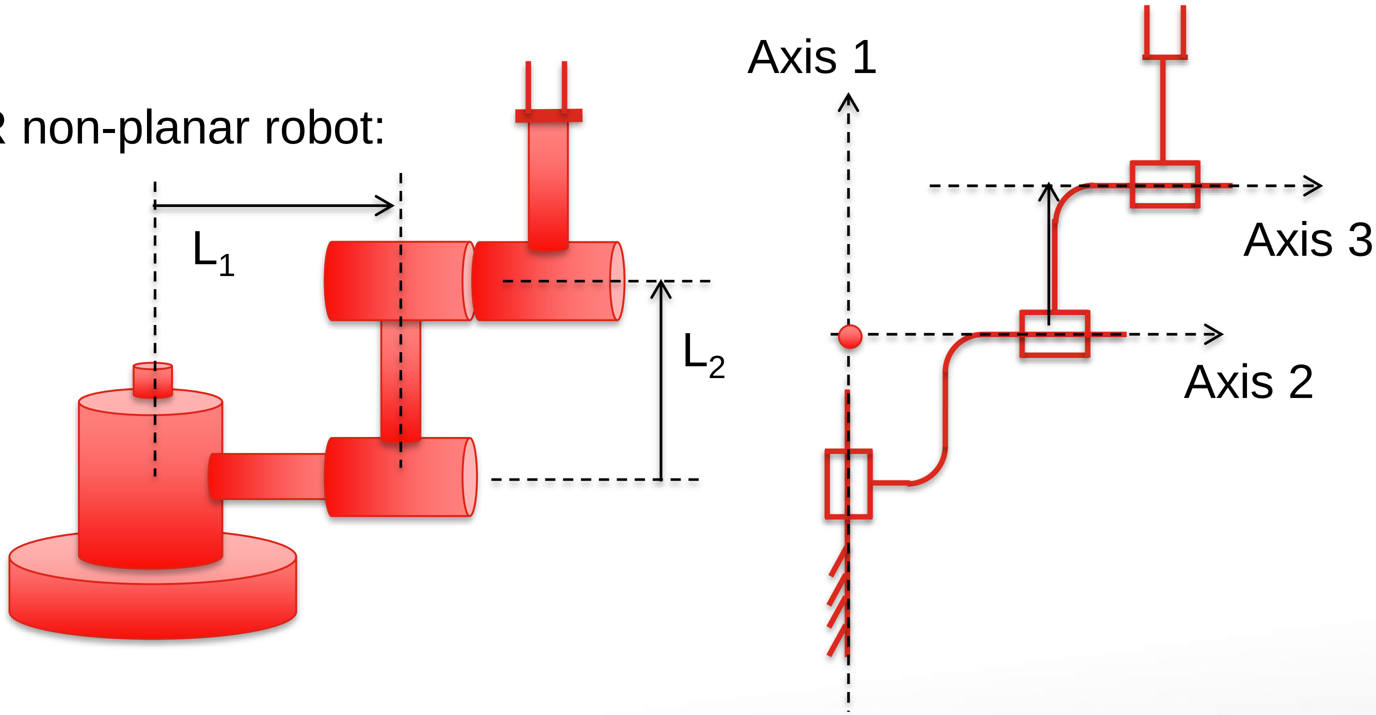
- 3-link RRR non-planar robot:



- Step 2, we draw the mutually perpendicular lines between axes.
  - Draw from lower axis to higher axis, e.g. 1 to 2, 2 to 3…
  - Axis 1 and Axis 2 intersect.
    - Line is perpendicular to the plane made by axes 1 and 2.
    - Because the axes intersect, the sense of direction is arbitrary (either into or out of the plane).

Example 3

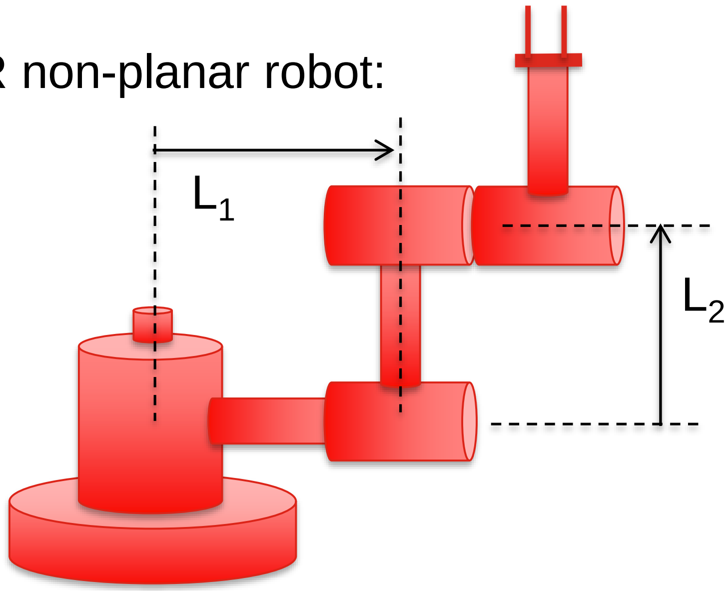
- 3-link RRR non-planar robot:

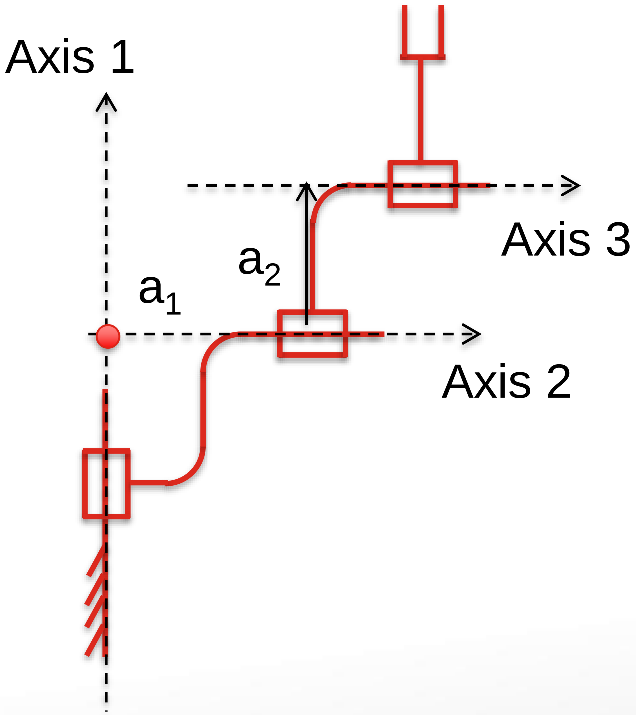


- Step 2, we draw the mutually perpendicular lines between axes.
  - Draw from lower axis to higher axis, e.g. 1 to 2, 2 to 3…
  - Axis 2 and Axis 3 are parallel.
    – Black arrow as shown.

Example 3

- 3-link RRR non-planar robot:





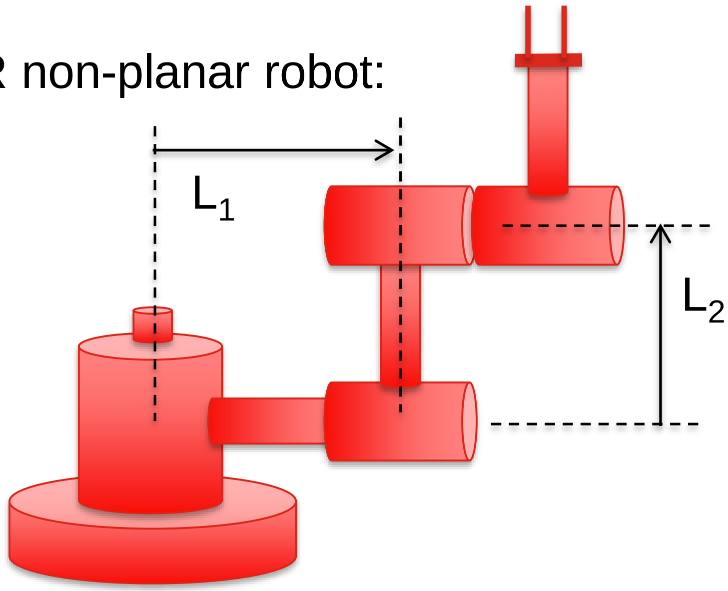
- Step 3, put in the link lengths a_{i-1}.
- Definition: a_{i-1} = length of mutual perpendicular, from axis i-1 to axis i.

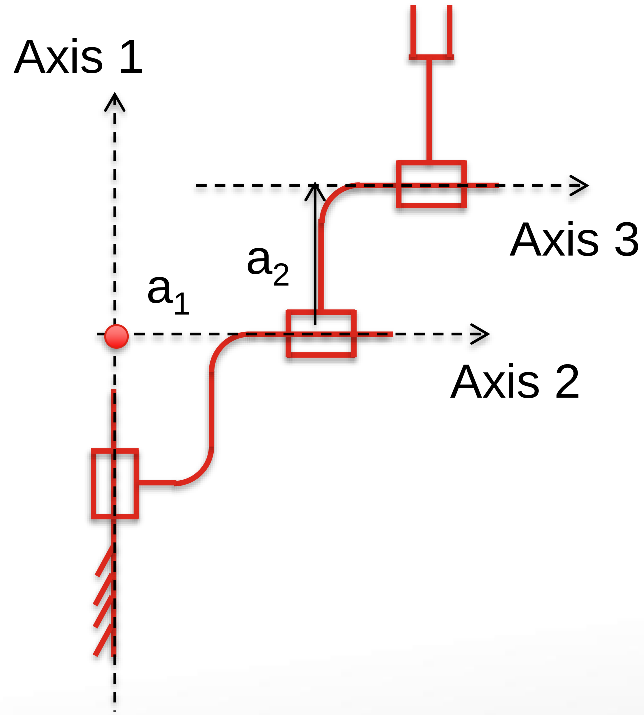
Do 1 to n-1

- a_1 = length of mutual perpendicular from axis 1 to 2 = 0 (intersect).
- a_2 = length of mutual perpendicular from axis 2 to 3 = L_2 .

Example 3

- 3-link RRR non-planar robot:

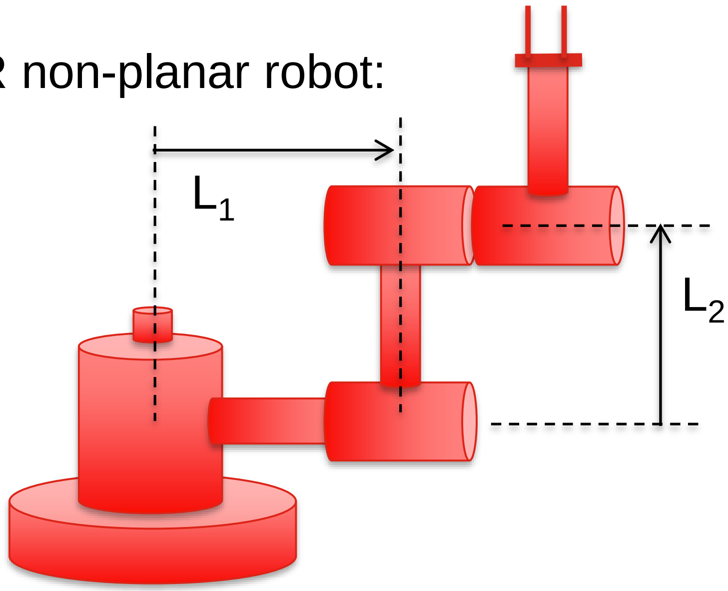


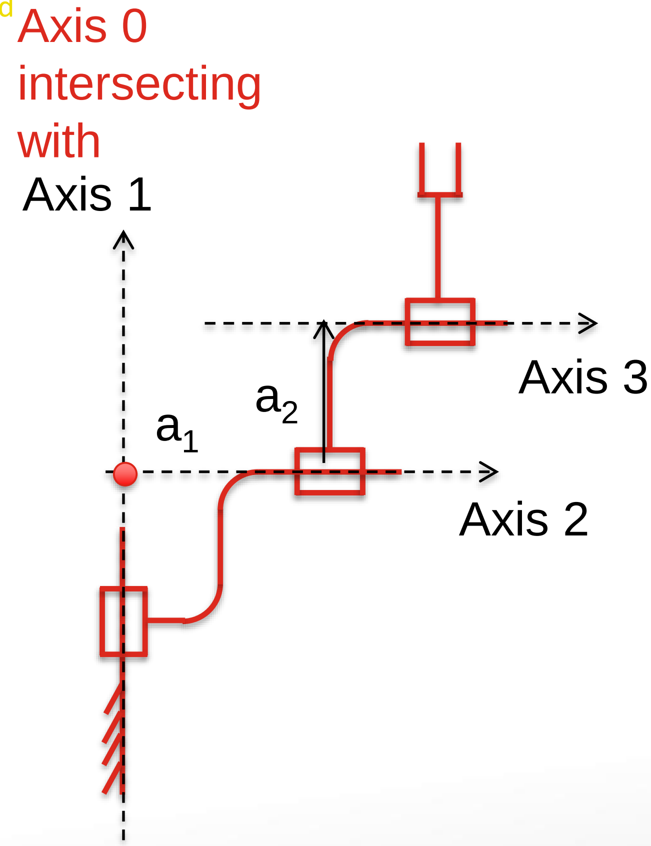


- What about a_0?
  - a_0 = length of mutual perpendicular from axis 0 to 1. However, axis 0 is not known yet.

Example 3

- 3-link RRR non-planar robot:

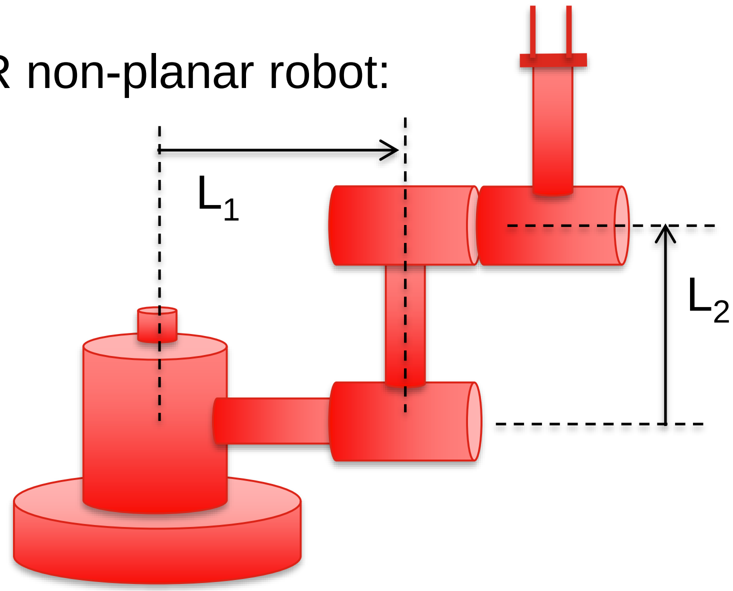


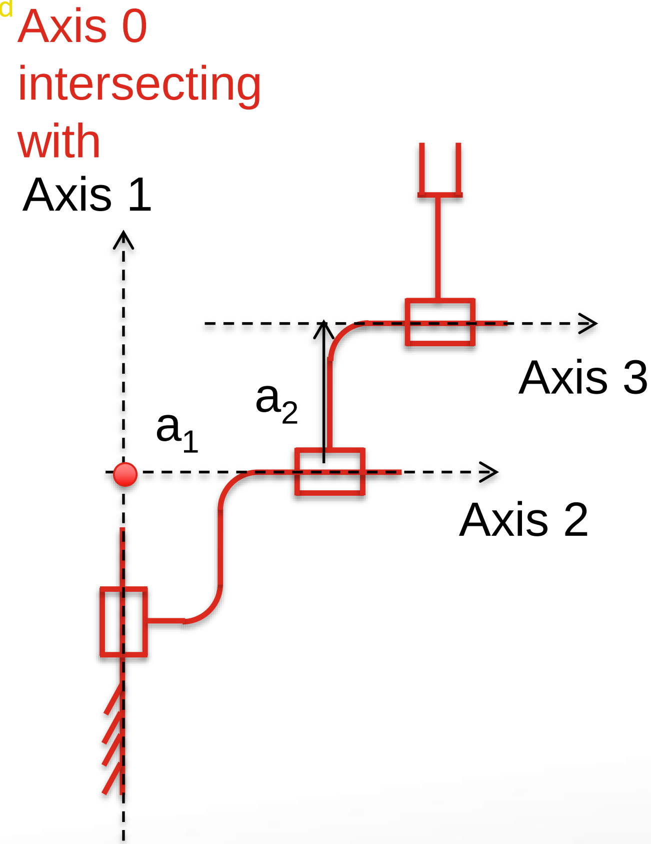


- What about a_0?
  - a_0 = length of mutual perpendicular from axis 0 to 1. However, axis 0 is not known yet.
  - By convention, $a_0 = 0$.
  - This means: Axis 0 and Axis 1 intersect with each other.

Example 3

- 3-link RRR non-planar robot:





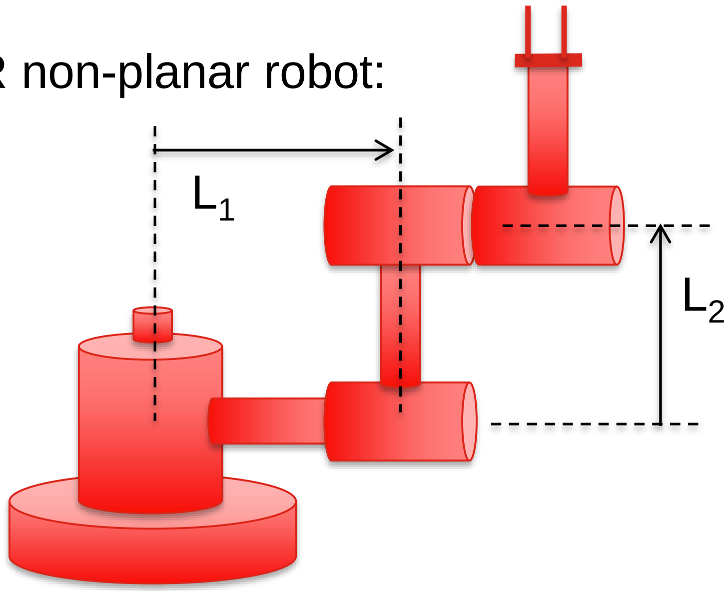
- Step 4, put in link twists α_{i-1}.
- Definition: α_{i-1} = angle between axis i-1 and axis i, in the right hand sense about a_{i-1}

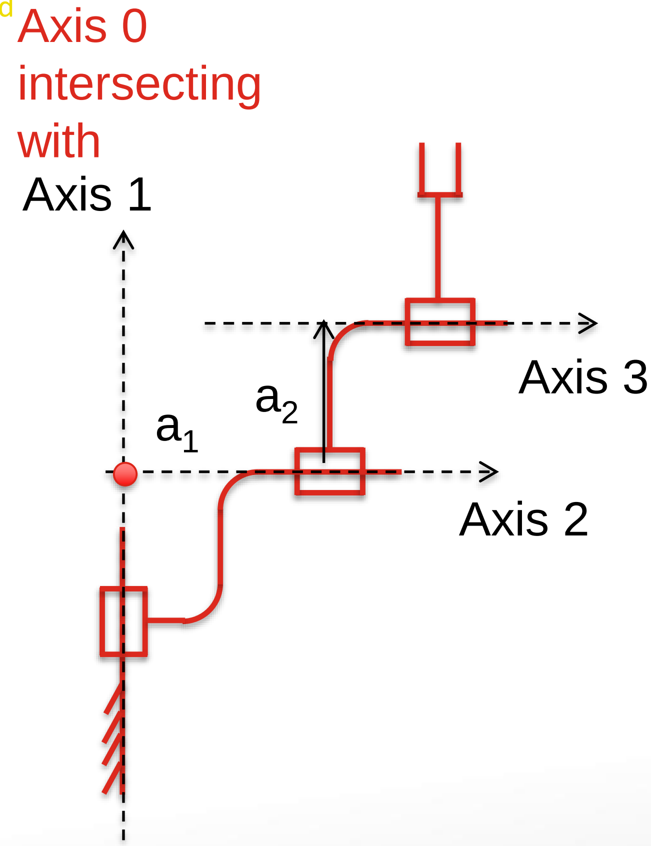
Do 1 to n-1

- α_1 = angle between axis 1 and axis 2, about a_1 = -90deg.
- α_2 = angle between axis 2 and axis 3, about a_2 = 0deg.

Example 3

- 3-link RRR non-planar robot:

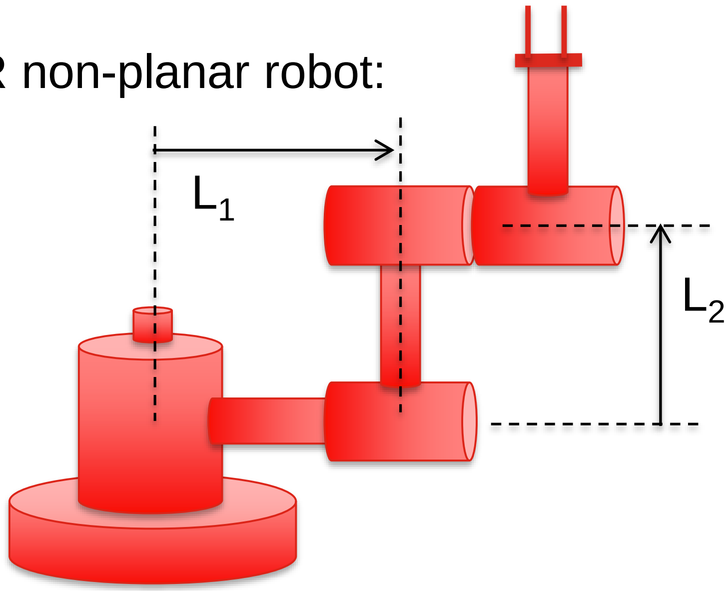


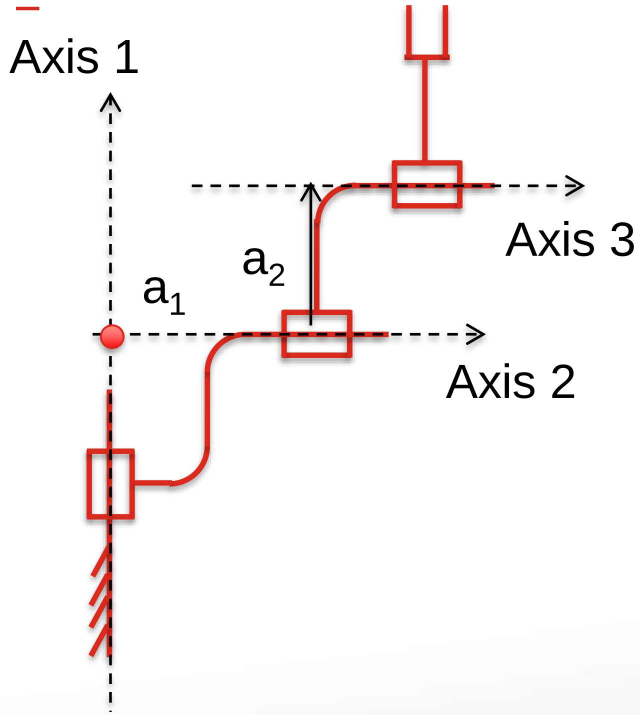


- What about α_0?
  - α_0 = angle between axis 0 and axis 1, about a_0 . However, axis 0 is not fully known yet.

Example 3

- 3-link RRR non-planar robot:

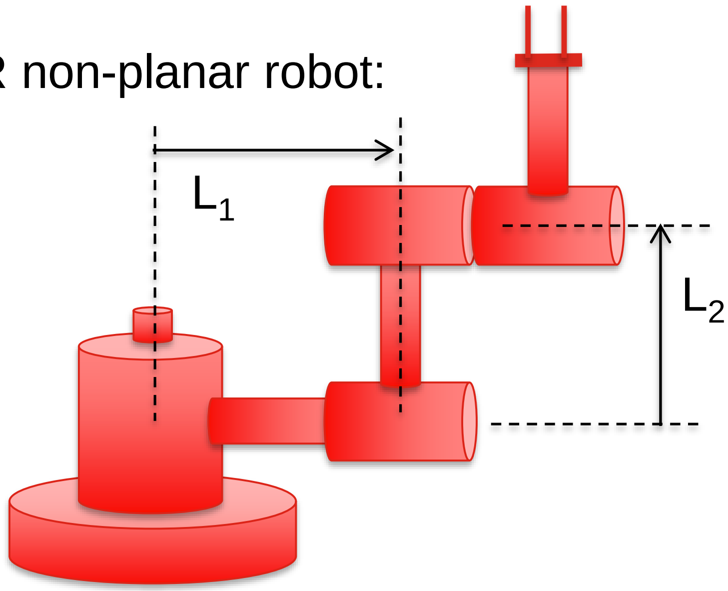


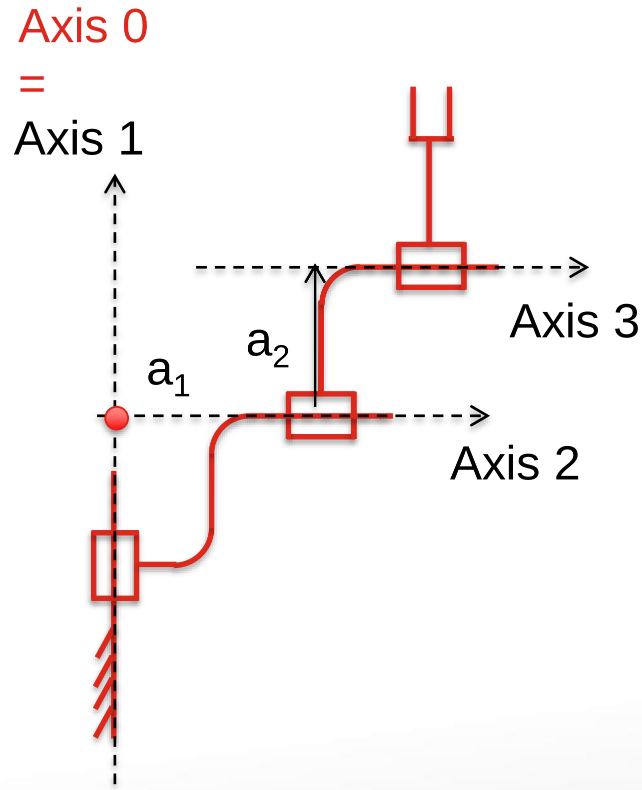


- What about α_0?
  - α_0 = angle between axis 0 and axis 1, about a_0 . However, axis 0 is not fully known yet.
  - By convention, $\alpha_0 = 0$.
  - This means: Axis 0 and Axis 1 are the same.

Example 3

- 3-link RRR non-planar robot:





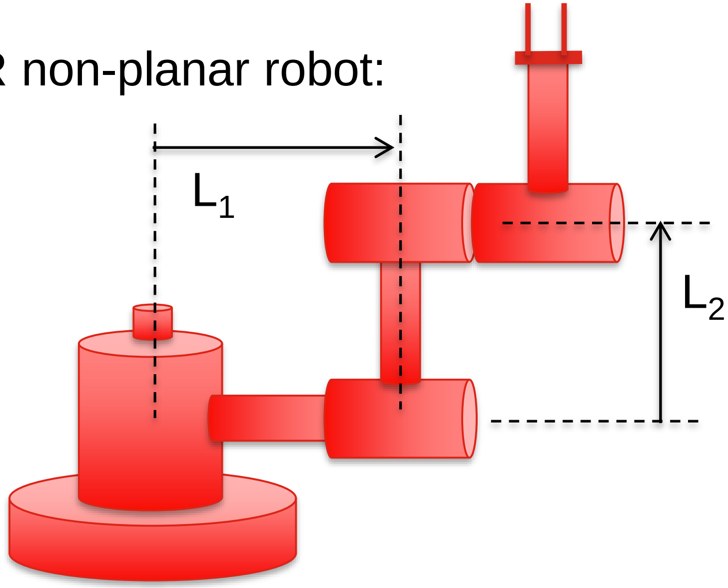
- Step 5, write down the link offsets d_i.
- Definition: d_i = distance from a_{i-1} to a_i, along axis i.

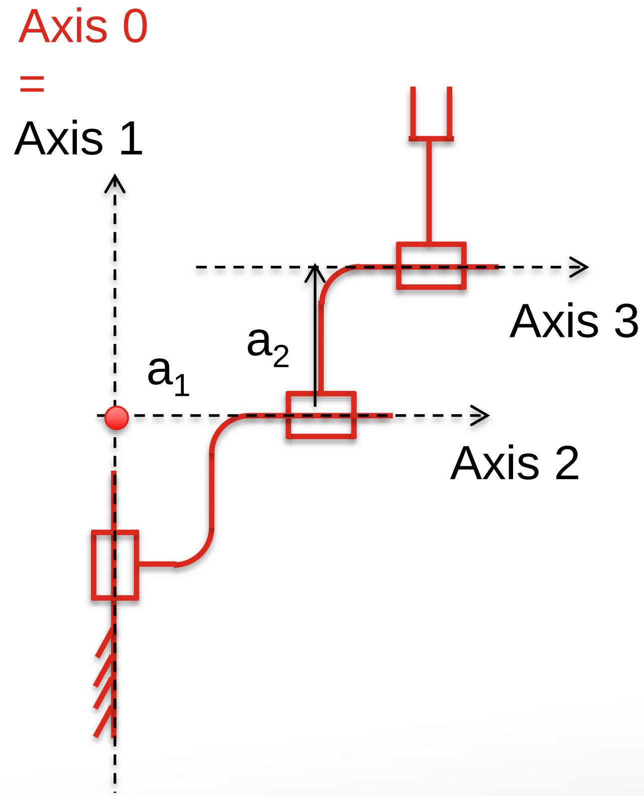
Do 2 to n-1

- d_2 = distance from a1 to a2, along axis 2, is L_1.

Example 3

- 3-link RRR non-planar robot:

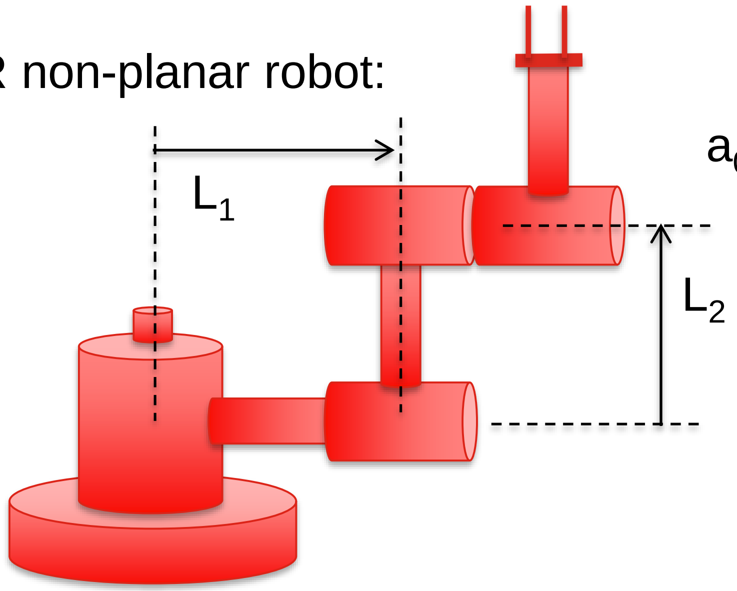


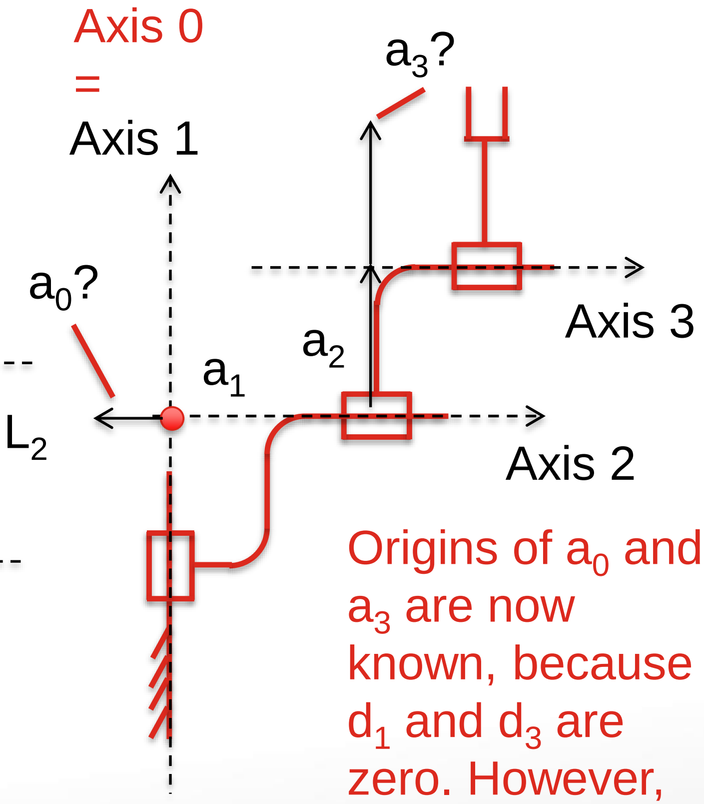


- What about d_1 and d_3?
  - d_1 = distance from a_0 to a_1, along axis 1.
  - d_3 = distance from a_2 to a_3, along axis 3.
  - But where exactly are a_0 and a_3?

Example 3

- 3-link RRR non-planar robot:



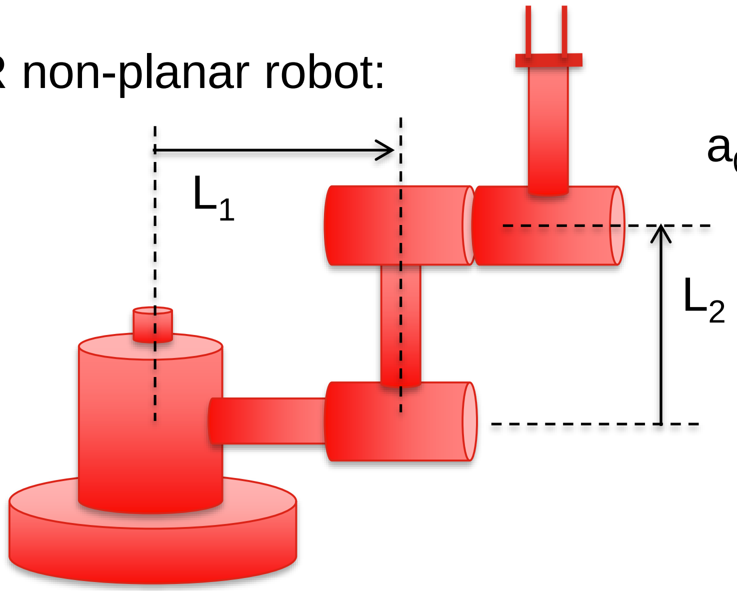


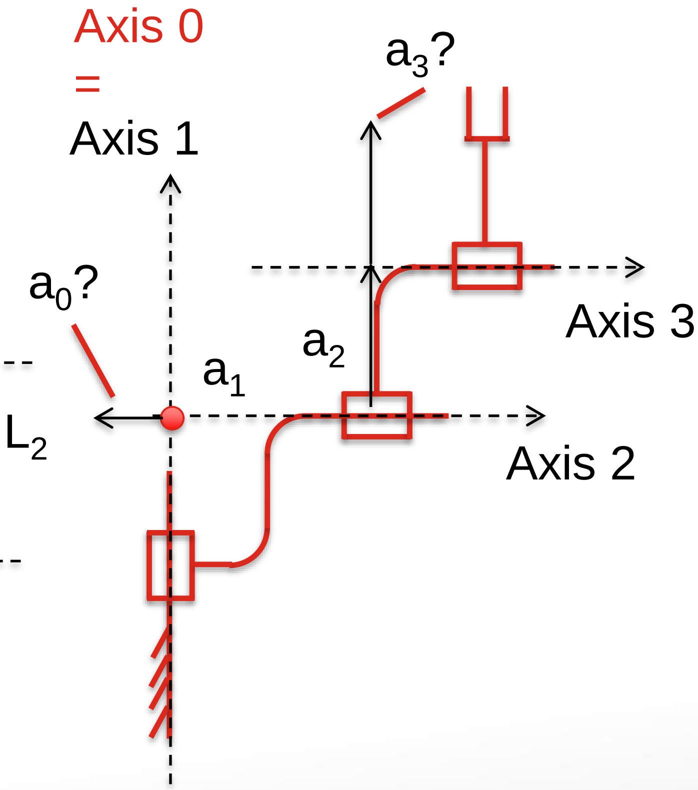
Origins of a_0 and a_3 are now known, because d_1 and d_3 are zero. However, the direction is not known.

- What about d_1 and d_3?
  - d_1 = distance from a_0 to a_1, along axis 1.
  - d_3 = distance from a_2 to a_3, along axis 3.
  - By convention: Zero for revolute joint, variable for prismatic joint
  - So in this case, d_1 and d_3 are both zero.

Example 3

- 3-link RRR non-planar robot:





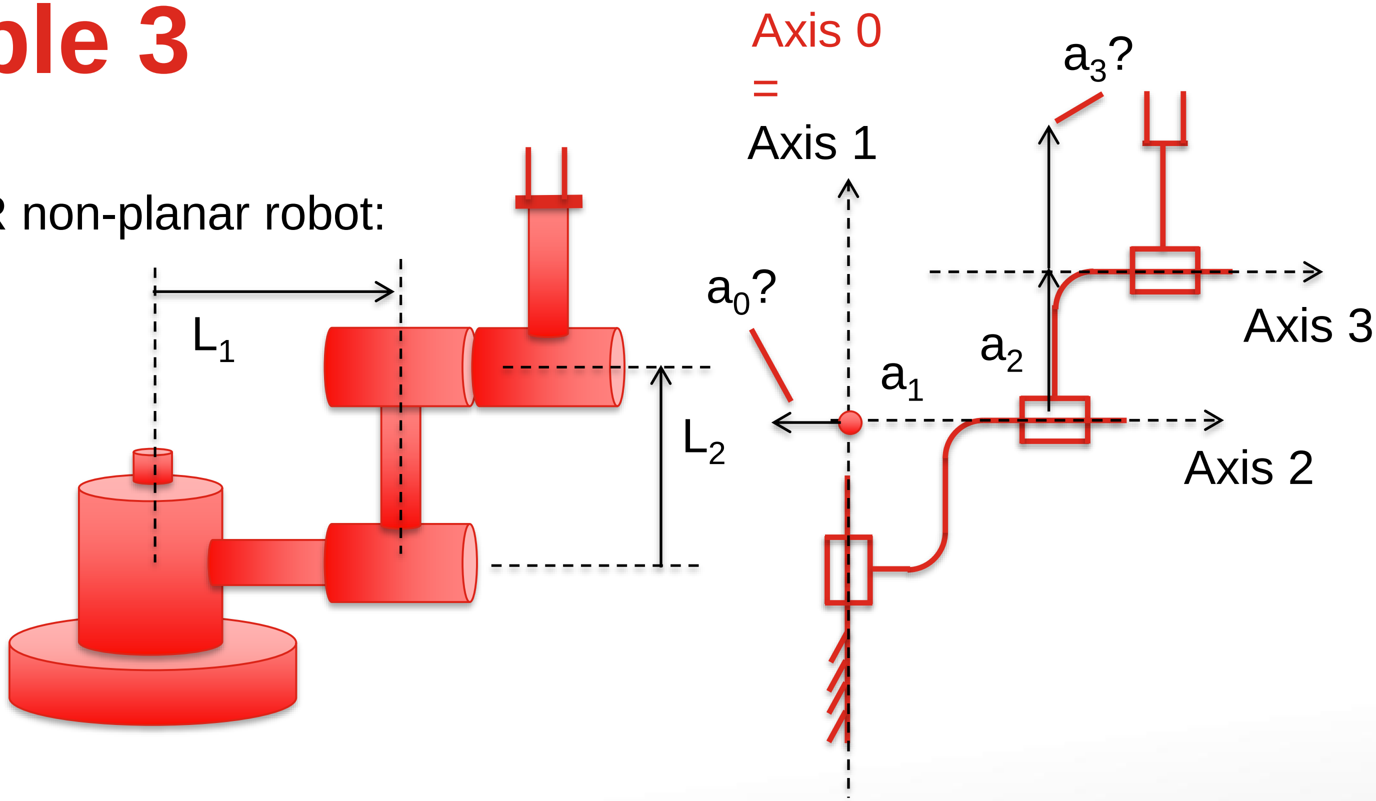
- Step 6, write down the joint angle θ_i.
- Definition: θ_i is the angle between the (extension of a_{i-1}) and a_i, measured about the axis i.

Do 2 to n-1

- θ_2 = angle between (extension of a1) and a2, about axis 2.
- Here, θ_2 is a variable.

Example 3

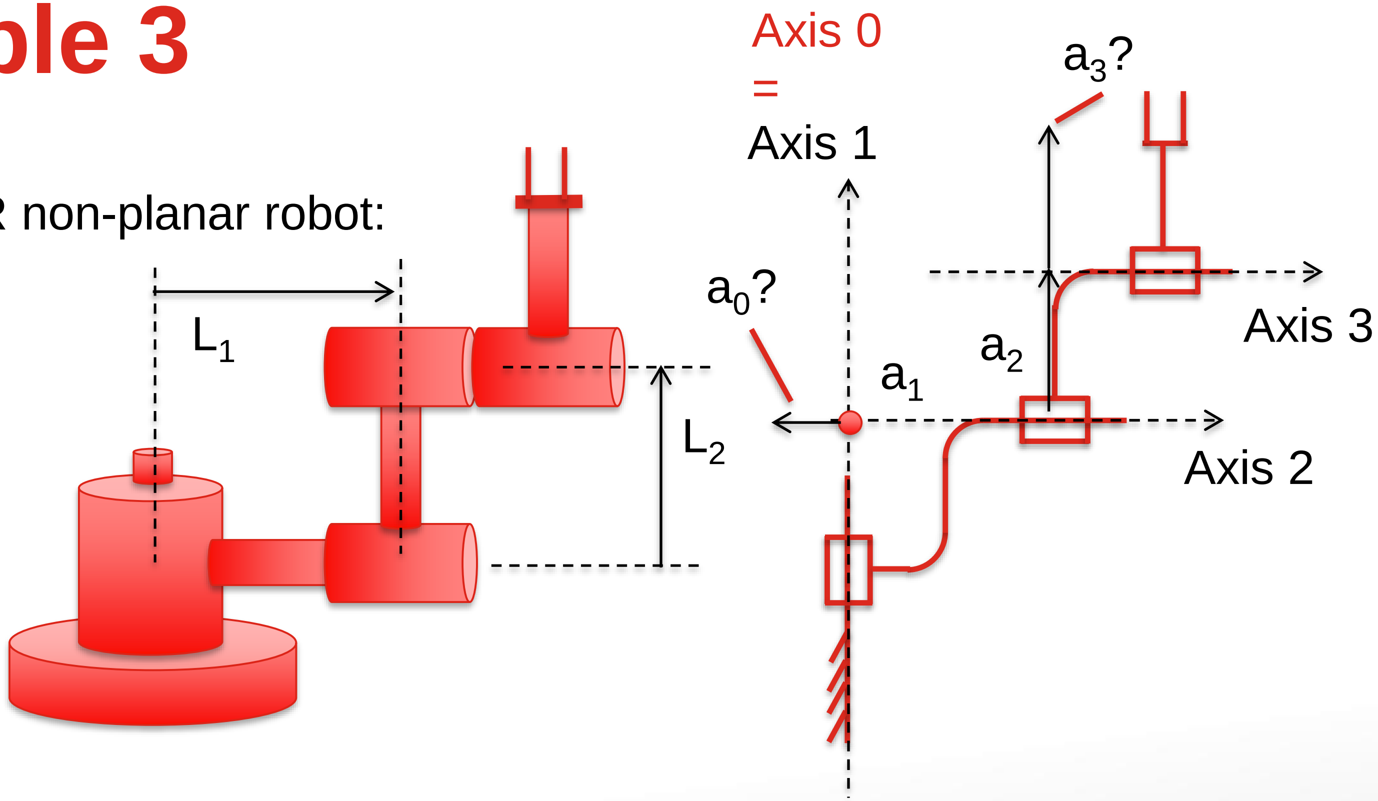
- 3-link RRR non-planar robot:



- What about θ_1 and θ_3?
  - θ_1 = angle between (extension of a0) and a1, about axis 1.
  - θ_3 = angle between (extension of a2) and a3, about axis 3.
  - By convention: Zero for prismatic joint, variable for revolute joint
  - So in this case, θ_1 and θ_3 are both variables.

Example 3

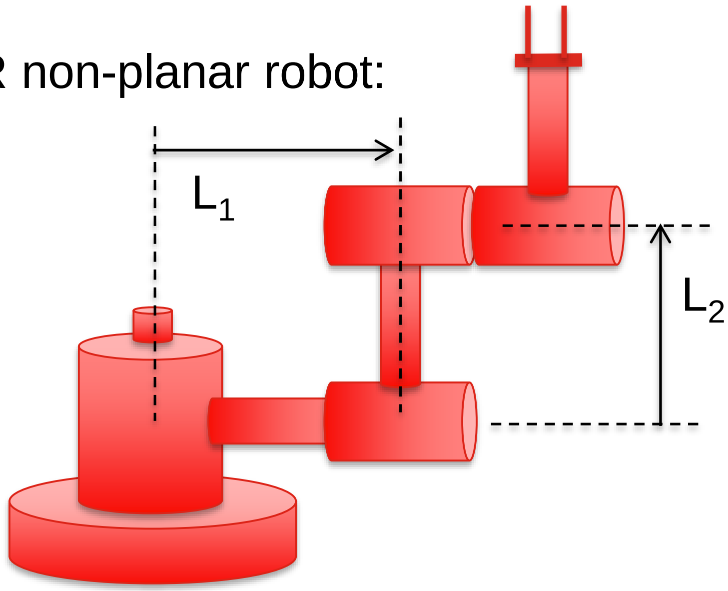
- 3-link RRR non-planar robot:

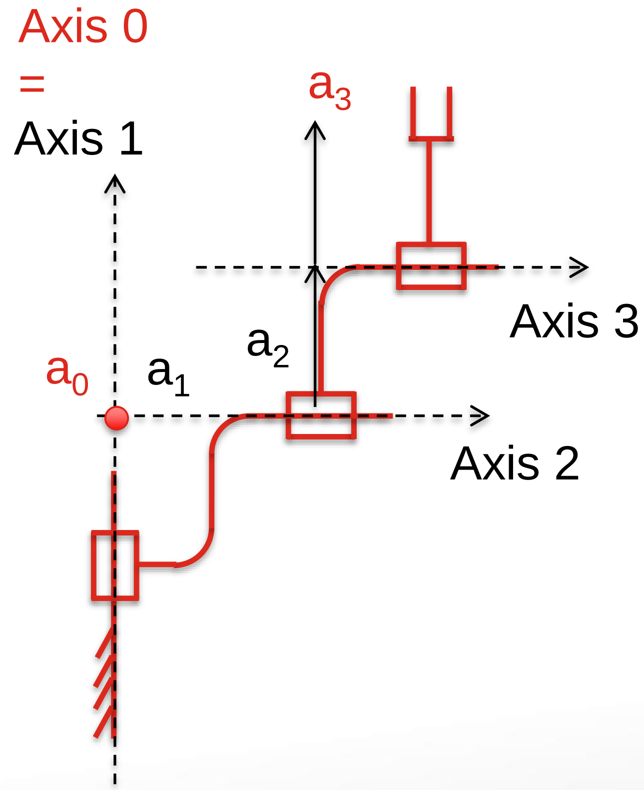


- We still have a problem. Since θ_1 and θ_3 are both variables, we need to determine their “zero”-angle position.

Example 3

- 3-link RRR non-planar robot:

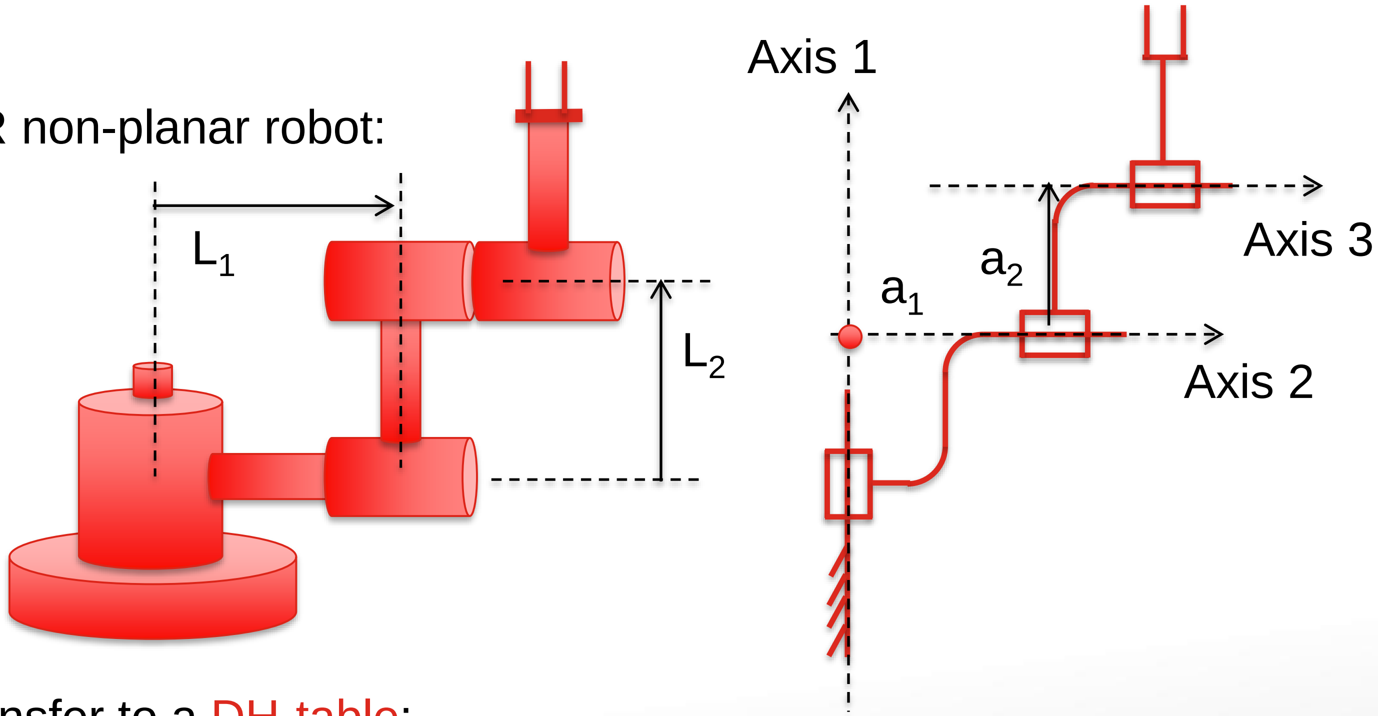




- For convenience, align a_0 with a_1 when the joint variable 1 is zero.
- As for joint n:
  - Revolute: align a_n with a_{n-1} when $\theta_n = 0$.
  - Prismatic: align a_n with a_{n-1} when $d_n = 0$.

Example 3

- 3-link RRR non-planar robot:

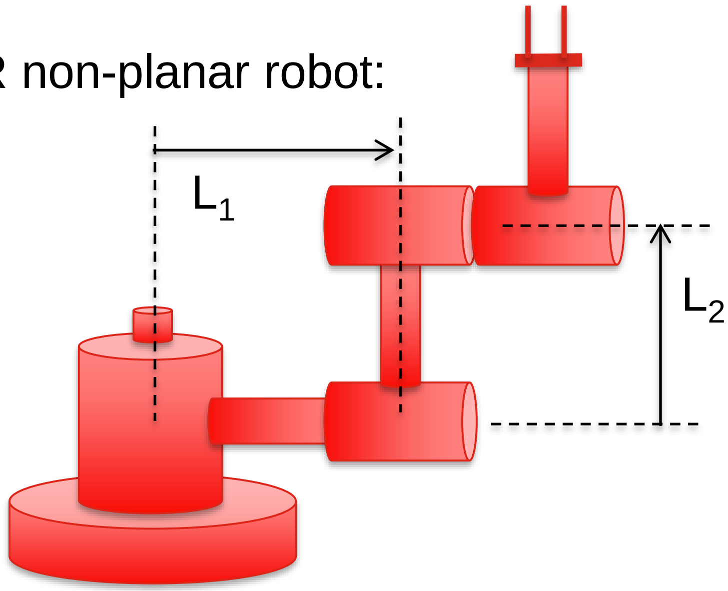


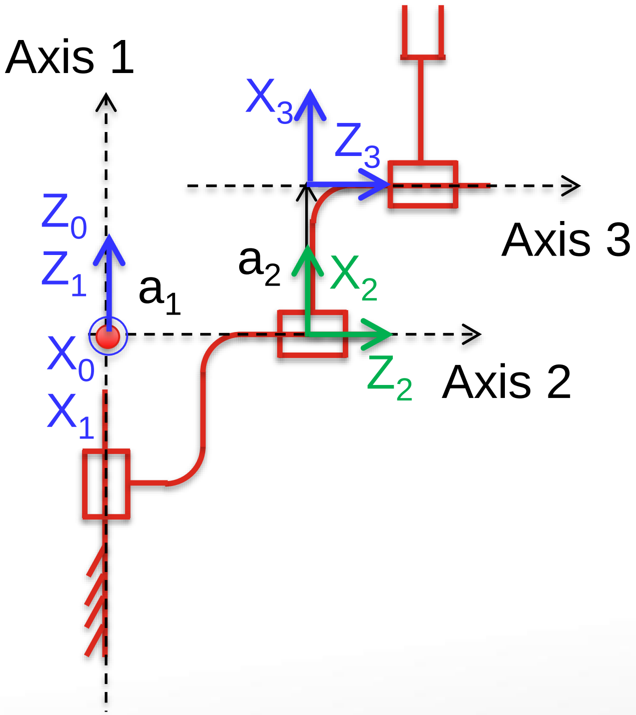
- Step 7, transfer to a DH-table:

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	-90	0	L_1	θ_2
3	0	L_2	0	θ_3

Example 3

- 3-link RRR non-planar robot:





- Step 8, insert the frames. Rules:
  - Z-axis of frame {i}, i.e. Z_i, is coincident with joint axis i.
  - Origin of frame {i} is where the a_i intersects the joint i axis.
  - X-axis of frame {i}, i.e. X_i, is coincident with a_i.

Example 3

- Step 9 (Final step!), calculate the transformations.

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	-90	0	L_1	θ_2
3	0	L_2	0	θ_3

$$ {}^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ c\alpha_{i-1}s\theta_i & c\alpha_{i-1}c\theta_i & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\alpha_{i-1}s\theta_i & s\alpha_{i-1}c\theta_i & c\alpha_{i-1} & c\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{0}_{1}T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

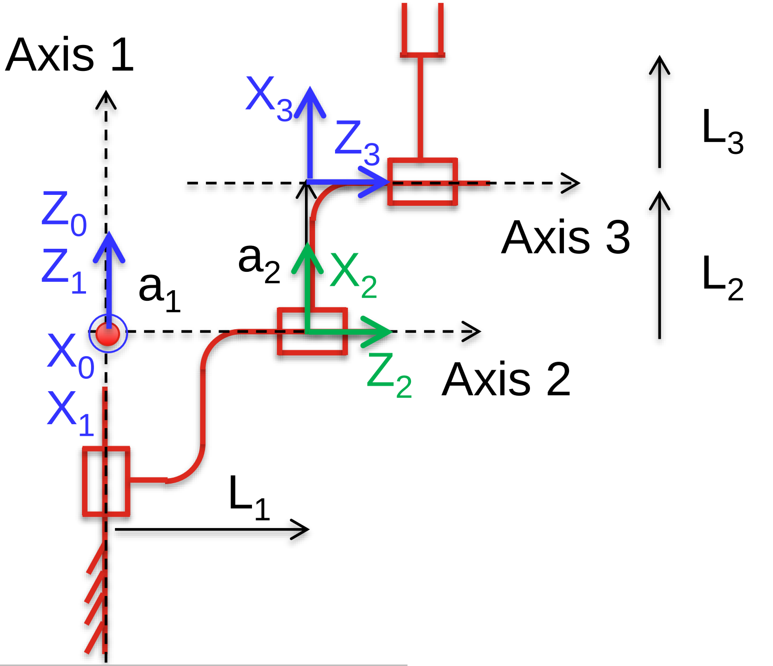
$$ {}^{1}_{2}T = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & 0 \\ 0 & 0 & 1 & L_1 \\ -s\theta_2 & -c\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{2}_{3}T = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & L_2 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{0}_{3}T = {}^{0}_{1}T \cdot {}^{1}_{2}T \cdot {}^{2}_{3}T = \begin{bmatrix} c_1c_{23} & -c_1s_{23} & -s_1 & L_2c_1c_2 - L_1s_1 \\ s_1c_{23} & -s_1s_{23} & c_1 & L_2s_1c_2 + L_1c_1 \\ -s_{23} & -c_{23} & 0 & -L_2s_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Example 3

- Verification.
- The end-effector, with reference to frame {3}, has position $[L_3, 0, 0]^T$, and same orientation as frame {3}. Therefore:

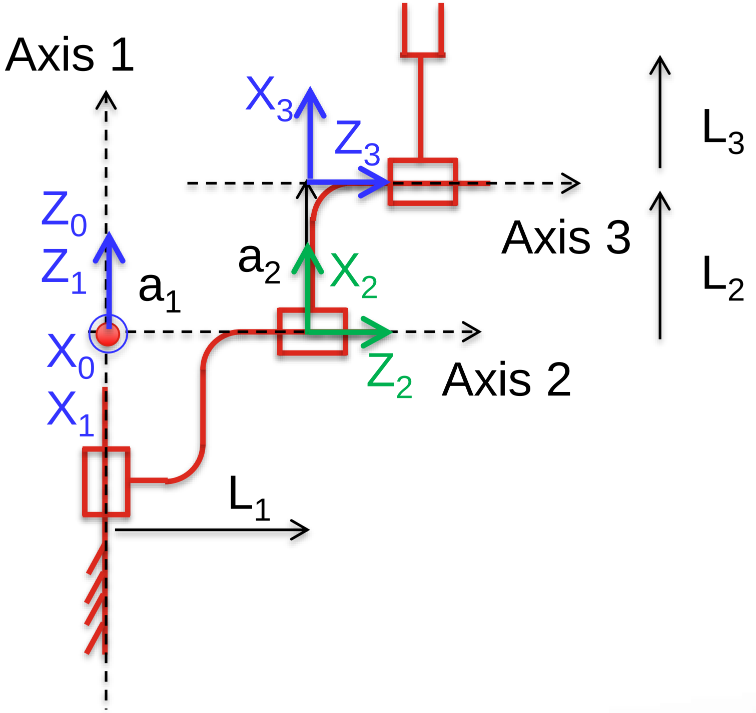


$$
{}^{0}P = {}^{0}_{3}T \cdot {}^{3}P
$$

$$
= \begin{bmatrix} c_1c_{23} & -c_1s_{23} & -s_1 & L_2c_1c_2 - L_1s_1 \\ s_1c_{23} & -s_1s_{23} & c_1 & L_2s_1c_2 + L_1c_1 \\ -s_{23} & -c_{23} & 0 & -L_2s_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} L_3 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} L_3c_1c_{23} + L_2c_1c_2 - L_1s_1 \\ L_3s_1c_{23} + L_2s_1c_2 + L_1c_1 \\ -L_3s_{23} - L_2s_2 \\ 1 \end{bmatrix}
$$

- Does this make sense?

Example 3



- Check:
  - When all θ = 0, we have:

$$^{0}P = \begin{bmatrix} L_3 + L_2 \\ L_1 c_1 \\ 0 \\ 1 \end{bmatrix}$$

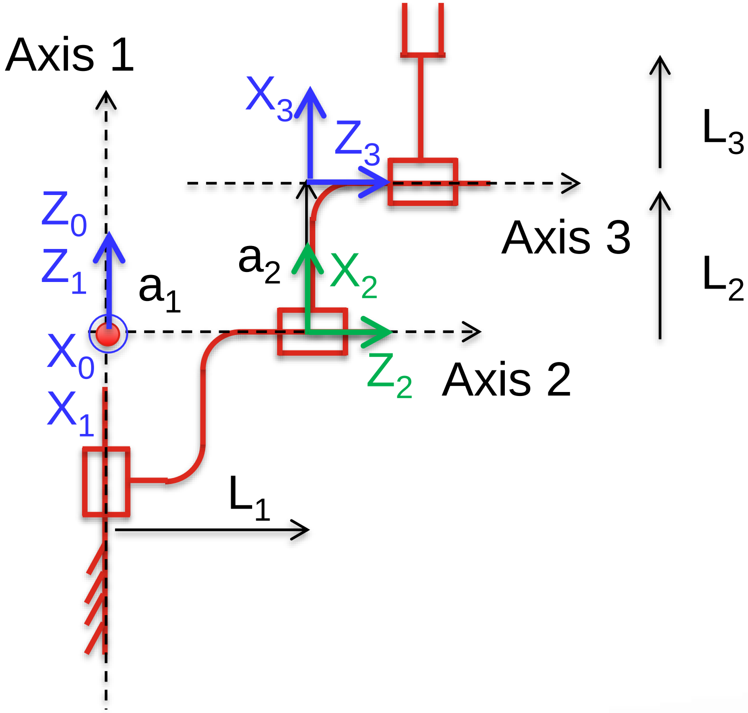
  - From the figure, it seems that the point should be at $[0, L_1c_1, L_3+L_2]^T$ instead.
  - What went wrong?
  - Nothing!!
  - Remember the definition of θ_i: angle between a_{i-1} and a_i (about axis i).
  - So the figure is actually showing $\theta_1 = 0$, $\theta_2 = -90$, $\theta_3 = 0$.
  - For the case of all θ = 0, we would have x_2 and x_3 pointed out of the plane, and the point is indeed $[L_3+L_2, L_1c_1, 0]^T$ as calculated.

Example 3

- Check:
  - As for $\theta_1 = 0$, $\theta_2 = -90$, $\theta_3 = 0$ as shown in figure, we have:

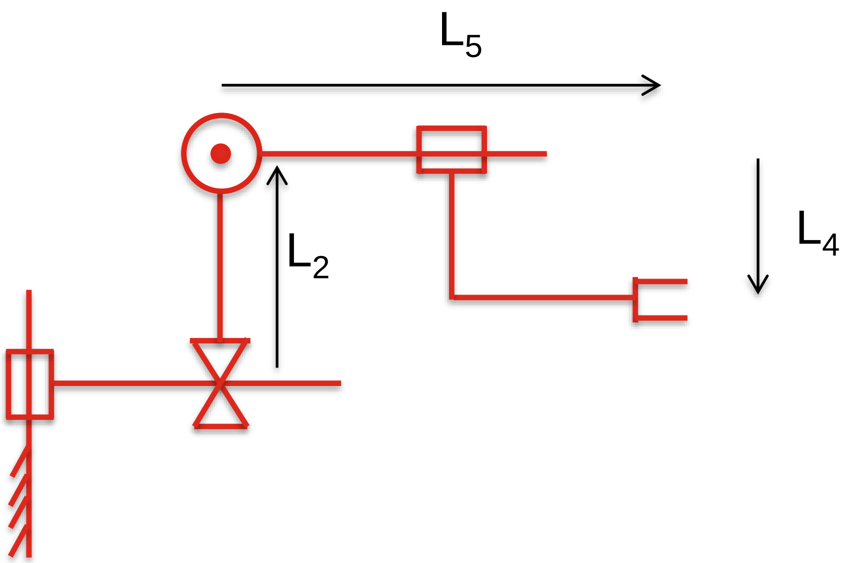
$$^{0}P = \begin{bmatrix} 0 \\ L_1 \\ L_3 + L_2 \\ 1 \end{bmatrix}$$

  - which looks correct from the figure.



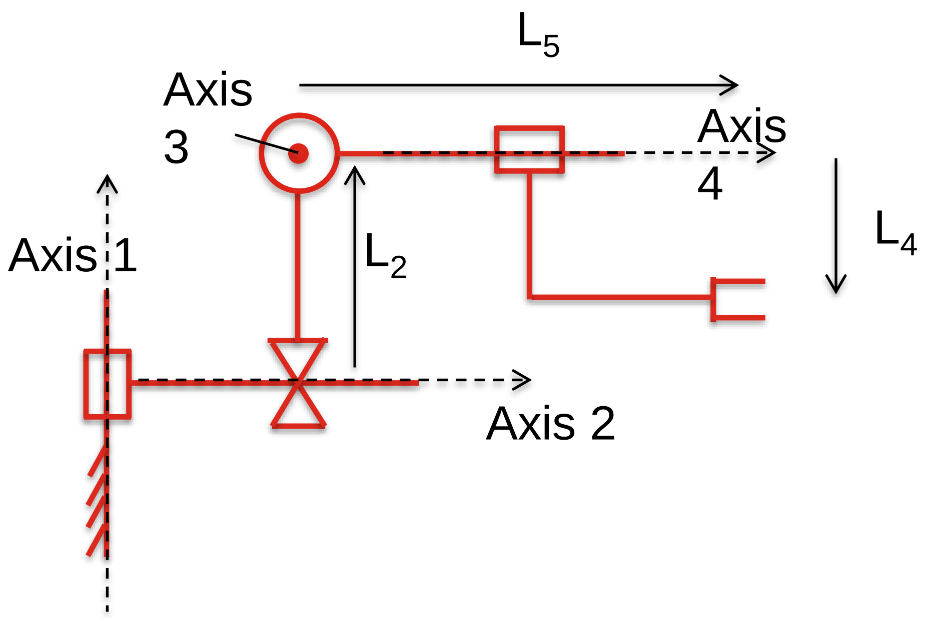
Example 4

- 4-link RPRR non-planar robot:



Example 4

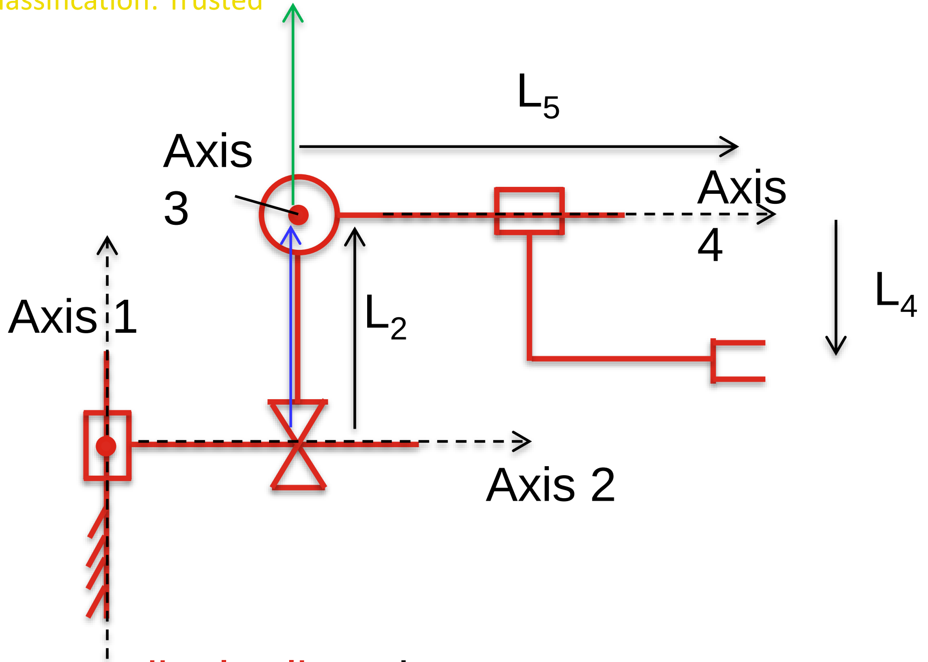
- 4-link RPRR non-planar robot:



- Step 1, draw the axes.
    - For rotary joint: About the rotation
    - For prismatic joint: Along the translation

Example 4

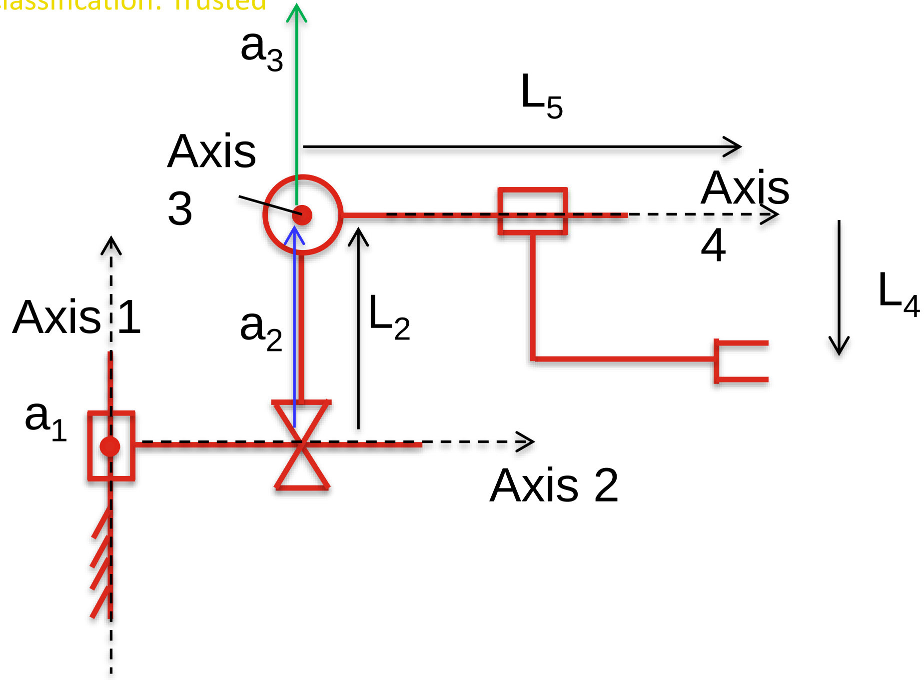
- 4-link RPRR non-planar robot:



- Step 2, we draw the mutually perpendicular lines between axes.
  - Draw from lower axis to higher axis, e.g. 1 to 2, 2 to 3…
  - Axis 1 and Axis 2 intersect.
    - Line is perpendicular to the plane made by axes 1 and 2.
  - Between Axis 2 and Axis 3
    - Blue arrow as shown.
  - Axis 3 and Axis 4 intersect.
    - Green arrow as shown.

Example 4

- 4-link RPRR non-planar robot:



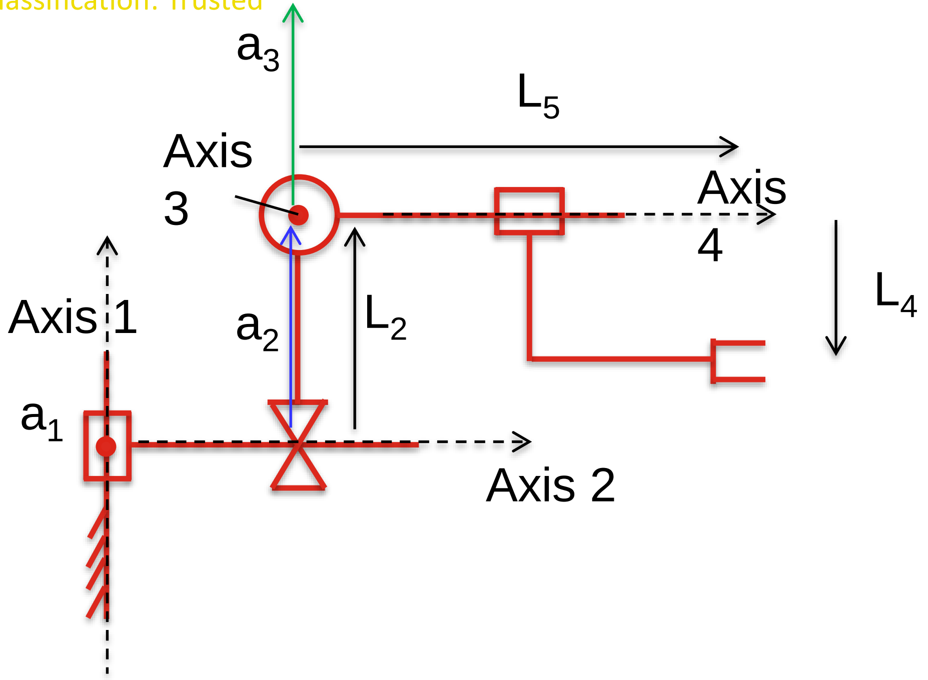
- Step 3, put in the link lengths a_{i-1}.
- $\boxed{\text{Definition: } a_{i-1} = \text{length of mutual perpendicular, from axis i-1 to axis i.}}$

Do 1 to n-1

- a_1 = length of mutual perpendicular from axis 1 to 2 = 0 (intersect).
- a_2 = length of mutual perpendicular from axis 2 to 3 = L_2 .
- a_3 = length of mutual perpendicular from axis 3 to 4 = 0 (intersect).

Example 4

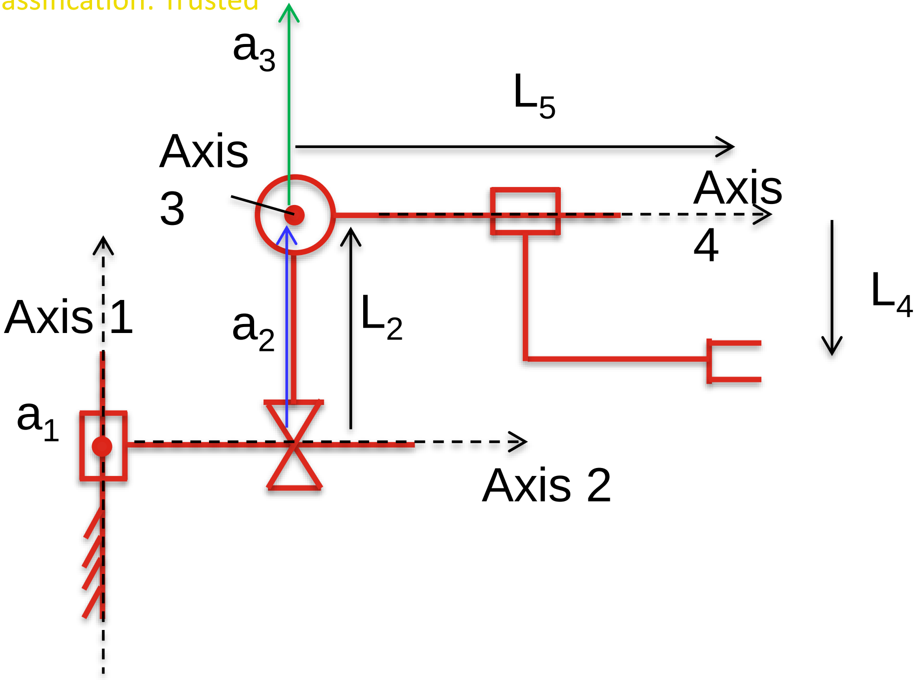
- 4-link RPRR non-planar robot:



- What about a_0?
  - a_0 = length of mutual perpendicular from axis 0 to 1. However, axis 0 is not known yet.

Example 4

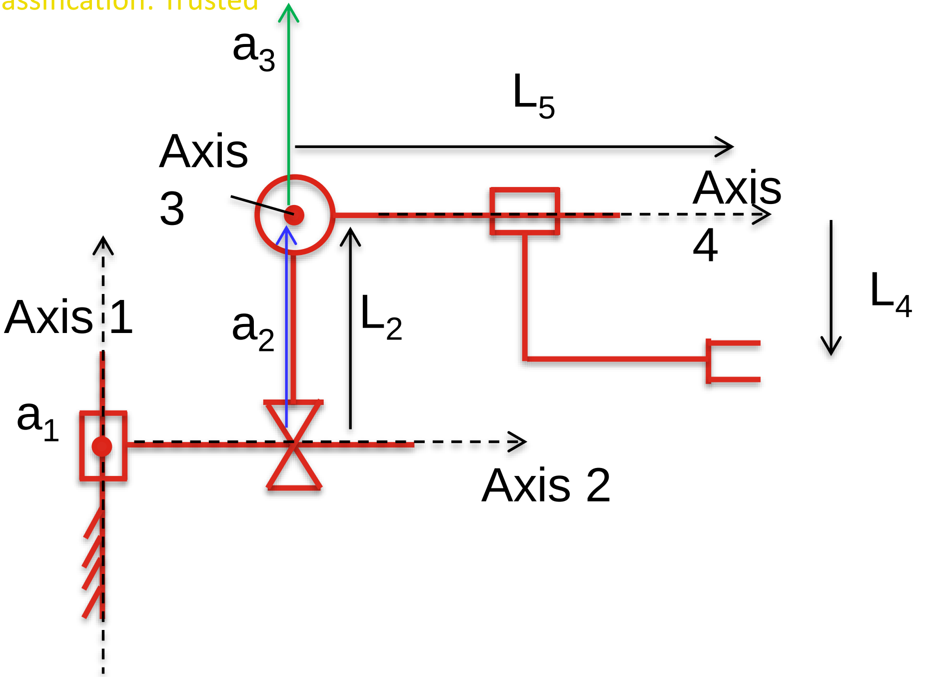
- 4-link RPRR non-planar robot: Axis 0 intersecting with Axis 1



- What about a_0?
  - a_0 = length of mutual perpendicular from axis 0 to 1. However, axis 0 is not known yet.
  - By convention, $a_0 = 0$.
  - This means: Axis 0 and Axis 1 intersect with each other.

Example 4

- 4-link RPRR non-planar robot:
  Axis 0 intersecting with Axis 1



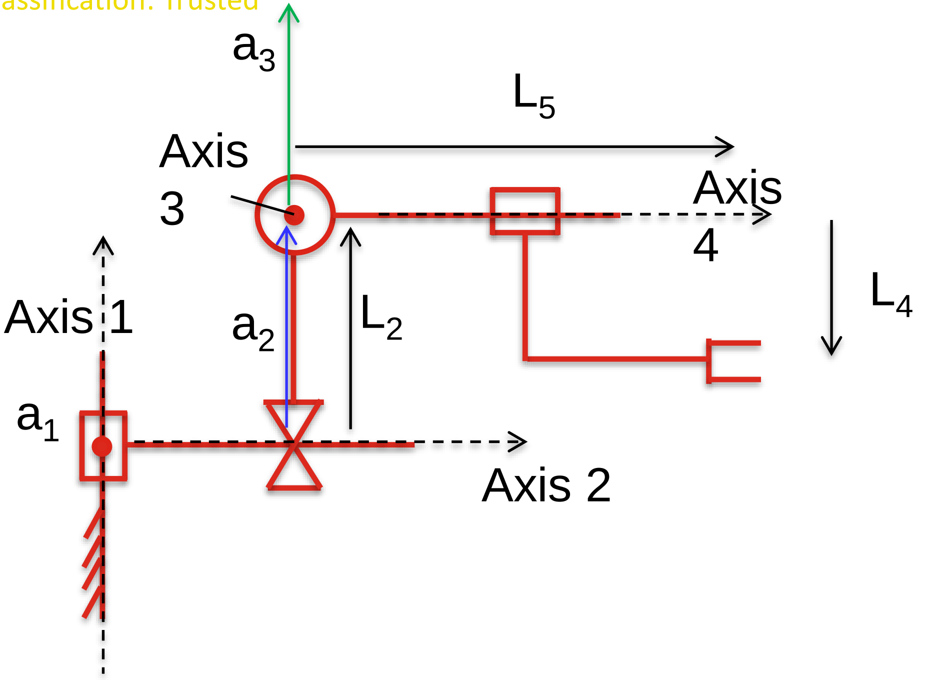
- Step 4, put in link twists α_{i-1}.
- Definition: α_{i-1} = angle between axis i-1 and axis i, in the right hand sense about a_{i-1}

Do 1 to n-1

- α_1 = angle between axis 1 and axis 2, about a_1 = -90deg.
- α_2 = angle between axis 2 and axis 3, about a_2 = -90deg.
- α_3 = angle between axis 3 and axis 4, about a_3 = 90deg.

Example 4

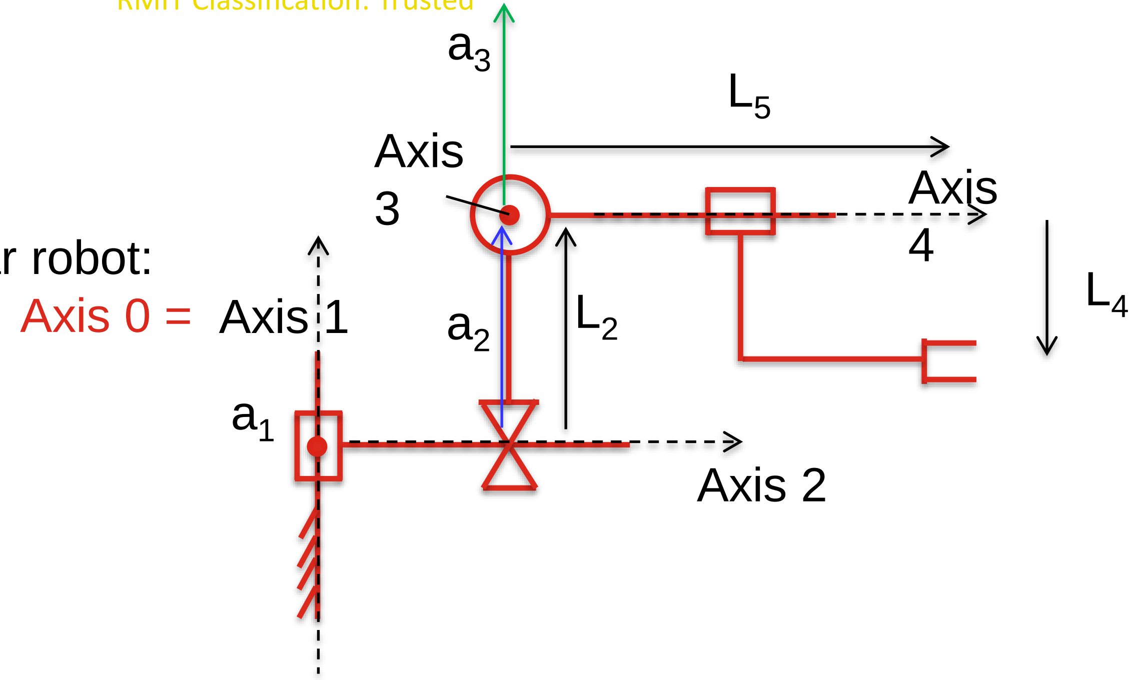
- 4-link RPRR non-planar robot:
  Axis 0 intersecting with Axis 1



- What about α_0?
  - α_0 = angle between axis 0 and axis 1, about a_0 . However, axis 0 is not fully known yet.

Example 4

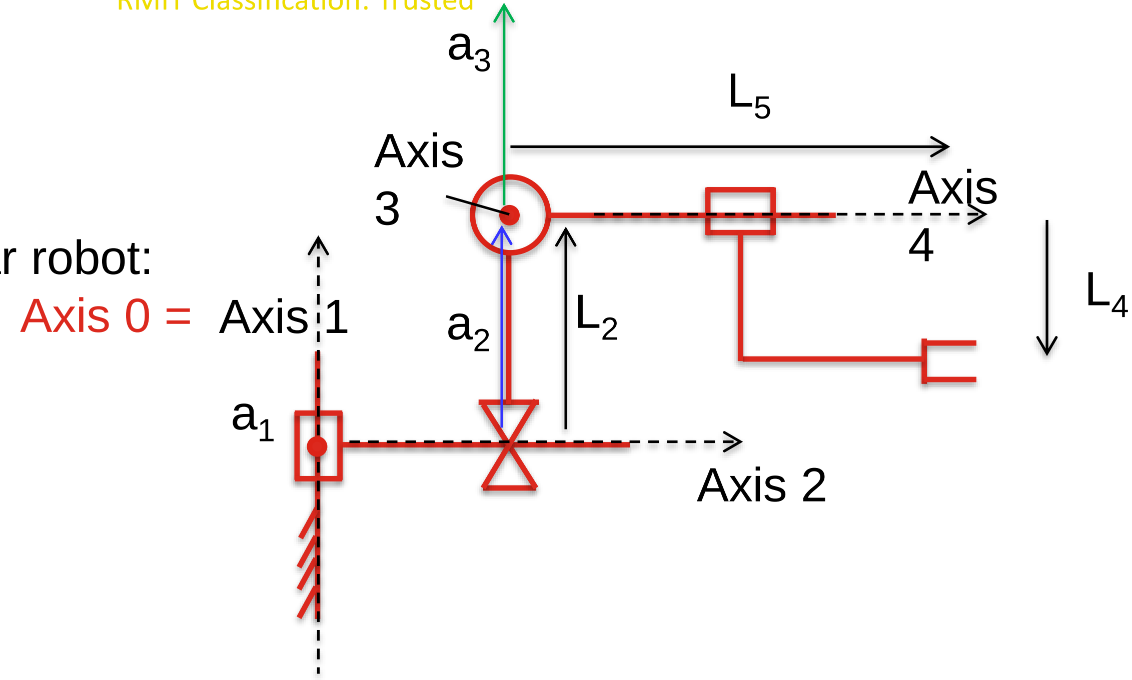
- 4-link RPRR non-planar robot:



- What about α_0?
  - α_0 = angle between axis 0 and axis 1, about a_0 . However, axis 0 is not fully known yet.
  - By convention, $\alpha_0 = 0$.
  - This means: Axis 0 and Axis 1 are the same.

Example 4

- 4-link RPRR non-planar robot:

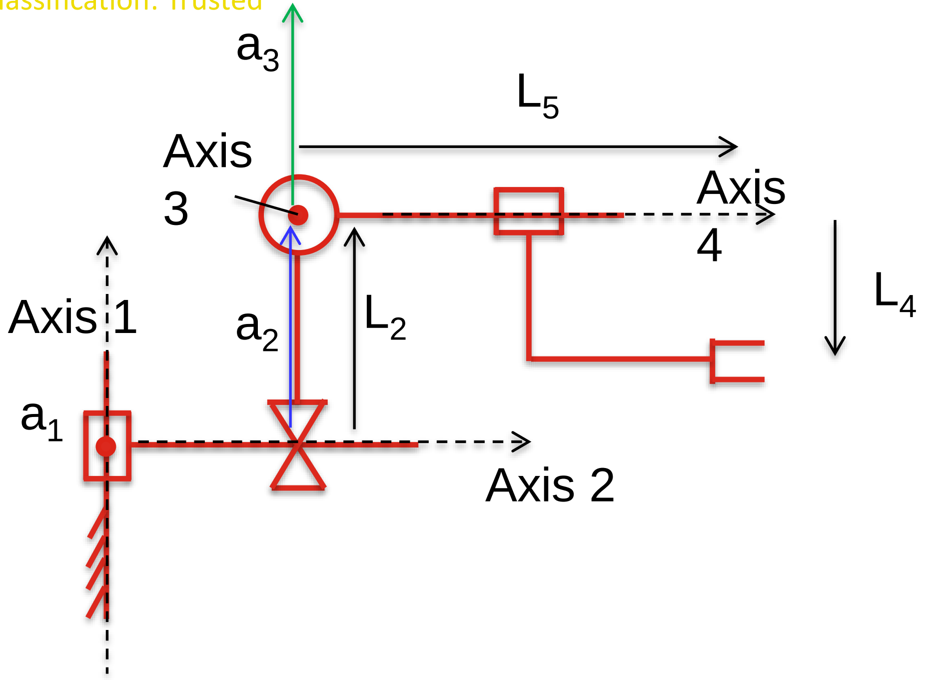


- Step 5, write down the link offsets d_i.
- Definition: d_i = distance from a_{i-1} to a_i, along axis i.

Do 2 to n-1
- d_2 = distance from a1 to a2, along axis 2, is a variable.
- d_3 = distance from a2 to a3, along axis 3, is a zero.

Example 4

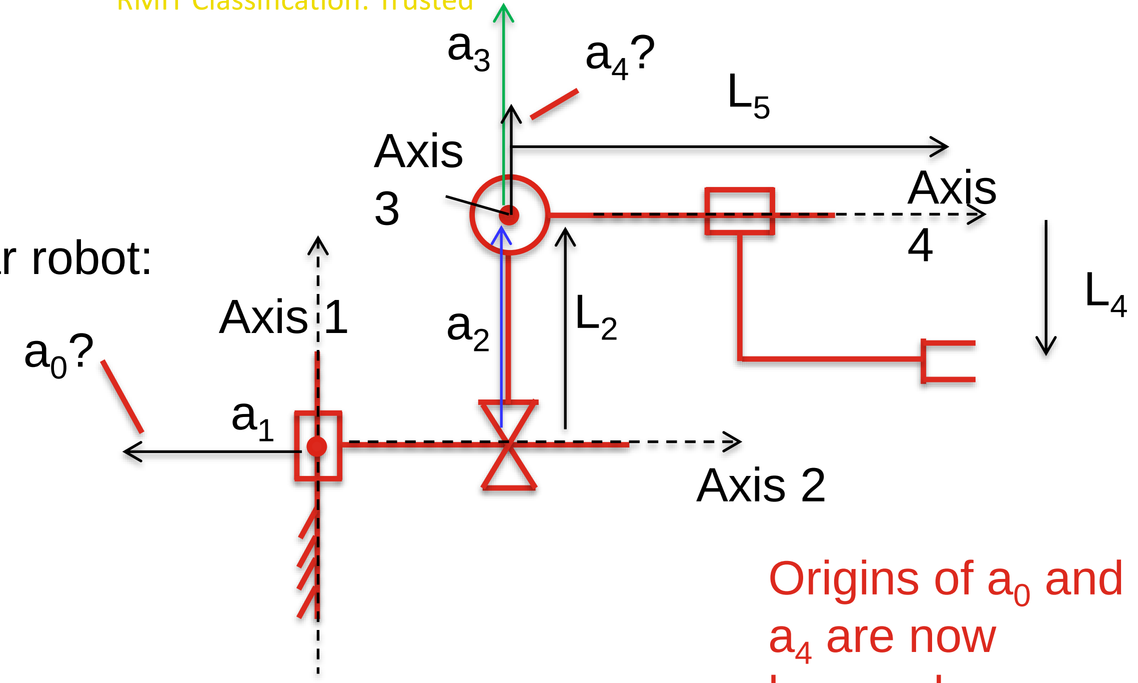
- 4-link RPRR non-planar robot:



- What about d_1 and d_4?
  - d_1 = distance from a_0 to a_1, along axis 1.
  - d_4 = distance from a_3 to a_4, along axis 4.
  - But where exactly are a_0 and a_4?

Example 4

- 4-link RPRR non-planar robot:

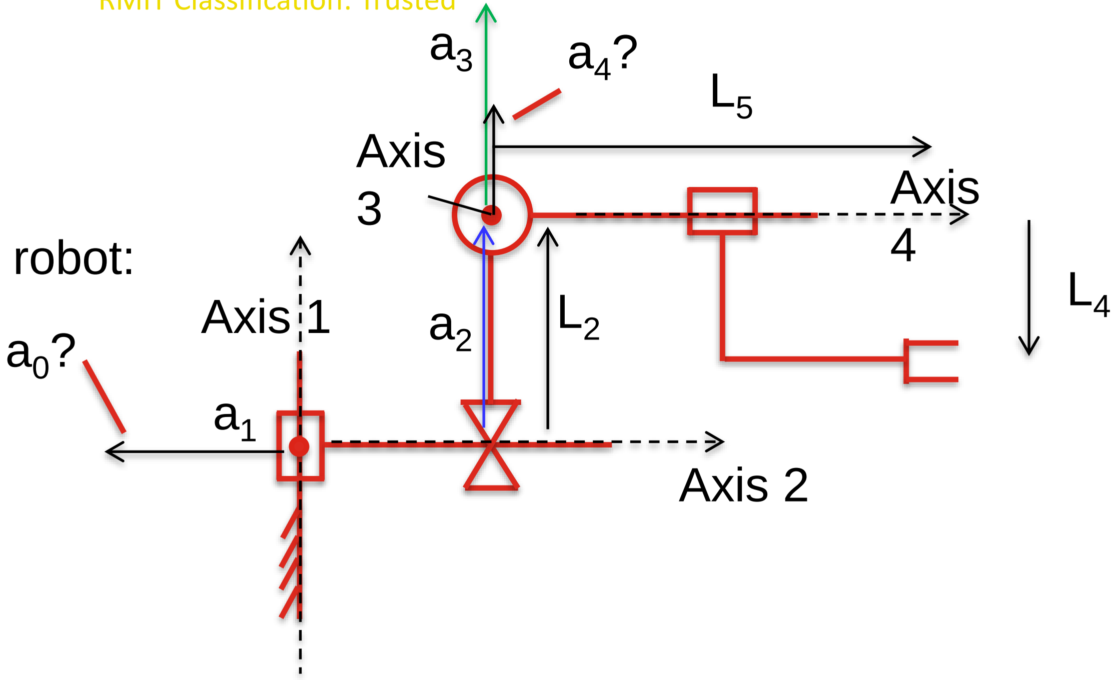


Origins of a_0 and a_4 are now known, because d_1 and d_4 are zero. However, the direction is not known.

- What about d_1 and d_4?
  - d_1 = distance from a_0 to a_1, along axis 1.
  - d_4 = distance from a_3 to a_4, along axis 4.
  - By convention: Zero for revolute joint, variable for prismatic joint.
  - So in this case, d_1 and d_4 are both zero.

Example 4

- 4-link RPRR non-planar robot:



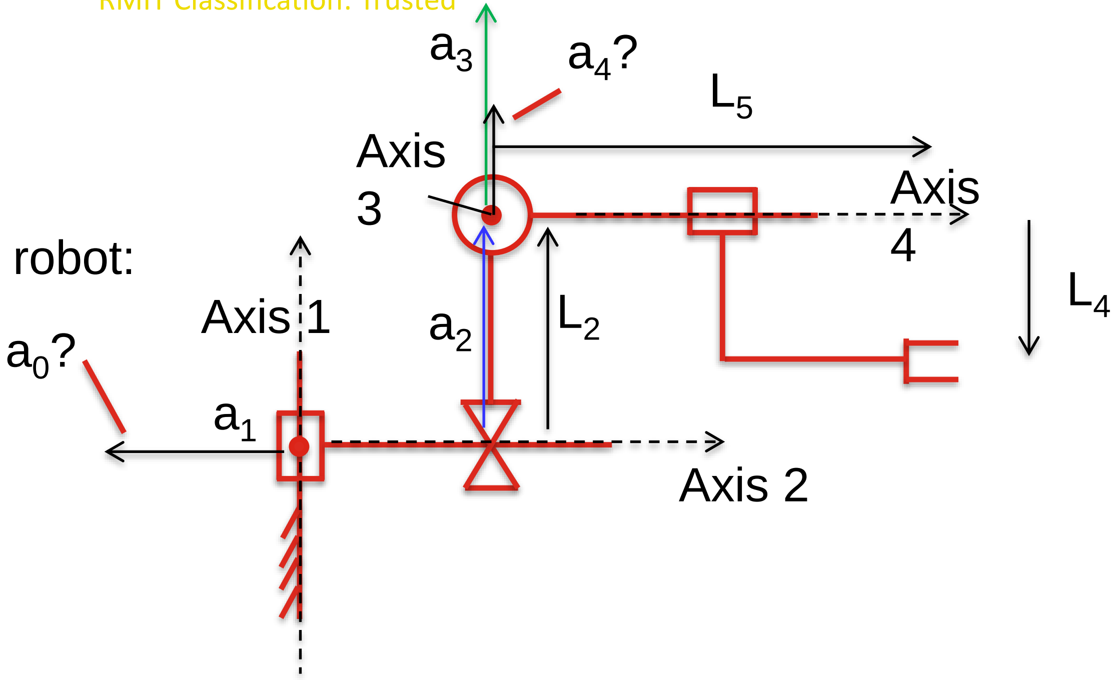
- Step 6, write down the joint angle θ_i.
- Definition: θ_i is the angle between the (extension of a_{i-1}) and a_i, measured about the axis i.

Do 2 to n-1

- θ_2 = angle between (extension of a1) and a2, about axis 2 = -90deg (constant because prismatic).
- θ_3 = angle between (extension of a2) and a3, about axis 3 = variable.

Example 4

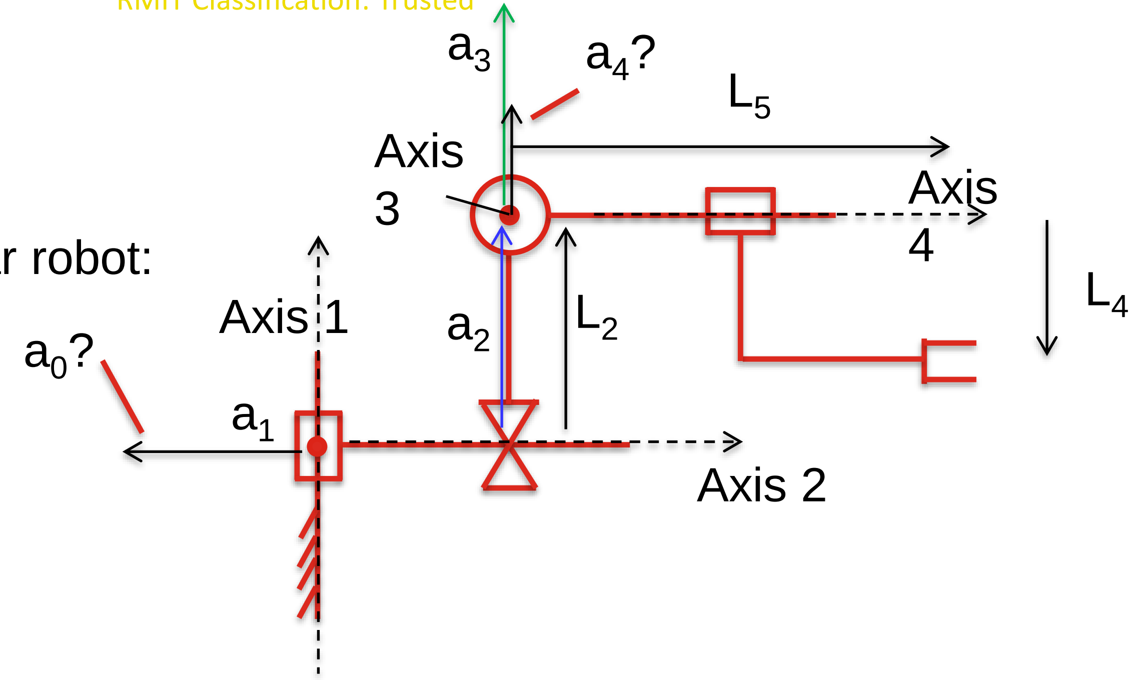
- 4-link RPRR non-planar robot:



- What about θ_1 and θ_4?
  - θ_1 = angle between (extension of a0) and a1, about axis 1.
  - θ_4 = angle between (extension of a3) and a4, about axis 4.
  - By convention: Zero for prismatic joint, variable for revolute joint.
  - So in this case, θ_1 and θ_4 are both variables.

Example 4

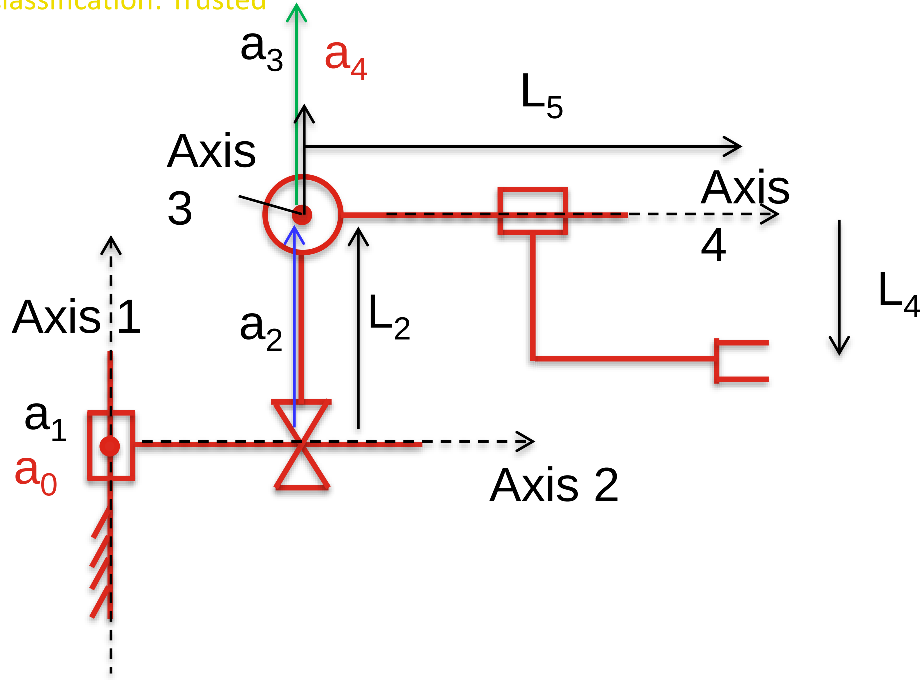
- 4-link RPRR non-planar robot:



- We still have a problem. Since θ_1 and θ_4 are both variables, we need to determine their “zero”-angle position.

Example 4

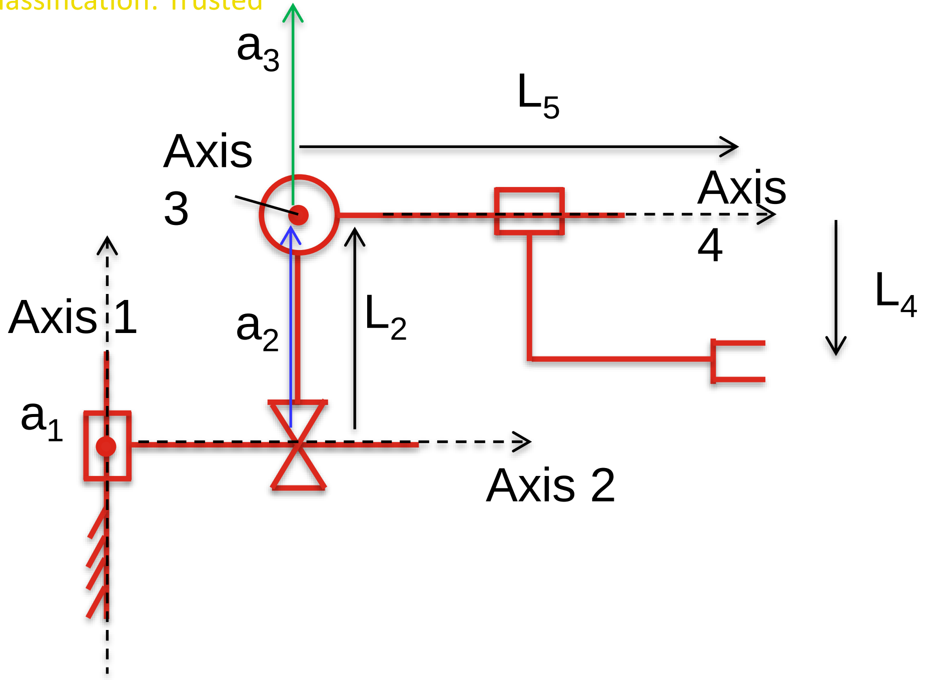
- 4-link RPRR non-planar robot:



- For convenience, align a_0 with a_1 when the joint variable 1 is zero.
- As for joint n:
  - Revolute: align a_n with a_{n-1} when $\theta_n = 0$.
  - Prismatic: align a_n with a_{n-1} when $d_n = 0$.

Example 4

- 4-link RPRR non-planar robot:

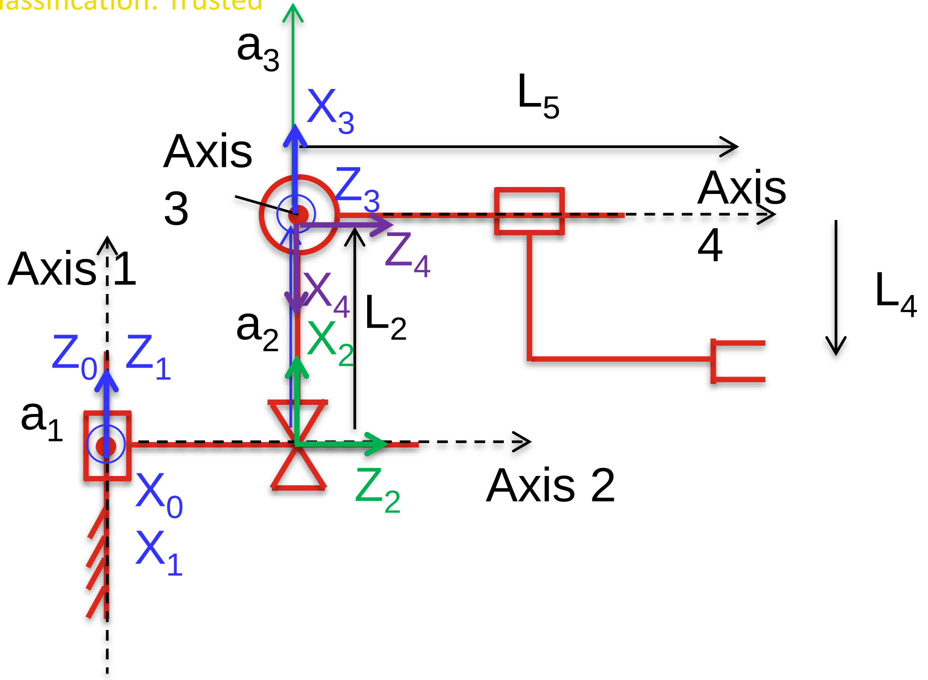


- Step 7, transfer to a DH-table:

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	-90	0	d_2	-90
3	-90	L_2	0	θ_3
4	90	0	0	θ_4

Example 4

- 4-link RPRR non-planar robot:



- Step 8, insert the frames. Rules:
  - Z-axis of frame {i}, i.e. Z_i, is coincident with joint axis i.
  - Origin of frame {i} is where the a_i intersects the joint i axis.
  - X-axis of frame {i}, i.e. X_i, is coincident with a_i.

Example 4

- Step 9 (Final step!), calculate the transformations.

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	-90	0	d_2	-90
3	-90	L_2	0	θ_3
4	90	0	0	θ_4

$$ {}^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ c\alpha_{i-1}s\theta_i & c\alpha_{i-1}c\theta_i & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\alpha_{i-1}s\theta_i & s\alpha_{i-1}c\theta_i & c\alpha_{i-1} & c\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{0}_{1}T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{1}_{2}T = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{2}_{3}T = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & L_2 \\ 0 & 0 & 1 & 0 \\ -s\theta_3 & -c\theta_3 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

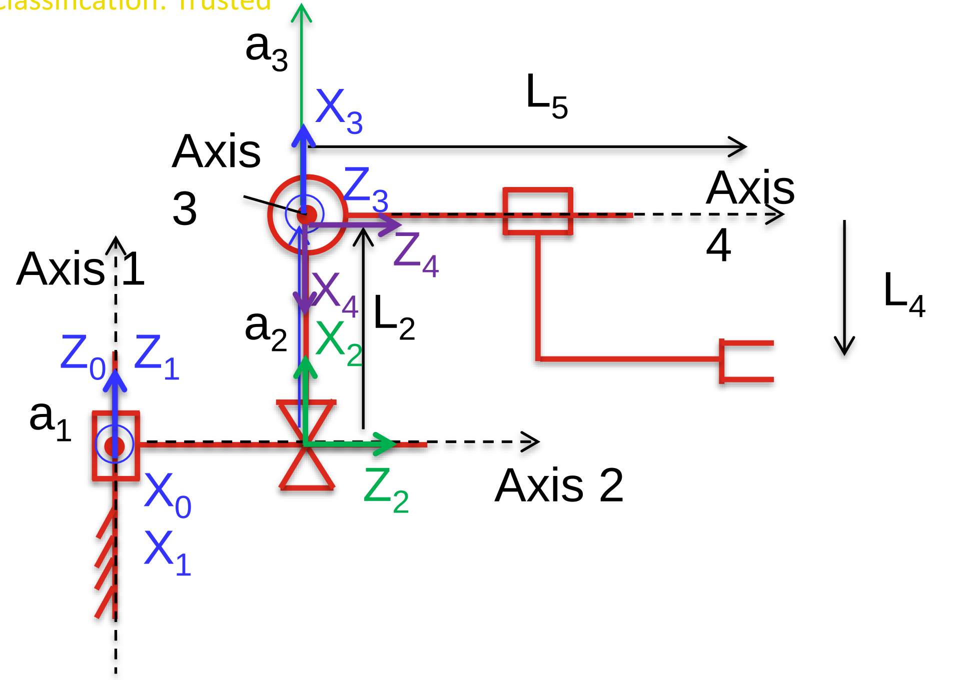
$$ {}^{3}_{4}T = \begin{bmatrix} c\theta_4 & -s\theta_4 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s\theta_4 & c\theta_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{0}_{3}T = {}^{0}_{1}T \cdot {}^{1}_{2}T \cdot {}^{2}_{3}T \cdot {}^{3}_{4}T $$

$$ = \begin{bmatrix} s_1s_3s_4 + c_1s_4 & -s_1s_3s_4 + c_1c_4 & -s_1c_3 & -d_2s_1 \\ -c_1s_3c_4 + s_1s_4 & c_1s_3s_4 + s_1c_4 & c_1c_3 & d_2c_1 \\ c_3c_4 & -c_3s_4 & s_3 & L_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Example 4

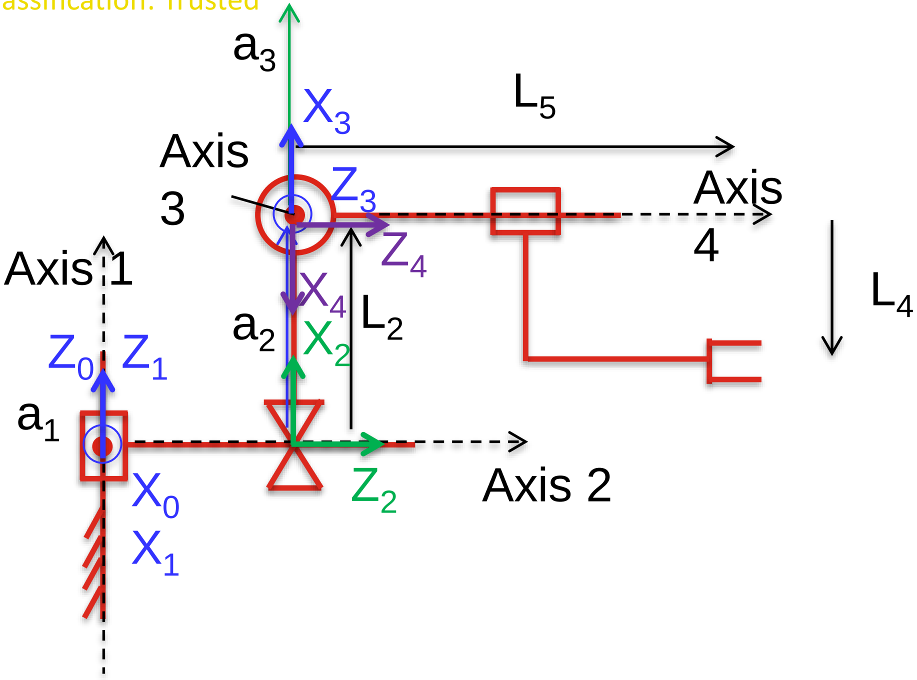
- Verification:
- The end-effector, with reference to frame {4}, has position $[L_4, 0, L_5]^T$, and same orientation as frame {4}. Therefore:



$$
{}^{0}P = {}^{0}_{4}T \cdot {}^{4}P
$$

$$
= \begin{bmatrix} s_1 s_3 s_4 + c_1 s_4 & -s_1 s_3 s_4 + c_1 c_4 & -s_1 c_3 & -d_2 s_1 \\ -c_1 s_3 c_4 + s_1 s_4 & c_1 s_3 s_4 + s_1 c_4 & c_1 c_3 & d_2 c_1 \\ c_3 c_4 & -c_3 s_4 & s_3 & L_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} L_4 \\ 0 \\ L_5 \\ 1 \end{bmatrix} = \begin{bmatrix} L_4(s_1 s_3 s_4 + c_1 s_4) - L_5 s_1 c_3 - d_2 s_1 \\ L_4(-c_1 s_3 c_4 + s_1 s_4) + L_5 c_1 c_3 + d_2 c_1 \\ L_4 c_3 c_4 + L_5 s_3 + L_2 \\ 1 \end{bmatrix}
$$

Example 4



- Verify this for $\theta_1 = 0$, $\theta_2 = -90$ (fixed), $\theta_3 = 0$, $\theta_4 = 180$ as shown in figure:

$$ {}^{0}P = \begin{bmatrix} 0 \\ L_5 + d_2 \\ -L_4 + L_2 \\ 1 \end{bmatrix} $$

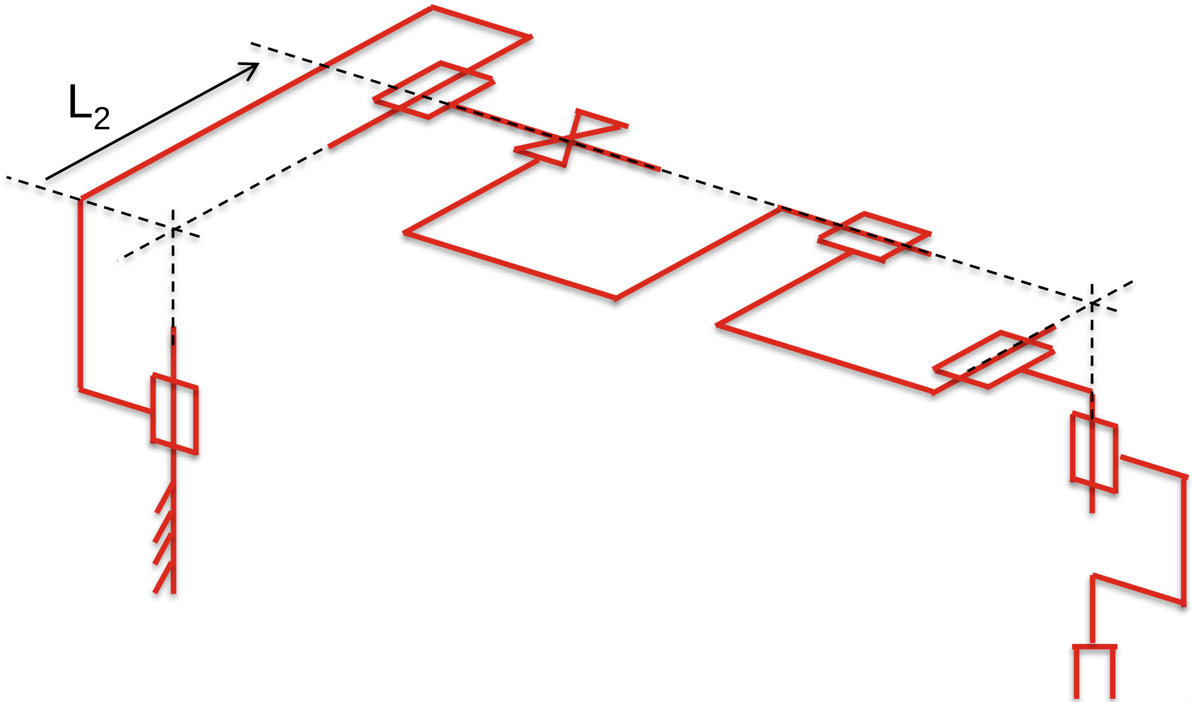
- It looks correct!
- Also, if we rotate {4} such that it aligns with {3}, i.e. $\theta_1 = 0$, $\theta_2 = -90$ (fixed), $\theta_3 = 0$, $\theta_4 = 0$, we would have:

$$ {}^{0}P = \begin{bmatrix} 0 \\ L_5 + d_2 \\ L_4 + L_2 \\ 1 \end{bmatrix} $$

- Reminder: Note the definition of θ!

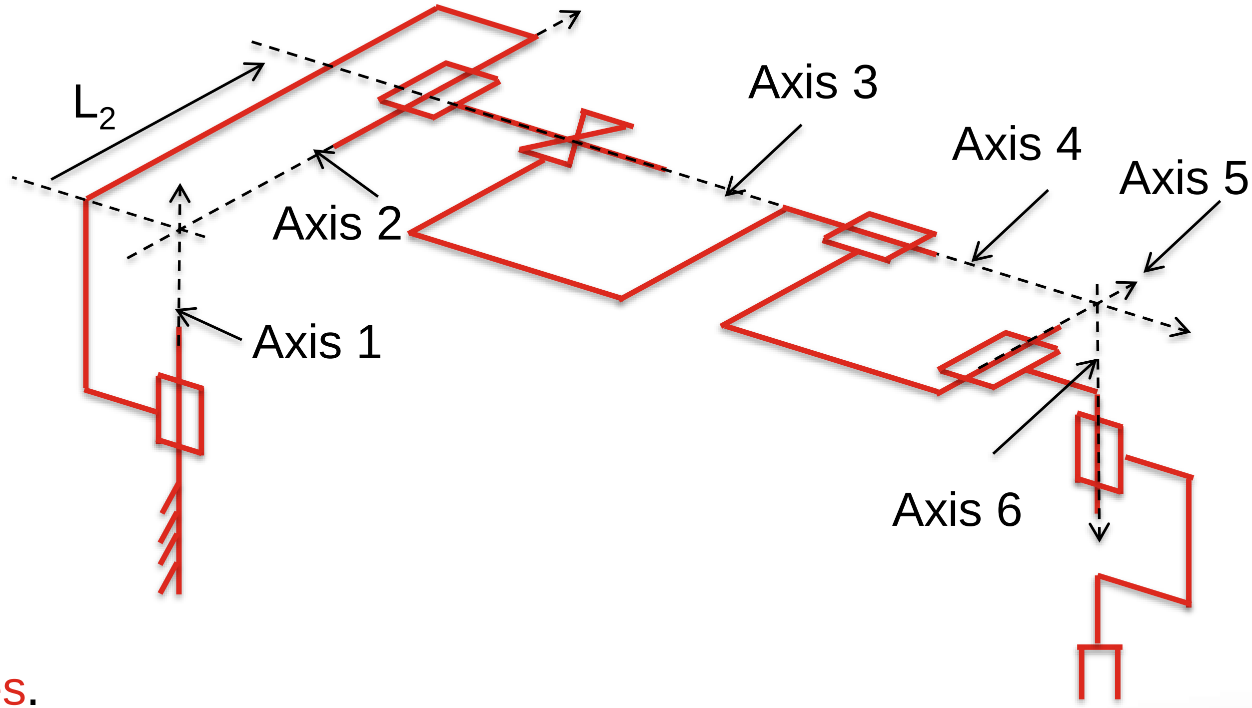
Example 5

- 6-link Stanford Scheinman robot:



Example 5

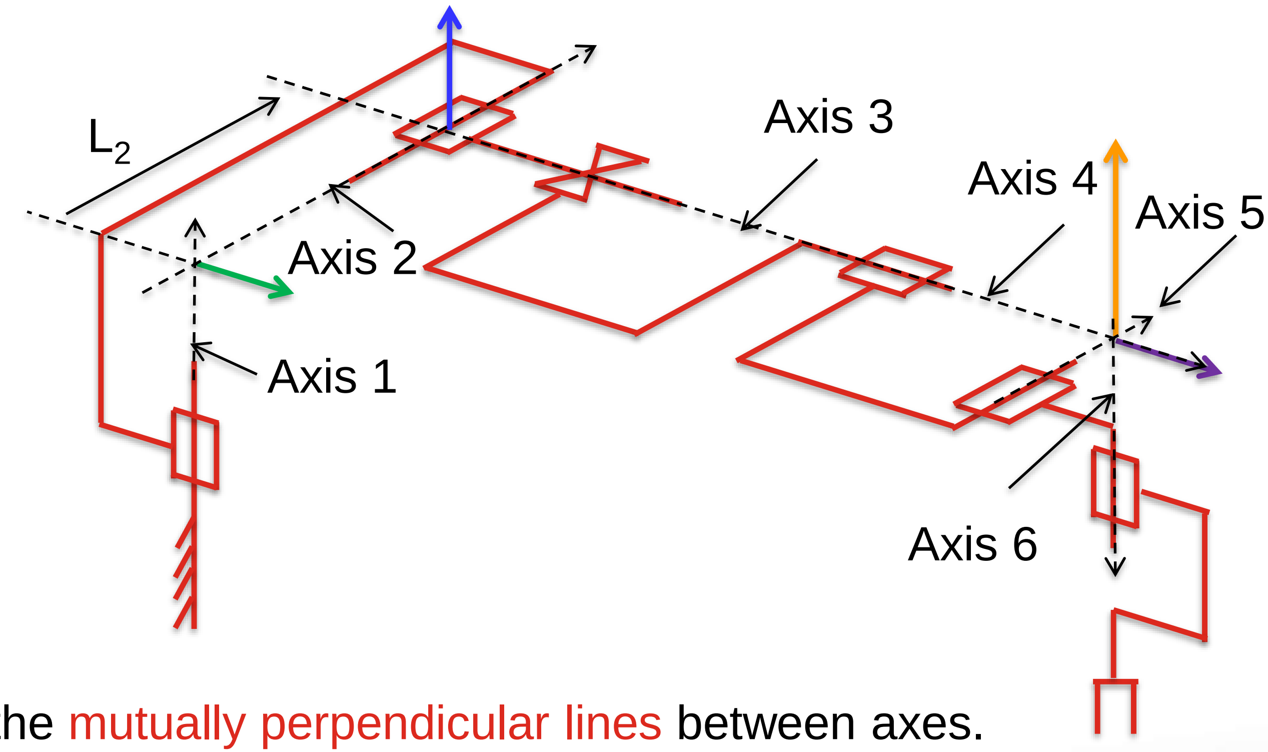
- 6-link Stanford Scheinman robot:



- Step 1, draw the axes.
    - For rotary joint: About the rotation
    - For prismatic joint: Along the translation

Example 5

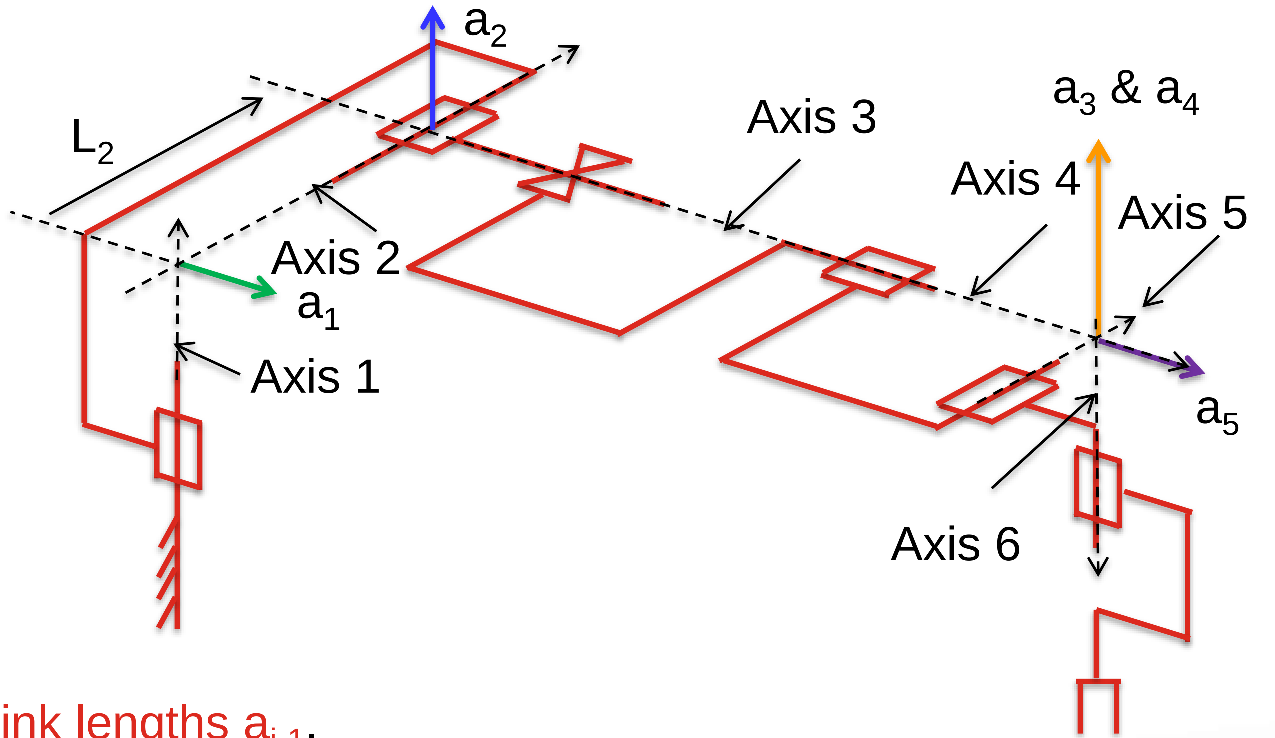
- 6-link Stanford Scheinman robot:



- Step 2, we draw the mutually perpendicular lines between axes.
  - Draw from lower axis to higher axis, e.g. 1 to 2, 2 to 3…
  - Axis 1 and Axis 2 intersect → Green arrow.
  - Axis 2 and Axis 3 intersect → Blue arrow.
  - Axis 3 and Axis 4 are the same → Arbitrary, Orange arrow (same as Axis 4 and 5 below).
  - Axis 4 and Axis 5 intersect → Orange arrow.
  - Axis 5 and Axis 6 intersect → Purple arrow.

Example 5

- 6-link Stanford Scheinman robot:



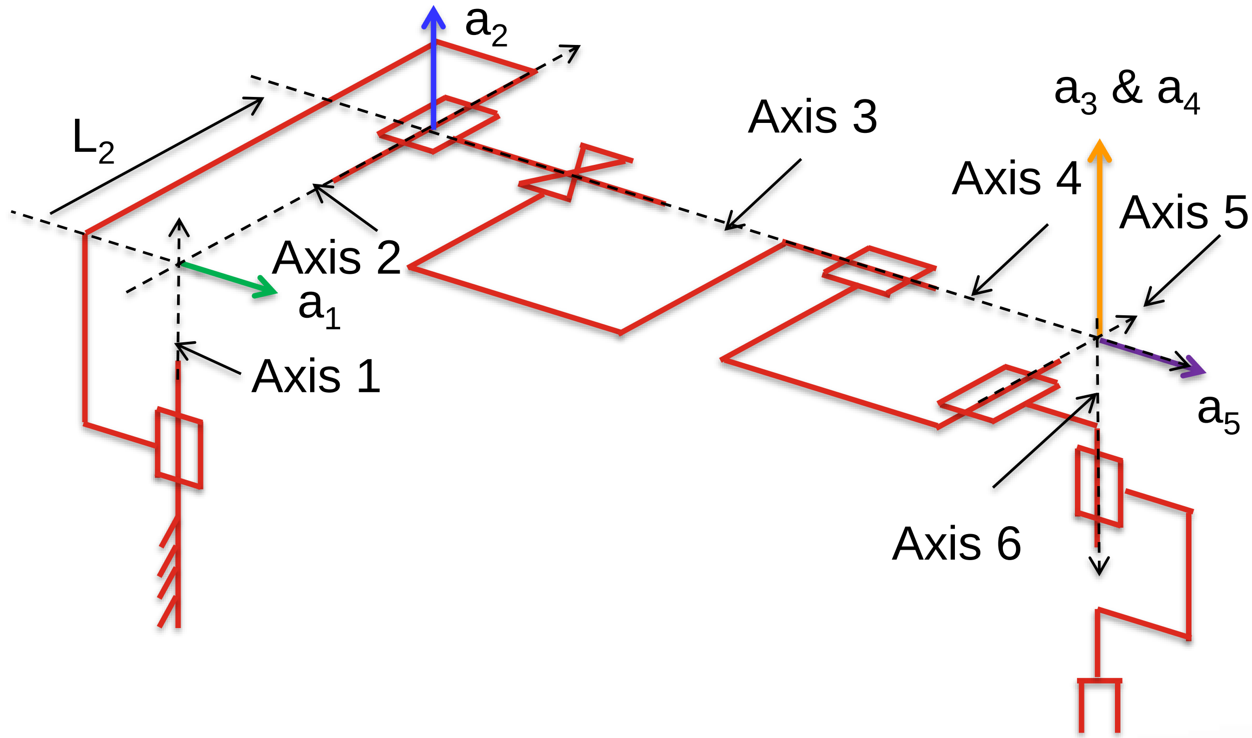
- Step 3, put in the link lengths a_{i-1}.
- Definition: a_{i-1} = length of mutual perpendicular, from axis i-1 to axis i.

Do 1 to n-1

- a_1 = length of mutual perpendicular from axis 1 to 2 = 0 (intersect).
- a_2 = length of mutual perpendicular from axis 2 to 3 = 0 (intersect).
- a_3 = length of mutual perpendicular from axis 3 to 4 = 0 (intersect).
- a_4 = length of mutual perpendicular from axis 4 to 5 = 0 (intersect).
- a_5 = length of mutual perpendicular from axis 5 to 6 = 0 (intersect).

Example 5

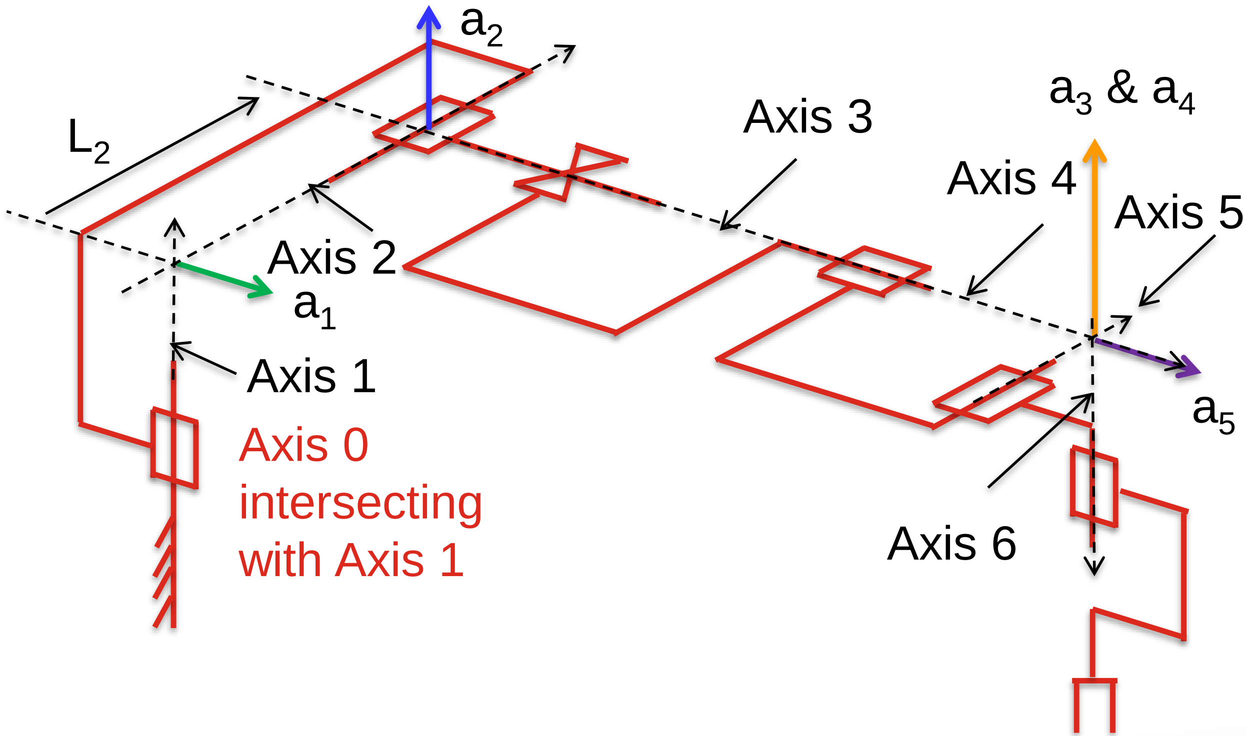
- 6-link Stanford Scheinman robot:



- What about a_0?
  - a_0 = length of mutual perpendicular from axis 0 to 1. However, axis 0 is not known yet.

Example 5

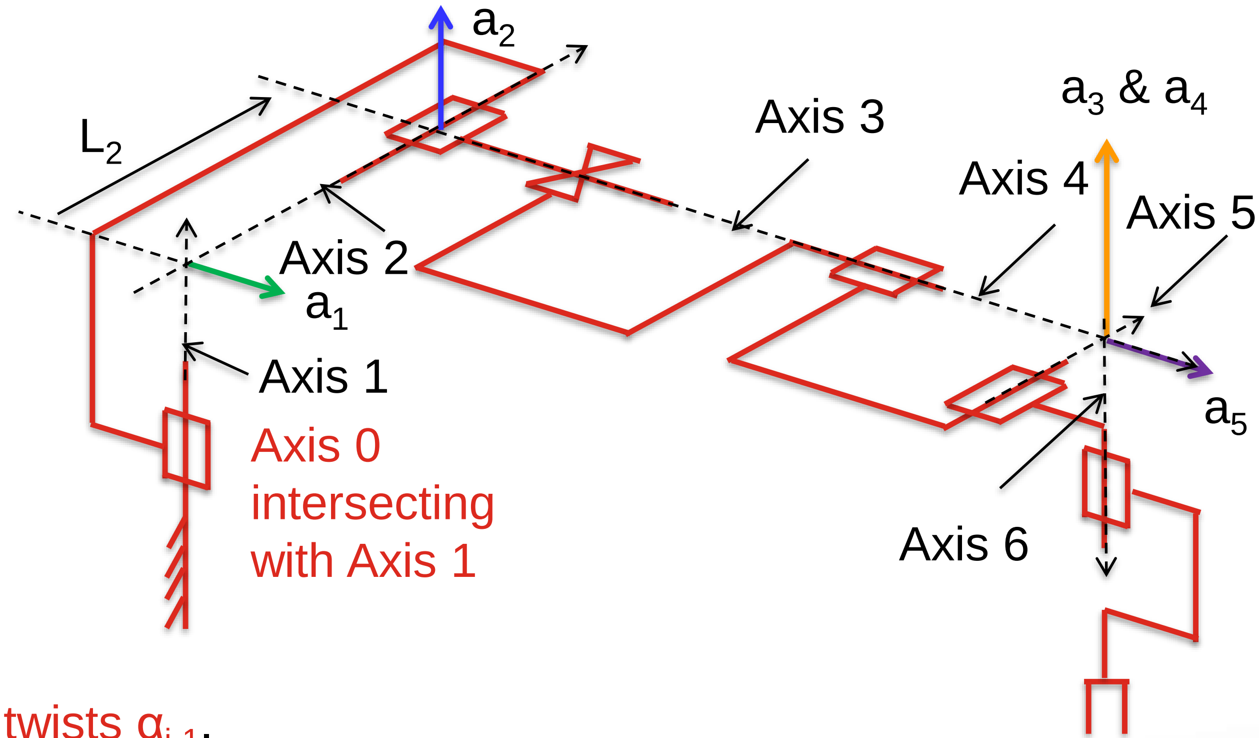
- 6-link Stanford Scheinman robot:



- What about a_0?
  - a_0 = length of mutual perpendicular from axis 0 to 1. However, axis 0 is not known yet.
  - By convention, $a_0 = 0$.
  - This means: Axis 0 and Axis 1 intersect with each other.

Example 5

- 6-link Stanford Scheinman robot:



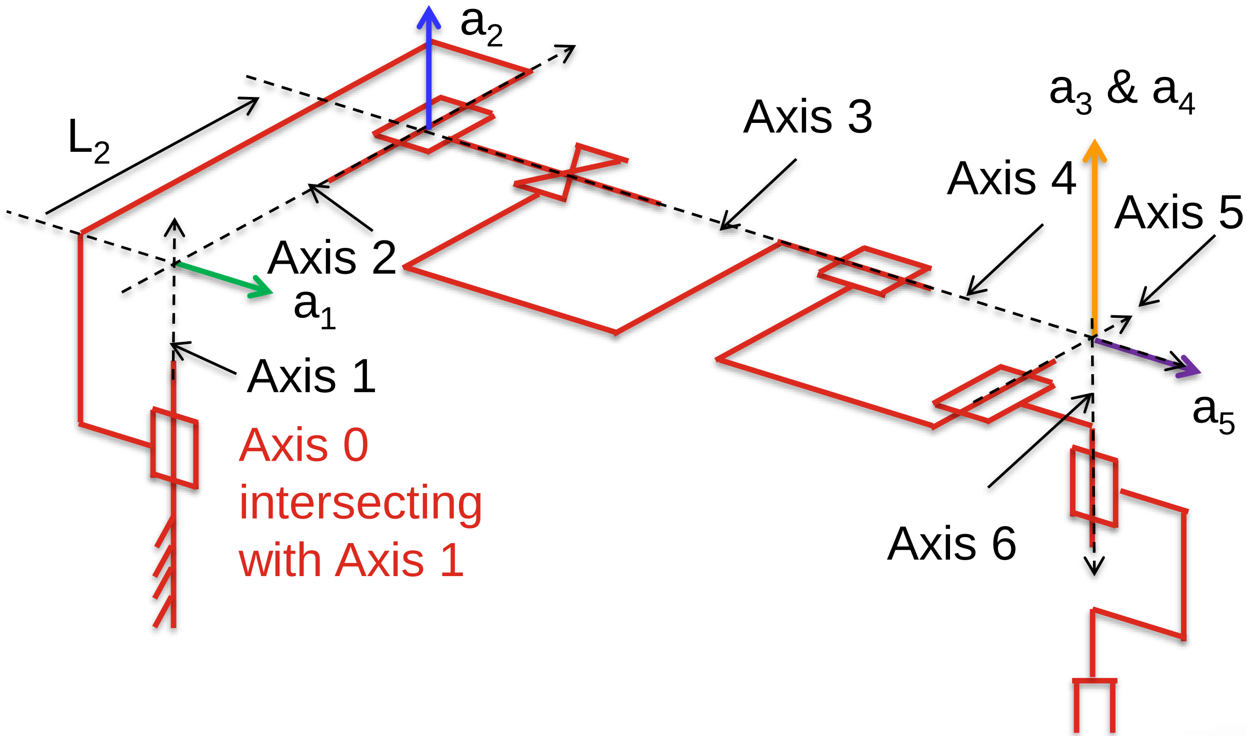
- Step 4, put in link twists α_{i-1}.
- Definition: α_{i-1} = angle between axis i-1 and axis i, in the right hand sense about a_{i-1}

Do 1 to n-1

- α_1 = angle between axis 1 and axis 2, about a_1 = -90deg.
- α_2 = angle between axis 2 and axis 3, about a_2 = -90deg.
- α_3 = angle between axis 3 and axis 4, about a_3 = 0deg.
- α_4 = angle between axis 4 and axis 5, about a_4 = 90deg.
- α_5 = angle between axis 5 and axis 6, about a_5 = -90deg.

Example 5

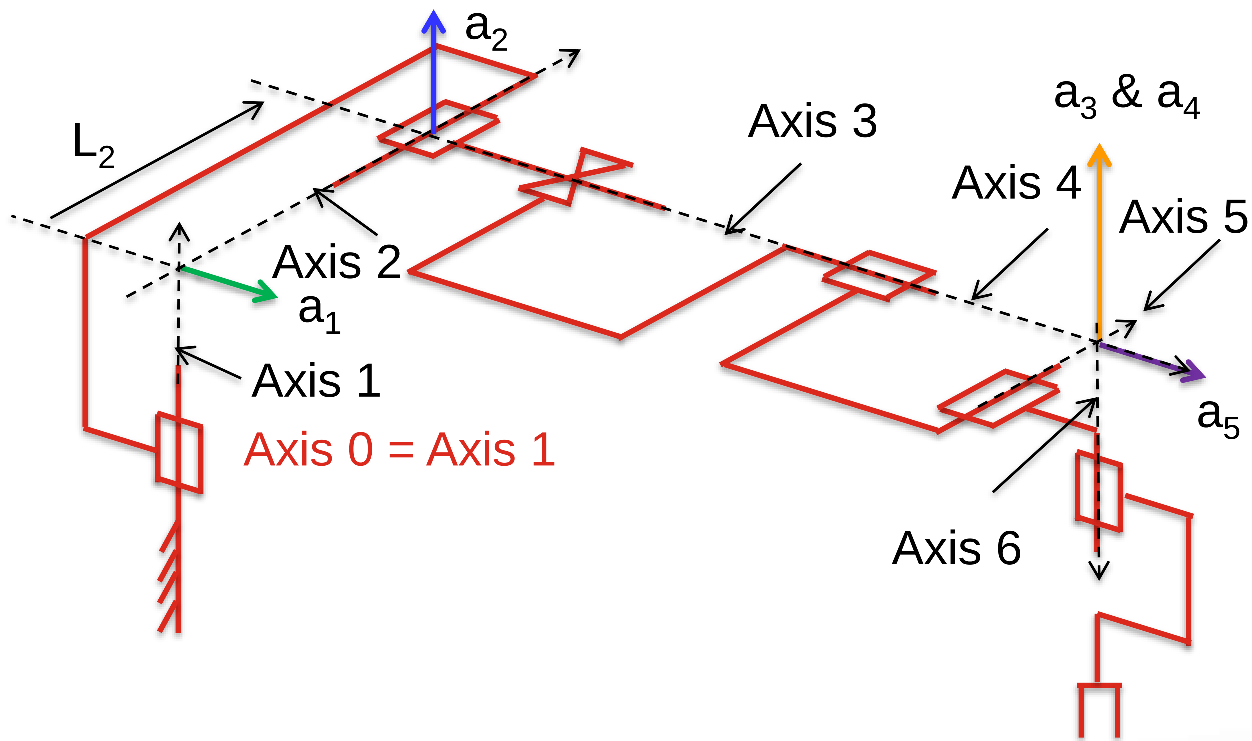
- 6-link Stanford Scheinman robot:



- What about α_0?
  - α_0 = angle between axis 0 and axis 1, about a_0 . However, axis 0 is not fully known yet.

Example 5

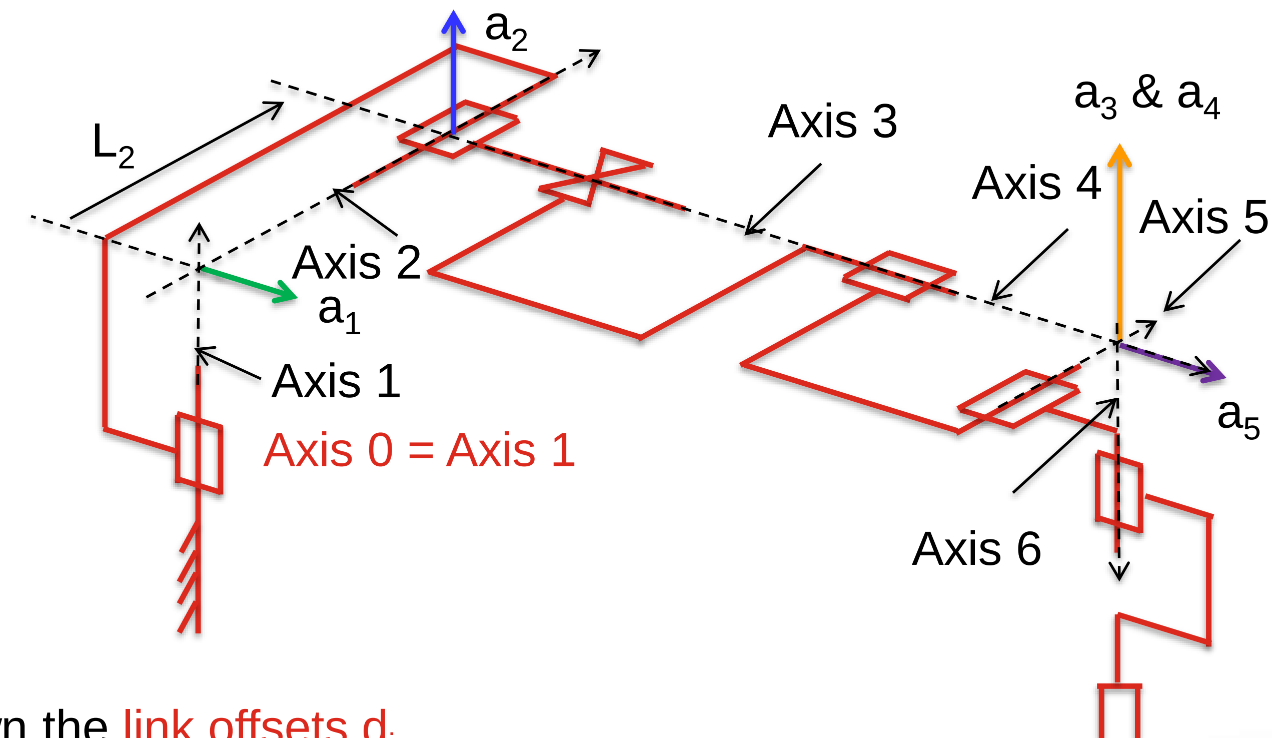
- 6-link Stanford Scheinman robot:



- What about α_0?
  - α_0 = angle between axis 0 and axis 1, about a_0 . However, axis 0 is not fully known yet.
  - By convention, $\alpha_0 = 0$.
  - This means: Axis 0 and Axis 1 are the same.

Example 5

- 6-link Stanford Scheinman robot:



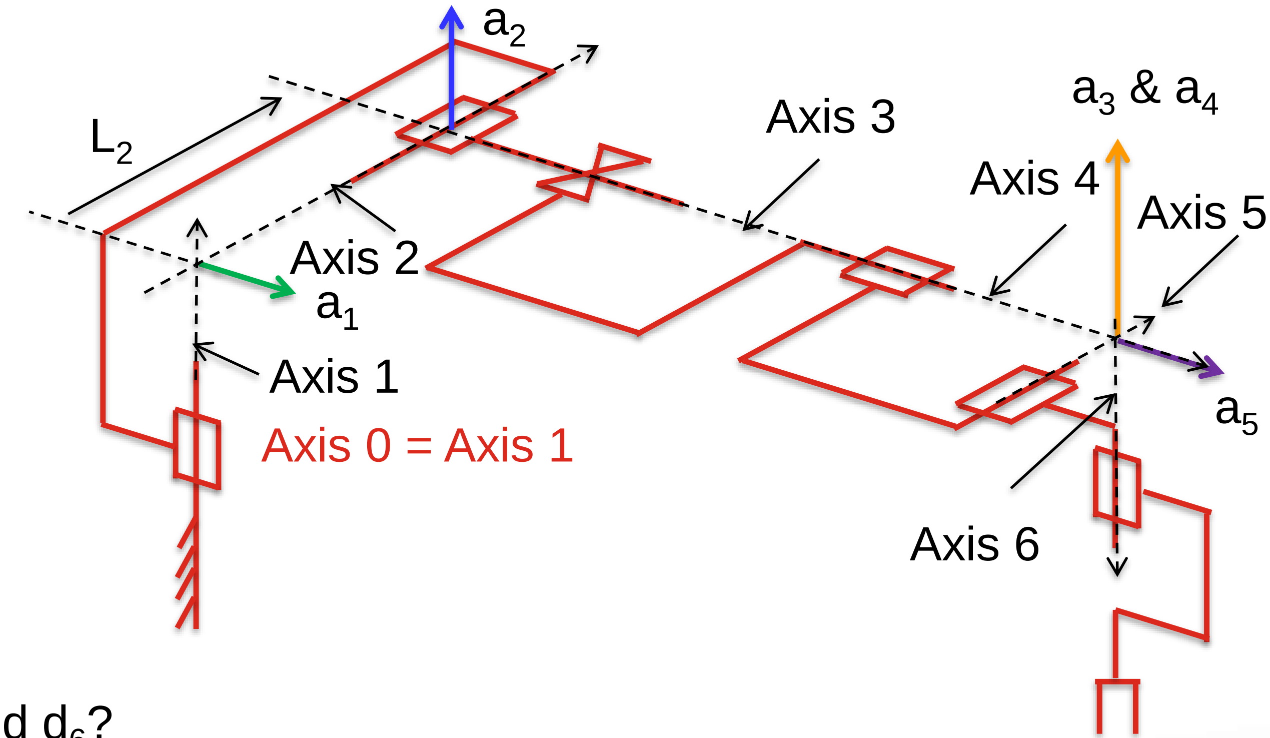
- Step 5, write down the link offsets d_i.
- Definition: d_i = distance from a_{i-1} to a_i, along axis i.

Do 2 to n-1

- d_2 = distance from a1 to a2, along axis 2, is L_2.
- d_3 = distance from a2 to a3, along axis 3, is a variable.
- d_4 = distance from a3 to a4, along axis 4, is 0.
- d_5 = distance from a4 to a5, along axis 5, is 0.

Example 5

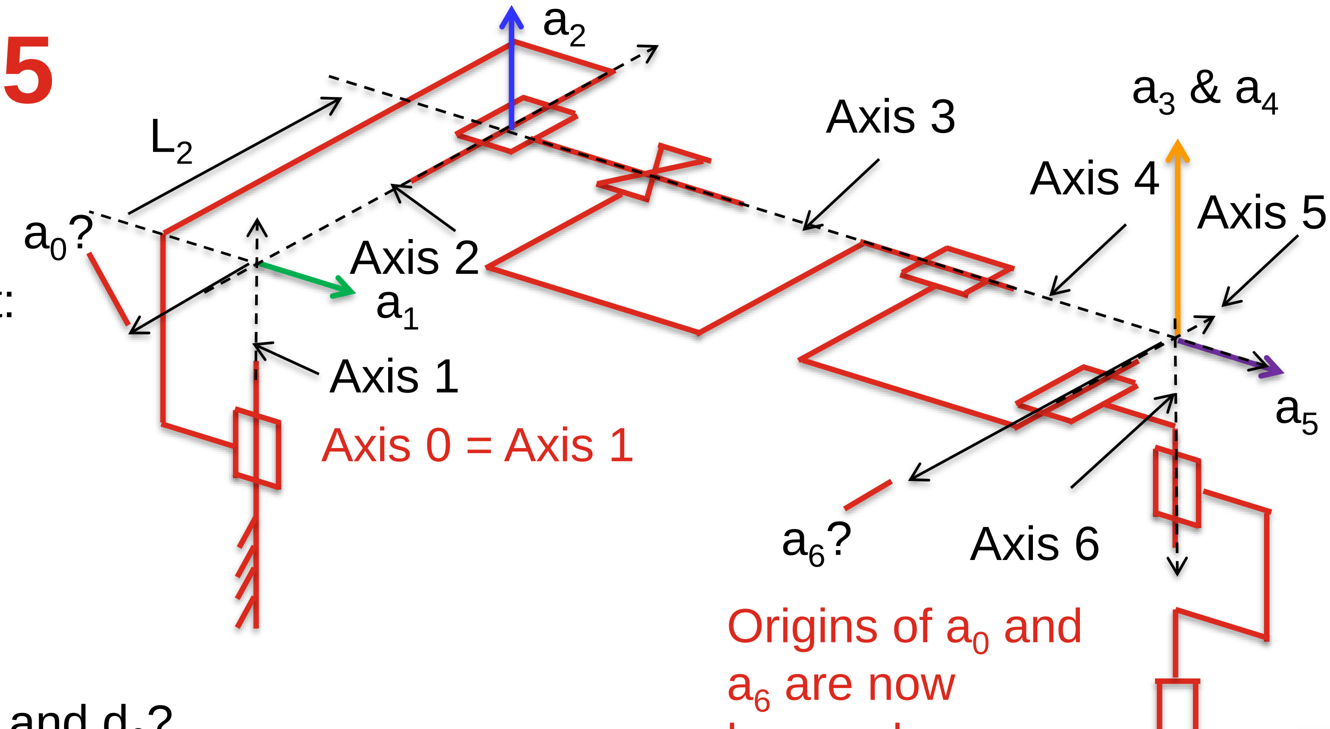
- 6-link Stanford Scheinman robot:



- What about d_1 and d_6?
  - d_1 = distance from a_0 to a_1, along axis 1.
  - d_6 = distance from a_5 to a_6, along axis 4.
  - But where exactly are a_0 and a_6?

Example 5

- 6-link Stanford Scheinman robot:

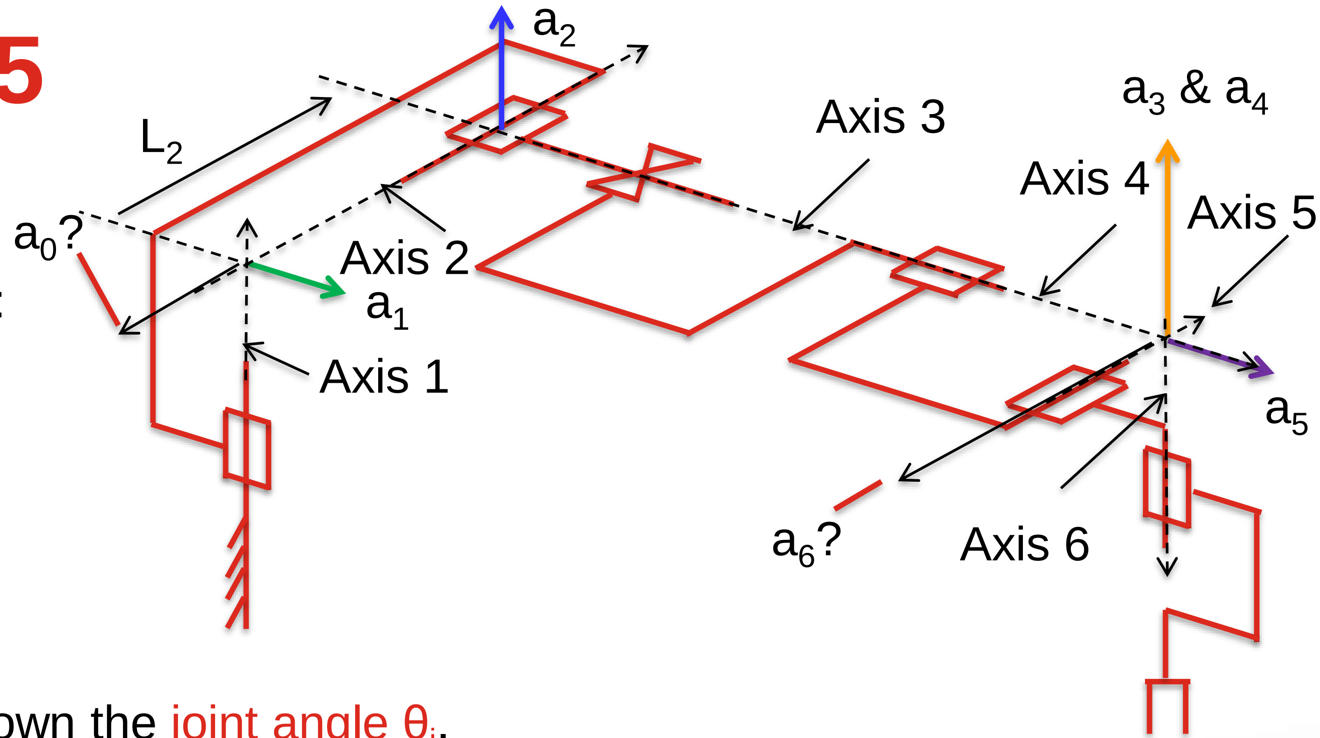


Origins of a_0 and a_6 are now known, because d_1 and d_6 are zero. However, the direction is not known.

- What about d_1 and d_6?
  - d_1 = distance from a_0 to a_1, along axis 1.
  - d_6 = distance from a_5 to a_6, along axis 4.
  - But where exactly are a_0 and a_6?
  - By convention: Zero for revolute joint, variable for prismatic joint.
  - So in this case, d_1 and d_4 are both zero.

Example 5

- 6-link Stanford Scheinman robot:



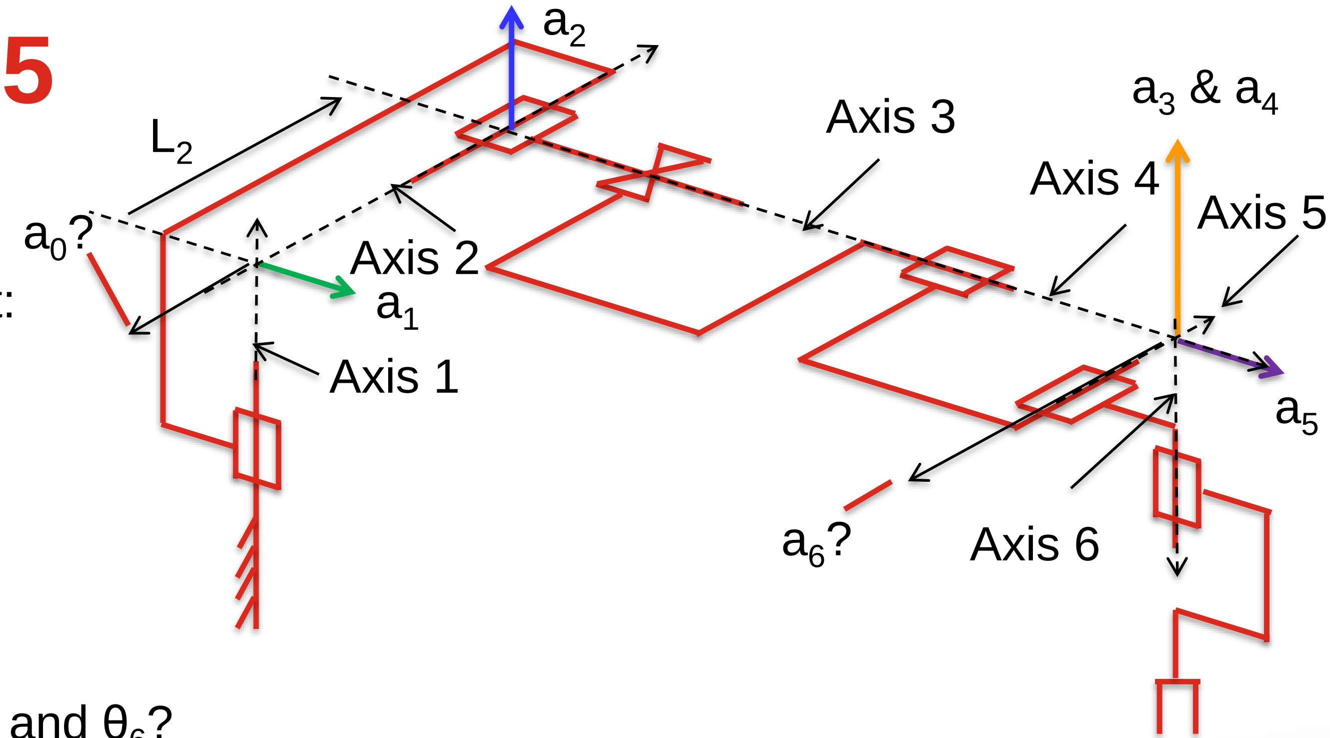
- Step 6, write down the joint angle θ_i.
- Definition: θ_i is the angle between the (extension of a_{i-1}) and a_i, measured about the axis i.

Do 2 to n-1

- θ_2 = angle between (extension of a1) and a2, about axis 2 = var.
- θ_3 = angle between (extension of a2) and a3, about axis 3 = 0deg.
- θ_4 = angle between (extension of a3) and a4, about axis 4 = var.
- θ_5 = angle between (extension of a4) and a5, about axis 5 = var.

Example 5

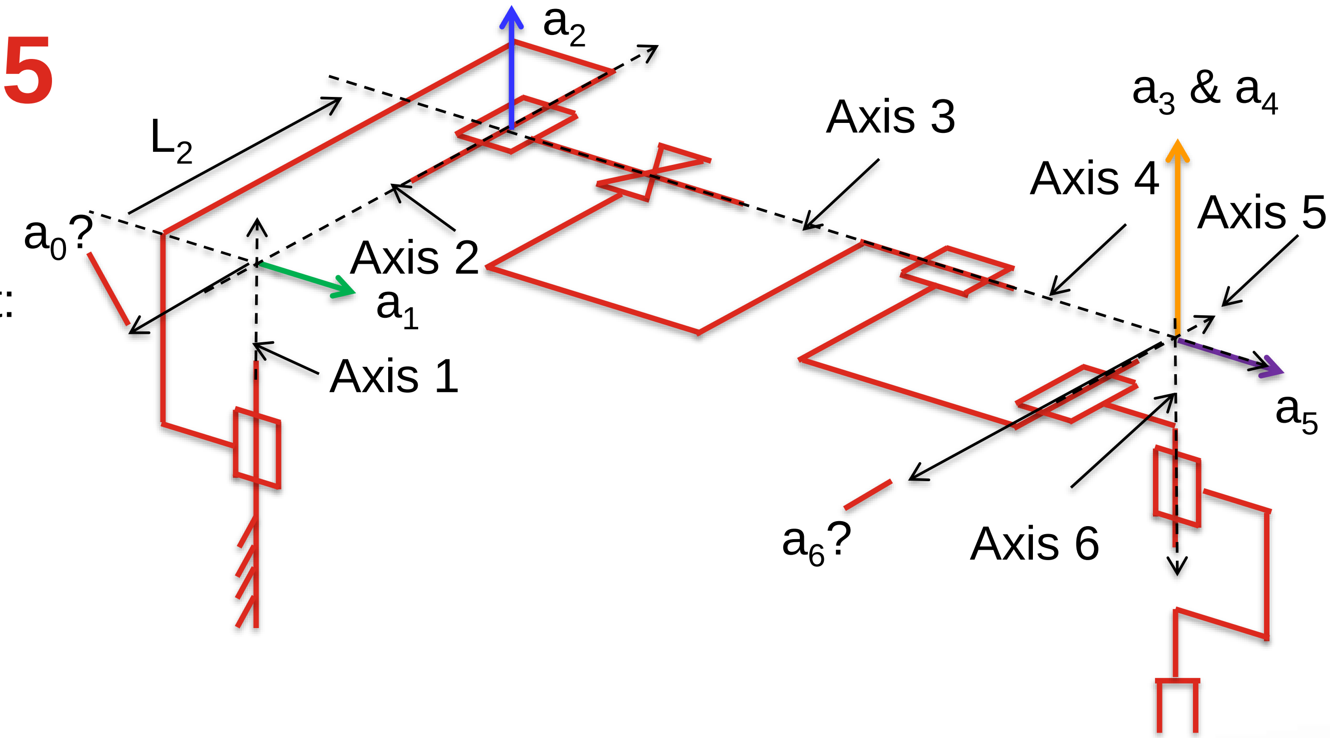
- 6-link Stanford Scheinman robot:



- What about θ_1 and θ_6?
  - θ_1 = angle between (extension of a0) and a1, about axis 1.
  - θ_6 = angle between (extension of a5) and a6, about axis 6.
  - By convention: Zero for prismatic joint, variable for revolute joint.
  - So in this case, θ_1 and θ_6 are both variables.

Example 5

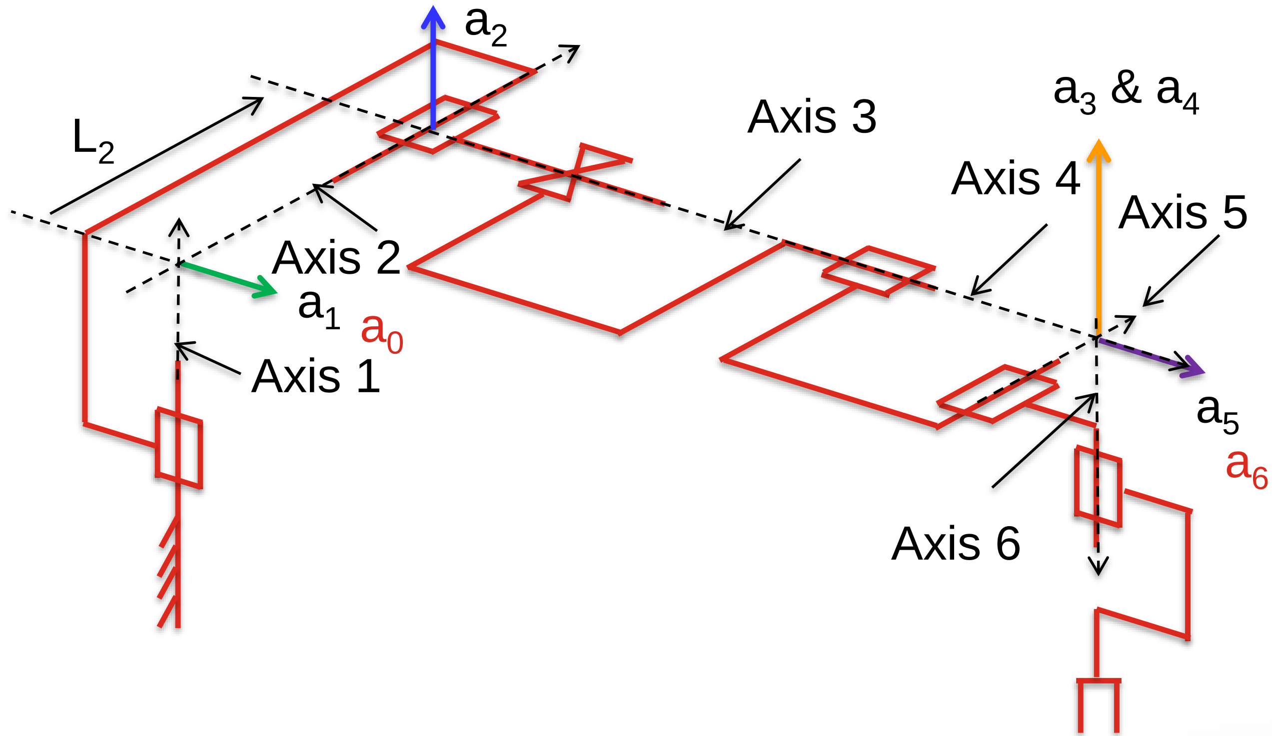
- 6-link Stanford Scheinman robot:



- We still have a problem. Since θ_1 and θ_6 are both variables, we need to determine their “zero”-angle position.

Example 5

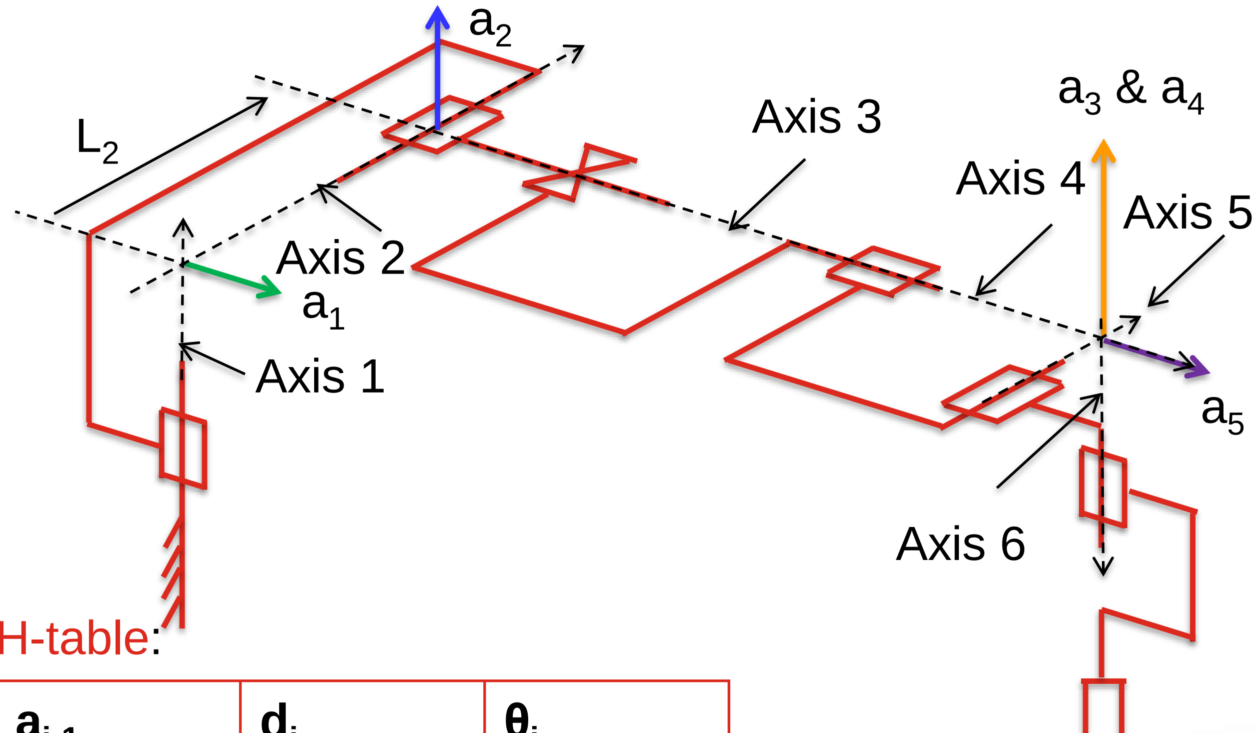
- 6-link Stanford Scheinman robot:



- For convenience, align a_0 with a_1 when the joint variable 1 is zero.
- As for joint n:
  - Revolute: align a_n with a_{n-1} when $\theta_n = 0$.
  - Prismatic: align a_n with a_{n-1} when $d_n = 0$.

Example 5

- 6-link Stanford Scheinman robot:

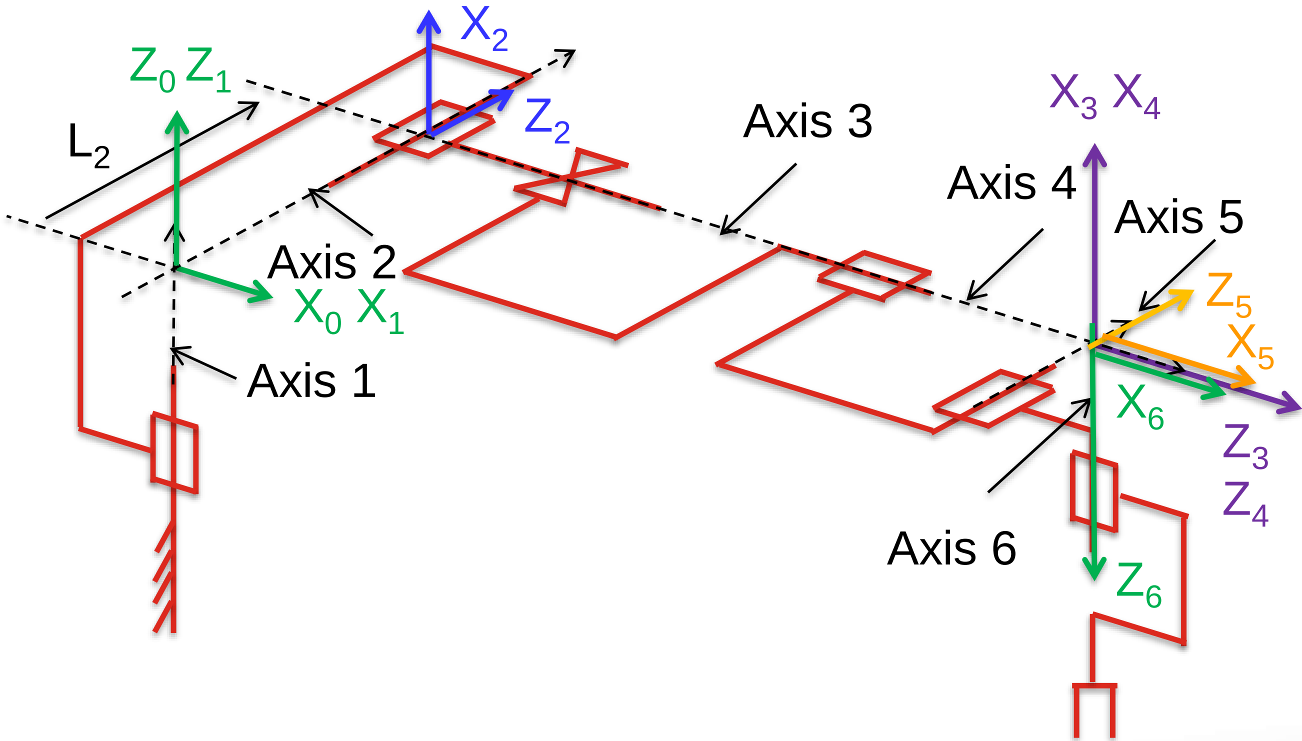


- Step 7, transfer to DH-table:

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	-90	0	L_2	θ_2
3	-90	0	d_3	0
4	0	0	0	θ_4
5	90	0	0	θ_5
6	-90	0	0	θ_6

Example 5

- 6-link Stanford Scheinman robot:



- Step 8, insert the frames. Rules:
  - Z-axis of frame {i}, i.e. Z_i, is coincident with joint axis i.
  - Origin of frame {i} is where the a_i intersects the joint i axis.
  - X-axis of frame {i}, i.e. X_i, is coincident with a_i.

Example 5

- Step 9 (Final step!), calculate the transformations.

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	-90	0	L_2	θ_2
3	-90	0	d_3	0
4	0	0	0	θ_4
5	90	0	0	θ_5
6	-90	0	0	θ_6

$$ {}^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ c\alpha_{i-1}s\theta_i & c\alpha_{i-1}c\theta_i & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\alpha_{i-1}s\theta_i & s\alpha_{i-1}c\theta_i & c\alpha_{i-1} & c\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{1}_{2}T = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & 0 \\ 0 & 0 & 1 & L_2 \\ -s\theta_2 & -c\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{2}_{3}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Example 5

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	-90	0	L_2	θ_2
3	-90	0	d_3	0
4	0	0	0	θ_4
5	90	0	0	θ_5
6	-90	0	0	θ_6

$$ {}^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ c\alpha_{i-1}s\theta_i & c\alpha_{i-1}c\theta_i & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\alpha_{i-1}s\theta_i & s\alpha_{i-1}c\theta_i & c\alpha_{i-1} & c\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{3}_{4}T = \begin{bmatrix} c\theta_4 & -s\theta_4 & 0 & 0 \\ s\theta_4 & c\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{4}_{5}T = \begin{bmatrix} c\theta_5 & -s\theta_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s\theta_5 & c\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{5}_{6}T = \begin{bmatrix} c\theta_6 & -s\theta_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_6 & -c\theta_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Example 5

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	-90	0	L_2	θ_2
3	-90	0	d_3	0
4	0	0	0	θ_4
5	90	0	0	θ_5
6	-90	0	0	θ_6

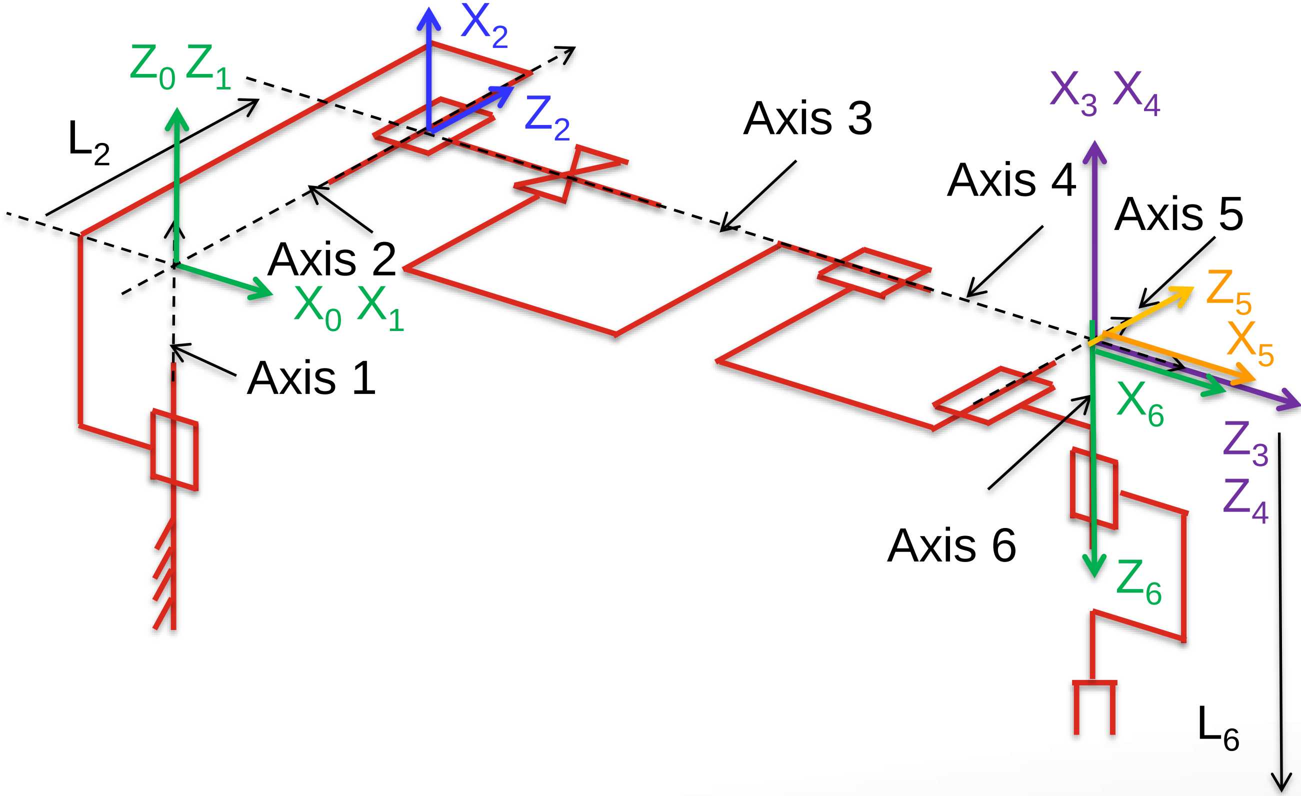
$$ {}^{i-1}_{\ \ i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ c\alpha_{i-1}s\theta_i & c\alpha_{i-1}c\theta_i & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\alpha_{i-1}s\theta_i & s\alpha_{i-1}c\theta_i & c\alpha_{i-1} & c\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^0_6T = {}^0_1T \cdot {}^1_2T \cdot {}^2_3T \cdot {}^3_4T \cdot {}^4_5T \cdot {}^5_6T $$

$$ = \begin{bmatrix} c_1c_2 & s_1 & -c_1s_2 & -d_3c_1s_2 - L_2s_1 \\ s_1c_2 & -c_1 & -s_1s_2 & -d_3s_1s_2 + L_2c_1 \\ -s_2 & 0 & -c_2 & -d_3c_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_4c_5c_6 - s_4s_6 & -c_4c_5c_6 - s_4c_6 & -c_4s_5 & 0 \\ s_4c_5c_6 + c_4s_6 & -s_4c_5s_6 + c_4c_6 & -s_4s_5 & 0 \\ s_5c_6 & -s_5s_6 & c_5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

RMIT UNIVERSITY

Example 5

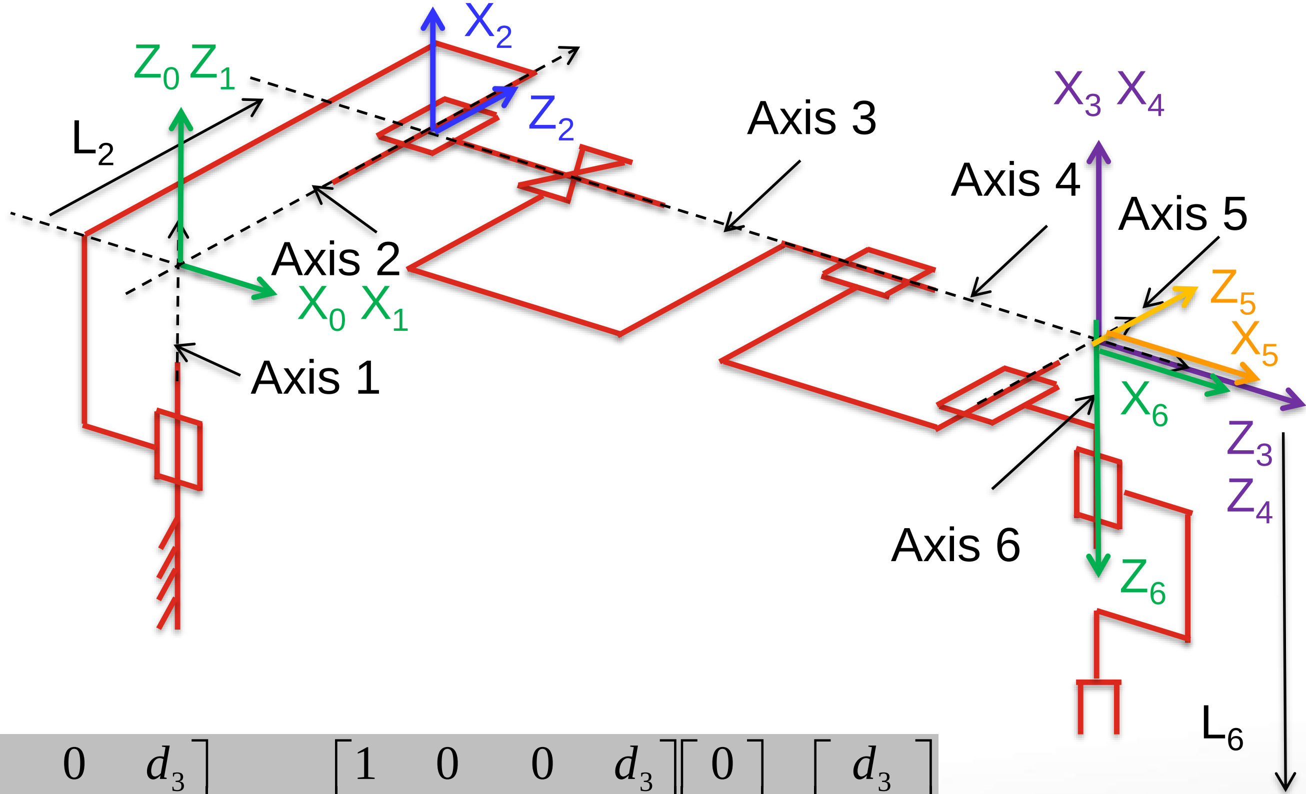


- Check if this is correct, for the configuration as shown, where:
  - $\theta_1 = 0$;
  - $\theta_2 = -90$;
  - $\theta_3 = 0$;
  - $\theta_4 = 0$;
  - $\theta_5 = 90$;
  - $\theta_6 = 0$;
- Therefore:

$$ {}^0_6T = {}^0_1T \cdot {}^1_2T \cdot {}^2_3T \cdot {}^3_4T \cdot {}^4_5T \cdot {}^5_6T $$

$$ = \begin{bmatrix} 0 & 0 & 1 & d_3 \\ 0 & -1 & 0 & L_2 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & d_3 \\ 0 & -1 & 0 & L_2 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Example 5



- The end-effector has position $[0, 0, L_6]^T$ with respect to frame {6}.
- Therefore:

$$
{}^{0}P = \begin{bmatrix} 1 & 0 & 0 & d_3 \\ 0 & -1 & 0 & L_2 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot {}^{6}P = \begin{bmatrix} 1 & 0 & 0 & d_3 \\ 0 & -1 & 0 & L_2 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ L_6 \\ 1 \end{bmatrix} = \begin{bmatrix} d_3 \\ L_2 \\ -L_6 \\ 1 \end{bmatrix}
$$

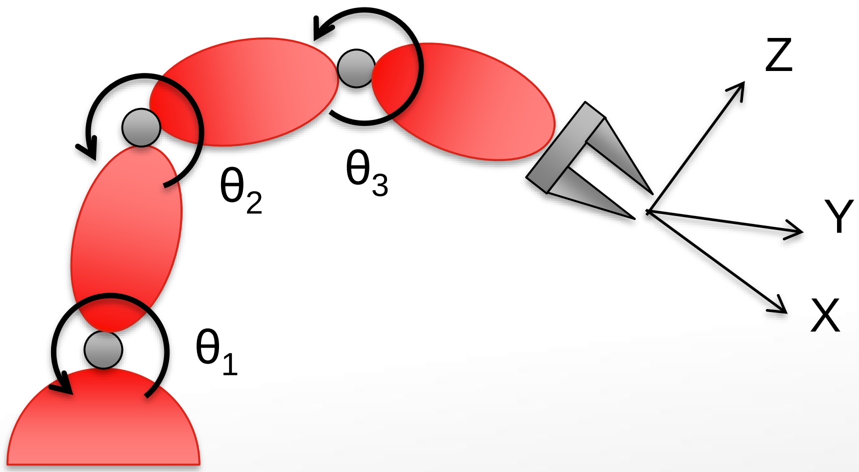
- This looks correct from the figure.

Content

- Forward Kinematics
  - Introduction
  - Denavit-Hartenberg Parameters
  - Introduction to DH Parameters, Link Frame Attachment, Forward Kinematics through Example 1
  - More Examples
- Inverse Kinematics
  - Introduction
  - Algebraic Approach
  - Geometric Approach

Introduction

- In previous section, we studied the following problem:
  - Given the joint space parameters (angles for revolute joints, or offsets for prismatic joints), as well as the lengths of the links, what is the position and orientation of the end-effector in Cartesian space?



- Now, we will look at the inverse problem (much more difficult!):
  - Given the desired position and orientation of the tool in Cartesian space, what is the set of joint angles that is required to achieve the desired outcome?

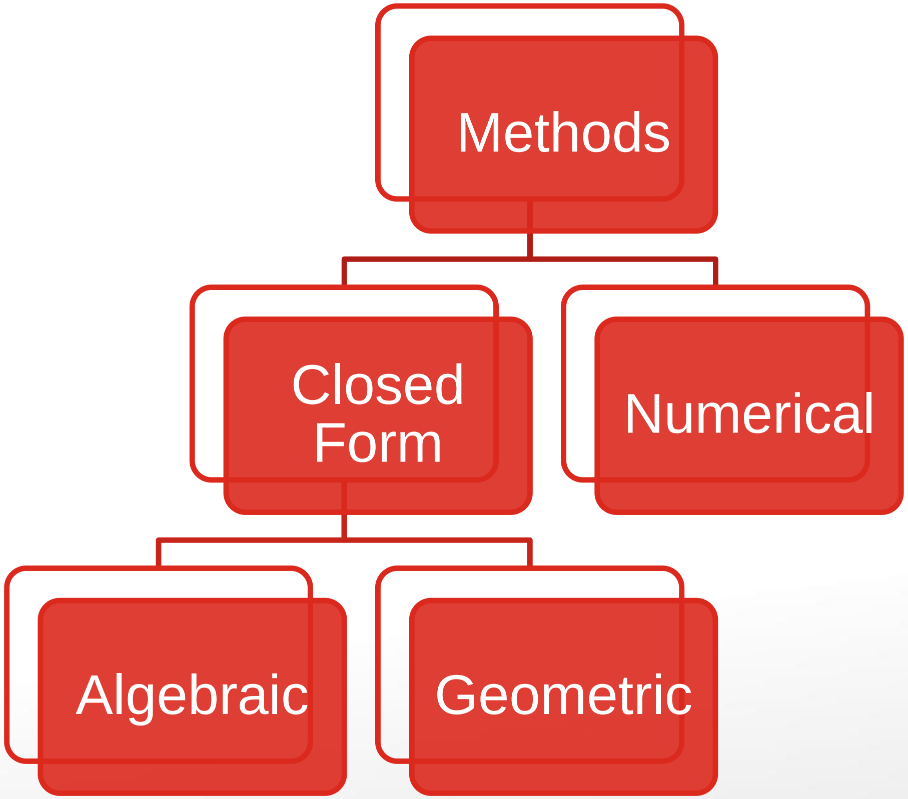
Introduction

- For example, consider a 6-link robot.
- The homogeneous transform from base to link 6 is:

$$
{}^0_6T = \underbrace{\begin{bmatrix} f_{11}(q_1,\cdots,q_n) & f_{12}(q_1,\cdots,q_n) & f_{13}(q_1,\cdots,q_n) & f_{14}(q_1,\cdots,q_n) \\ f_{21}(q_1,\cdots,q_n) & f_{22}(q_1,\cdots,q_n) & f_{23}(q_1,\cdots,q_n) & f_{24}(q_1,\cdots,q_n) \\ f_{31}(q_1,\cdots,q_n) & f_{32}(q_1,\cdots,q_n) & f_{33}(q_1,\cdots,q_n) & f_{34}(q_1,\cdots,q_n) \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\text{General}} = \underbrace{\begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\text{Particular / Wanted}}
$$

  - There are 12 non-trivial values in the matrix.
  - From the 9 values related to rotation, only 3 are independent.
  - And we have 3 values related to position.
  - Therefore, there are altogether 6 values / equations.
- From the homogeneous transform, we would like to find the 6 joint angles.
- 6 equations and 6 unknowns ☺
- However, it is not easy to solve…

Methods of Solutions



- We will only look at the closed-form solutions.
- Note: There is no general solution. Every robot has to be analysed in a case-by-case basis.

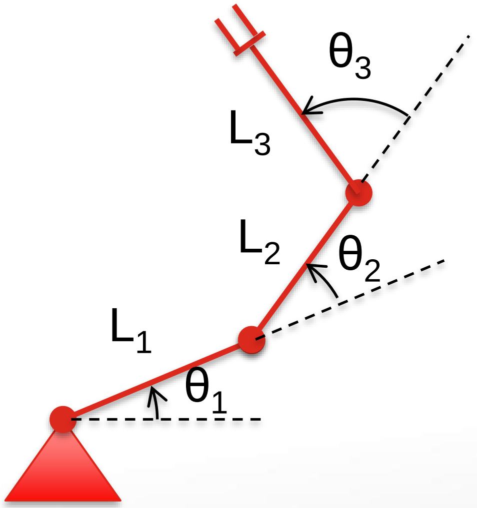
Content

- Forward Kinematics
  - Introduction
  - Denavit-Hartenberg Parameters
  - Introduction to DH Parameters, Link Frame Attachment, Forward Kinematics through Example 1
  - More Examples
- Inverse Kinematics
  - Introduction
  - Algebraic Approach
  - Geometric Approach

Algebraic Solutions

- Because there is no general algorithm to solve the inverse kinematic problems, we will only show an example to present the idea.
- E.g. 3-link RRR manipulator.

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	0	L_1	0	θ_2
3	0	L_2	0	θ_3



- The general transformation from frame {0} to frame {3} was:

$$ {}^0_3T = {}^0_1T \cdot {}^1_2T \cdot {}^2_3T = \begin{bmatrix} c\theta_{123} & -s\theta_{123} & 0 & L_2c\theta_{12} + L_1c\theta_1 \\ s\theta_{123} & c\theta_{123} & 0 & L_2s\theta_{12} + L_1s\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Algebraic Solutions

- Assume we want to put the end-effector at positions [x, y, 0]^T with orientation Φ.
- Thus the specific transformation is:

$$ {}^{0}_{3}T = \begin{bmatrix} c_\phi & -s_\phi & 0 & x \\ s_\phi & c_\phi & 0 & y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

- By comparing the general transform and the specific transform, we see that there are four equations and three unknowns:
  - Equations

$$ c_\phi = c_{123} $$
$$ s_\phi = s_{123} $$
$$ x = L_1 c_1 + L_2 c_{12} $$
$$ y = L_1 s_1 + L_2 s_{12} $$

Unknowns:

$$ \theta_1 $$
$$ \theta_2 $$
$$ \theta_3 $$

Algebraic Solutions

- First, square both x and y equations and add them:

$$
\begin{aligned}
x^2 &= L_1^2 c_1^2 + 2L_1L_2c_1c_{12} + L_2^2 c_{12}^2 \\
y^2 &= L_1^2 s_1^2 + 2L_1L_2s_1s_{12} + L_2^2 s_{12}^2 \\
x^2 + y^2 &= L_1^2 + 2L_1L_2\left(c_1c_{12} + s_1s_{12}\right) + L_2^2 \\
&= L_1^2 + 2L_1L_2\cos\left(\theta_1 - \left(\theta_1 + \theta_2\right)\right) + L_2^2 \\
&= L_1^2 + 2L_1L_2c_2 + L_2^2
\end{aligned}
$$

Note:
cos(- θ) = cos(θ)

- Thus: $$c_2 = \frac{x^2 + y^2 - L_1^2 - L_2^2}{2L_1L_2}$$
  - This value must be between -1 and 1 for a solution to exist.
- We also write: $s_2 = \pm\sqrt{1 - c_2^2}$ where c_2 is a value calculated above.

Algebraic Solutions

- With these, we can compute θ_2 using:

$$\theta_2 = \arctan 2(s_2, c_2)$$

- Note: The choice of sign in $s_2 = \pm\sqrt{1-c_2^2}$ corresponds to the "elbow-up" or "elbow-down" solutions.
    - This is an example of multiple solutions.

- Next, we shall try to solve for θ_1.

$$\begin{aligned} x &= L_1 c_1 + L_2 c_{12} = L_1 c_1 + L_2 c_1 c_2 - L_2 s_1 s_2 = (L_1 + L_2 c_2) c_1 - (L_2 s_2) s_1 = K_1 c_1 - K_2 s_1 \\ y &= L_1 s_1 + L_2 s_{12} = L_1 s_1 + L_2 s_1 c_2 + L_2 c_1 s_2 = (L_1 + L_2 c_2) s_1 + (L_2 s_2) c_1 = K_1 s_1 + K_2 c_1 \end{aligned}$$

- where

$$\begin{aligned} K_1 &= L_1 + L_2 c_2 \\ K_2 &= L_2 s_2 \end{aligned}$$

Algebraic Solutions

- Introduce: $$r = +\sqrt{K_1^2 + K_2^2}$$ $$\gamma = \arctan 2(K_2, K_1)$$
- Then K_1 and K_2 can be written as: $$K_1 = r\cos\gamma$$ $$K_2 = r\sin\gamma$$
- With these, x and y can be written as:

$$x = (r\cos\gamma)c_1 - (r\sin\gamma)s_1$$
$$y = (r\cos\gamma)s_1 + (r\sin\gamma)c_1$$

$$\frac{x}{r} = \cos\gamma\cos\theta_1 - \sin\gamma\sin\theta_1 = \cos(\gamma + \theta_1)$$
$$\frac{y}{r} = \cos\gamma\sin\theta_1 + \sin\gamma\cos\theta_1 = \sin(\gamma + \theta_1)$$

- Finally: $$\gamma + \theta_1 = \arctan 2\left(\frac{y}{r}, \frac{x}{r}\right)$$

$$\theta_1 = \arctan 2\left(\frac{y}{r}, \frac{x}{r}\right) - \arctan 2(K_2, K_1)$$

Algebraic Solutions

- Finally, we can solve for θ_3 easily:

$$c_\phi = c_{123}$$
$$s_\phi = s_{123}$$

$$\tan(\theta_1 + \theta_2 + \theta_3) = \frac{s_\phi}{c_\phi}$$

$$\theta_1 + \theta_2 + \theta_3 = \arctan 2(s_\phi, c_\phi)$$

$$\Rightarrow \quad \theta_3 = \arctan 2(s_\phi, c_\phi) - \theta_1 - \theta_2$$

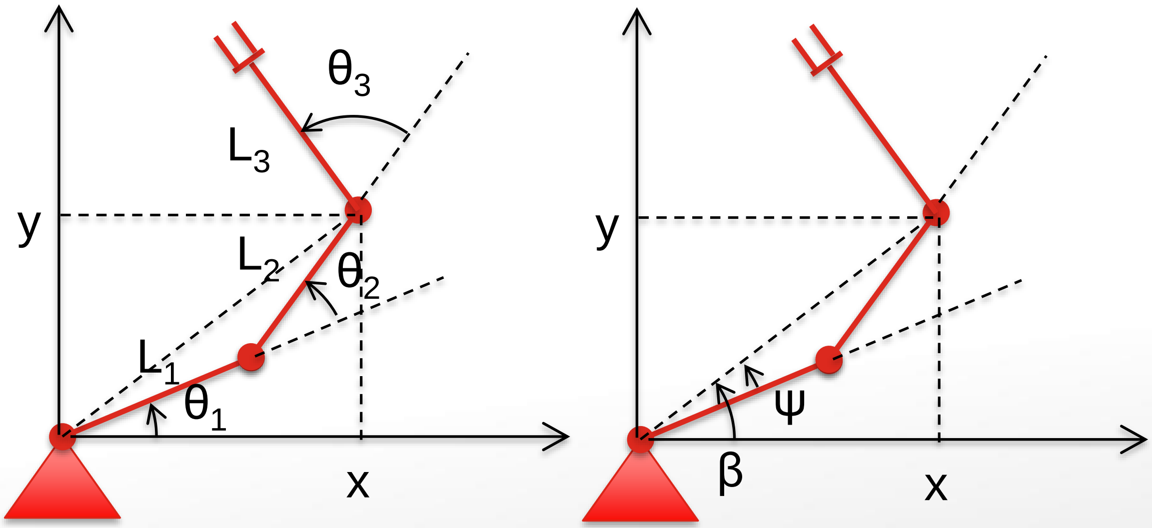
Note:
θ_1 and θ_2 known

Content

- Forward Kinematics
  - Introduction
  - Denavit-Hartenberg Parameters
  - Introduction to DH Parameters, Link Frame Attachment, Forward Kinematics through Example 1
  - More Examples
- Inverse Kinematics
  - Introduction
  - Algebraic Approach
  - Geometric Approach

Geometric Solutions

- Sometimes (for planar robot), the inverse kinematic problem can be solved easier using geometric approach.
- Again, this is done on a case-by-case basis.
- Same example:



- Using cosine rule: $$\begin{aligned} x^2 + y^2 &= L_1^2 + L_2^2 - 2L_1L_2\cos(180 - \theta_2) \\ &= L_1^2 + L_2^2 + 2L_1L_2\cos\theta_2 \end{aligned}$$ $$c_2 = \frac{x^2 + y^2 - L_1^2 - L_2^2}{2L_1L_2}$$
- Using symmetry for “Elbow-up” case: $\theta_2^{'} = -\theta_2$

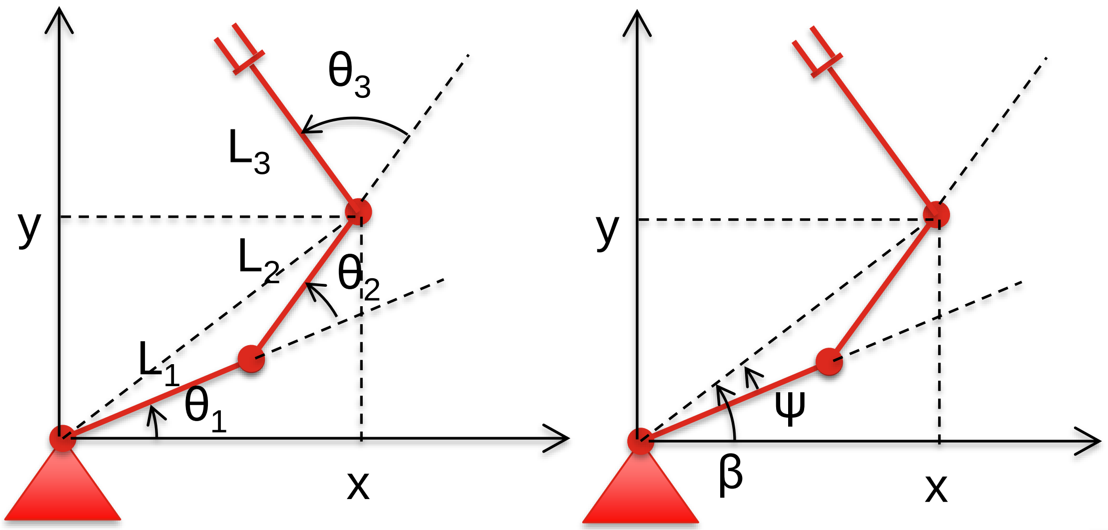
Geometric Solutions

- To solve for θ_1, note that:

$$\beta = \arctan 2(y, x)$$

$$\cos\psi = \frac{L_1^2 + x^2 + y^2 - L_2^2}{2L_1\sqrt{x^2 + y^2}}$$

(cosine rule)



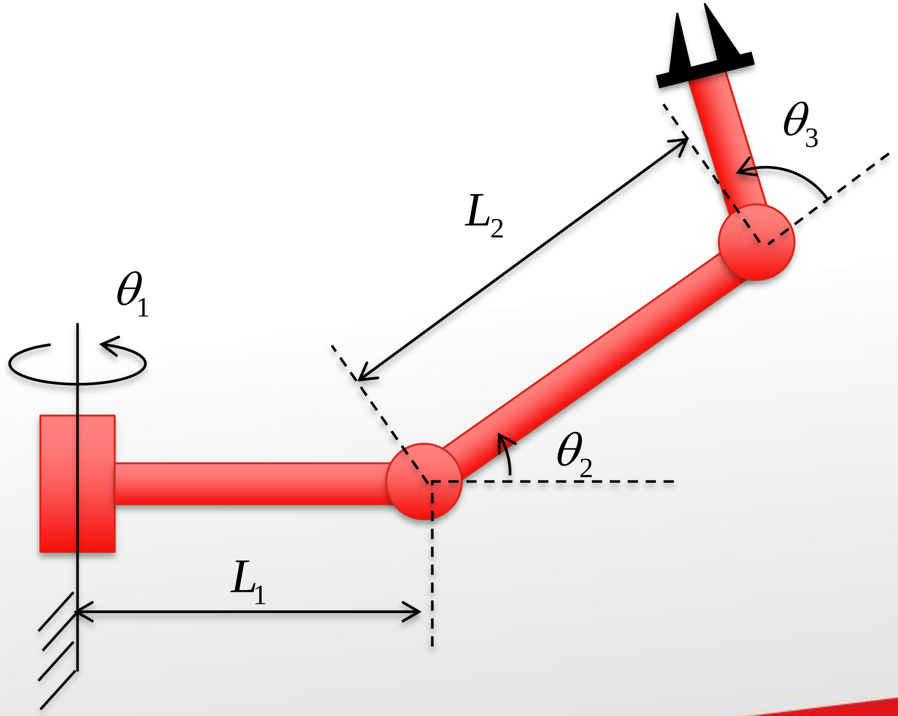
- The arc-cosine must be solved so that $0 \le \psi \le 180^\circ$
- Finally: $\theta_1 = \beta - \psi$ and using symmetry for "elbow up" case: $\theta_1 = \beta + \psi$
- θ_3 can be solved easily, because the sum of joint angles = final orientation.

$$\theta_1 + \theta_2 + \theta_3 = \phi \quad \Rightarrow \quad \theta_3 = \phi - \theta_1 - \theta_2$$

Tutorial Assignments

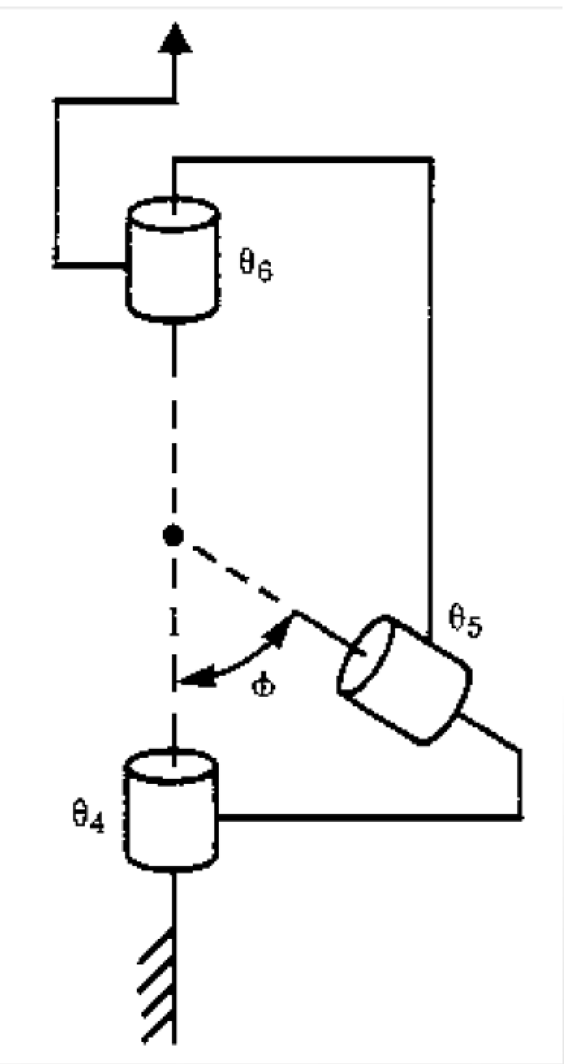
Tutorial Assignments

- **Question 1:**
  - For the robot shown on the right:
  - Derive the DH parameters,
  - Sketch the frames,
  - And calculate the Transform matrix.



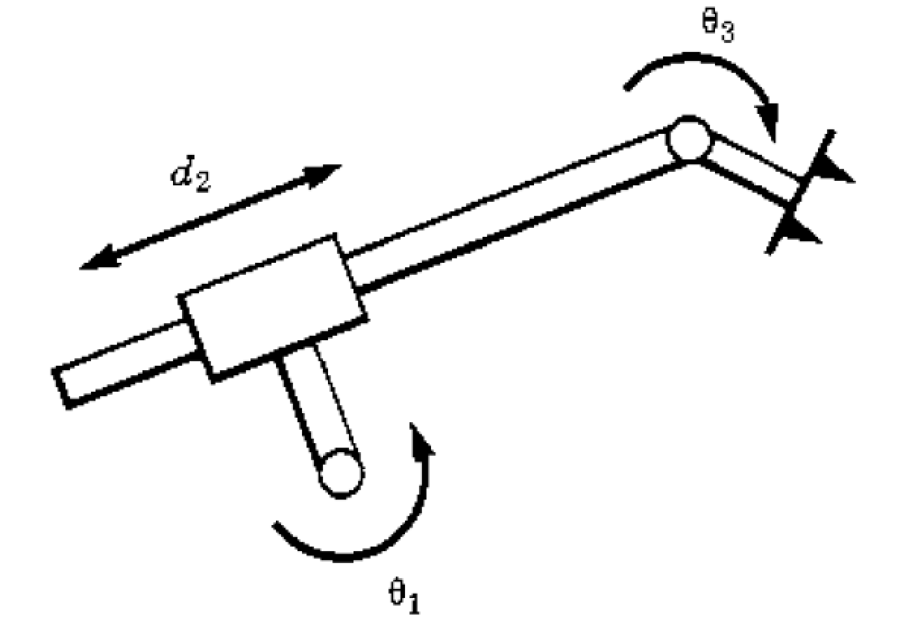
Tutorial Assignments

- **Question 2:**
  - For the robot shown on the right:
  - Derive the DH parameters (you may add in any constants i.e. link lengths or offset as required),
  - Sketch the frames,
  - And calculate the Transform matrix.



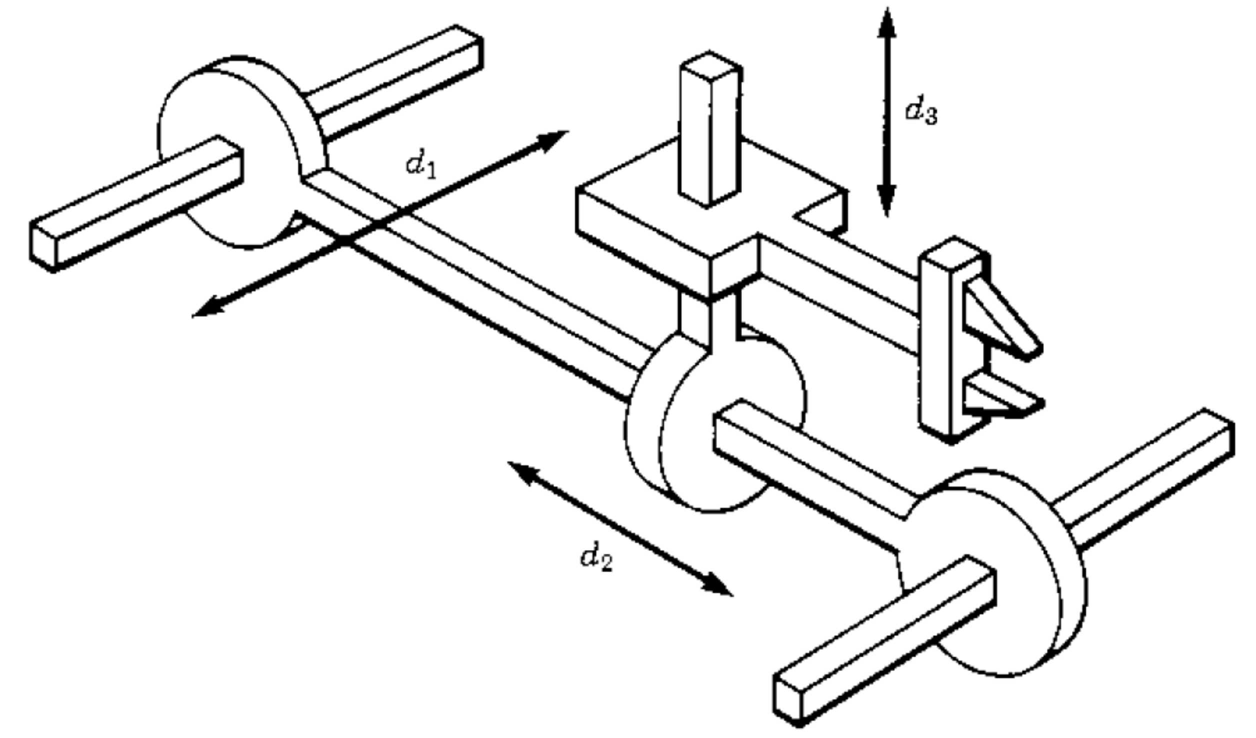
Tutorial Assignments

- **Question 3:**
  - For the robot shown below,
  - Derive the DH parameters (you may add in any constants i.e. link lengths or offset as required), sketch the frames, and calculate the Transform matrix.



Tutorial Assignments

- **Question 4:**
  - For the robot shown below,
  - Derive the DH parameters (you may add in any constants i.e. link lengths or offset as required), sketch the frames, and calculate the Transform matrix.



Thank you!

Have a good evening.

RMIT UNIVERSITY