MCT344/MCT342/CSE373/CSE471

Industrial Robotics

Lecture 4: Inverse Kinematics for Robotics

Ain Shams University
Faculty of Engineering

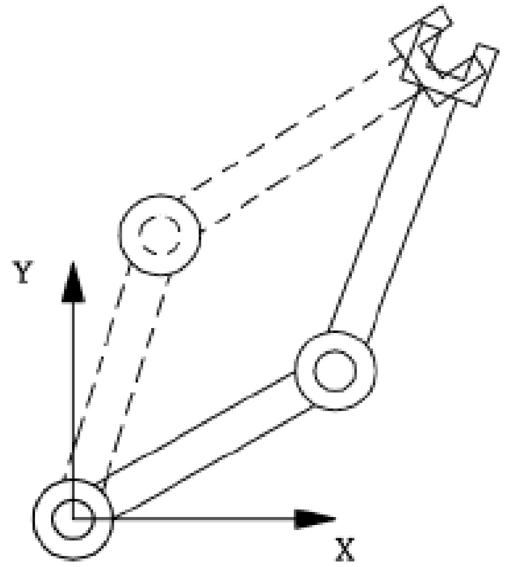
Presented By: Dr. Diaa Emad

Today's Agenda

- *Introduction.*
- *Kinematic Decoupling.*
- *Inverse Position (Geometric Approach).*
- *Inverse Orientation. (Algebraic Approach)*

Introduction

Inverse Kinematics



Given: Desired position and orientation of end-effector, $\mathbf{p}$.
Required: Joint Variables $\mathbf{q}$ (θ or d) to get $\mathbf{p}$

$$\mathbf{q} = f(\mathbf{p})$$

Given the desired 4×4 homogeneous transformation H

$$H = \begin{bmatrix} R_n^0 & o_n^0 \\ 0 & 1 \end{bmatrix}$$

The task is to find a solution (possibly one of many) of the equation:

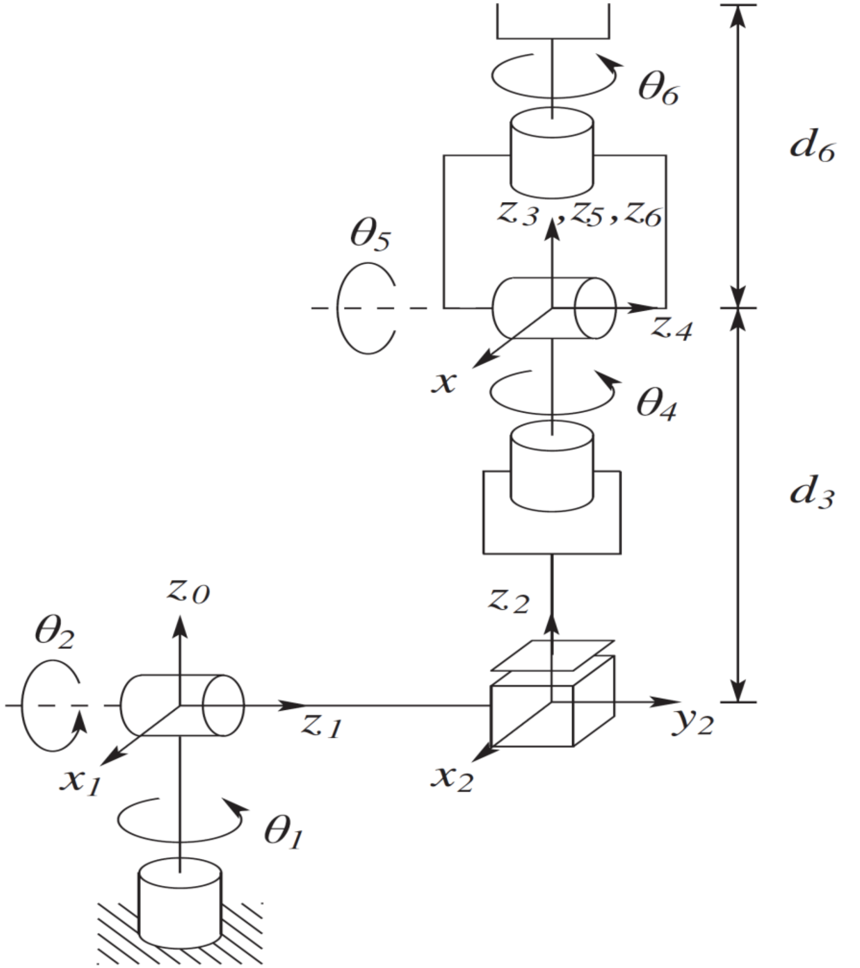
$$T_n^0(q_1, \ldots, q_n) = A_1(q_1)\ A_2(q_2) \ldots A_n(q_n) = H$$

It is 12 equations with respect to n variables $q_1, \ldots, q_n$

Introduction

Example 1

Recall the Stanford Manipulator. Suppose that the desired position and orientation of the final frame are given by:



$$T_6^0 = A_1 \cdots A_6 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H = \begin{bmatrix} 0 & 1 & 0 & -0.154 \\ 0 & 0 & 1 & 0.763 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Introduction

Example 1

To find the corresponding joint variables $\boldsymbol{\theta_1}$, $\boldsymbol{\theta_2}$, $\boldsymbol{d_3}$, $\boldsymbol{\theta_4}$, $\boldsymbol{\theta_5}$, and $\boldsymbol{\theta_6}$ we must solve the following simultaneous set of nonlinear trigonometric equations:

$$
\begin{aligned}
c_1[c_2(c_4c_5c_6 - s_4s_6) - s_2s_5c_6] - s_1(s_4c_5c_6 + c_4s_6) &= 0 \\
s_1[c_2(c_4c_5c_6 - s_4s_6) - s_2s_5c_6] + c_1(s_4c_5c_6 + c_4s_6) &= 0 \\
-s_2(c_4c_5c_6 - s_4s_6) - c_2s_5c_6 &= 1 \\
c_1[-c_2(c_4c_5s_6 + s_4c_6) + s_2s_5s_6] - s_1(-s_4c_5s_6 + c_4c_6) &= 1 \\
s_1[-c_2(c_4c_5s_6 + s_4c_6) + s_2s_5s_6] + c_1(-s_4c_5s_6 + c_4c_6) &= 0 \\
s_2(c_4c_5s_6 + s_4c_6) + c_2s_5s_6 &= 0 \\
c_1(c_2c_4s_5 + s_2c_5) - s_1s_4s_5 &= 0 \\
s_1(c_2c_4s_5 + s_2c_5) + c_1s_4s_5 &= 1 \\
-s_2c_4s_5 + c_2c_5 &= 0 \\
c_1s_2d_3 - s_1d_2 + d_6(c_1c_2c_4s_5 + c_1c_5s_2 - s_1s_4s_5) &= -0.154 \\
s_1s_2d_3 + c_1d_2 + d_6(c_1s_4s_5 + c_2c_4s_1s_5 + c_5s_1s_2) &= 0.763 \\
c_2d_3 + d_6(c_2c_5 - c_4s_2s_5) &= 0
\end{aligned}
$$

Introduction

Example 1

If the values of the nonzero DH parameters are $d_2 = 0.154$ and $d_6 = 0.263$, one solution to this set of equations is given by:

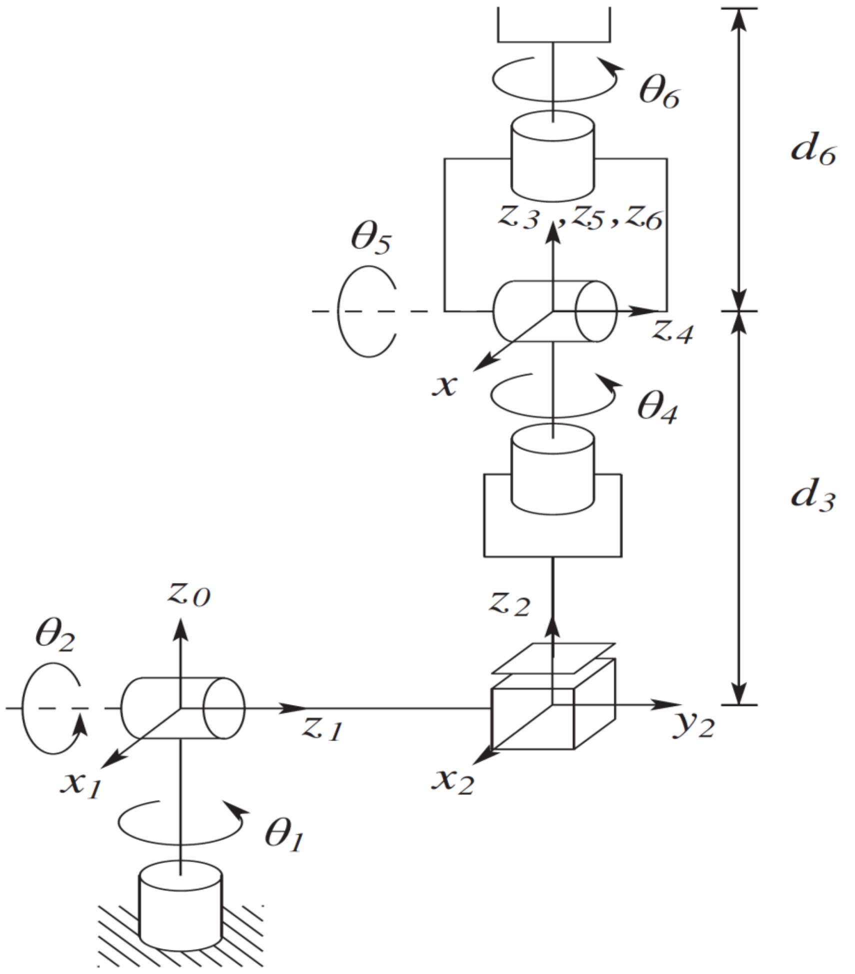
$$\theta_1 = \pi/2, \quad \theta_2 = \pi/2, \quad d_3 = 0.5, \quad \theta_4 = \pi/2, \quad \theta_5 = 0, \quad \theta_6 = \pi/2$$

- It satisfies the forward kinematics equations for the Stanford Arm.
- The equations in the example are much too difficult to solve directly in closed form. This is the case for most robot arms.
- Therefore, we need to develop efficient and systematic techniques that exploit the particular kinematic structure of the manipulator.
- Whereas the **forward kinematics problem** always has a **unique solution** that can be obtained simply by evaluating the forward equations, the inverse kinematics problem may or may not have a solution. Even if a solution exists, it may or may not be unique.
- The inverse kinematics problem may be solved **numerically** or in **closed form**. Finding a closed-form solution means finding an explicit relationship:

$$q_k = f_k(h_{11}, \ldots, h_{34}), \qquad k = 1, \ldots, n$$

Kinematic Decoupling

- For manipulators having six joints with three consecutive joint axes intersecting at a point (such as the **spherical wrist** in Stanford Manipulator), it is possible to decouple the inverse kinematics problem into two simpler problems, known respectively as **inverse position kinematics**, and **inverse orientation kinematics**.
- First finding the position of the intersection of the wrist axes, called the **wrist center,** and then finding the orientation of the wrist.
- let us suppose that there are exactly six degrees of freedom and that the last three joint axes intersect at a point o_c.
- We express two sets of equations representing the rotational and positional equations



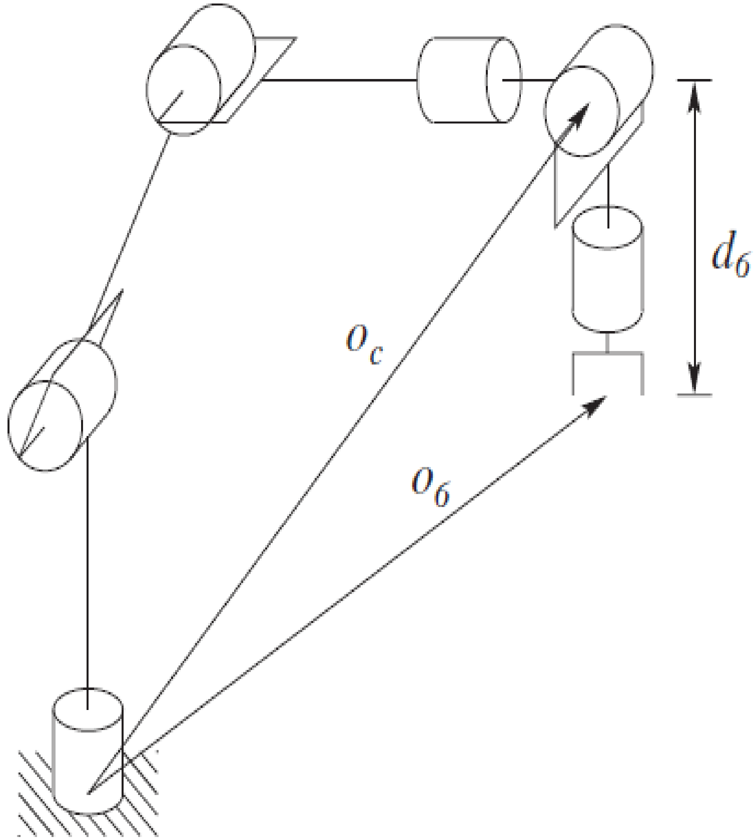
$$T_n^0(q_1, \ldots, q_n) = H \longrightarrow H = \begin{bmatrix} R_n^0 & o_n^0 \\ 0 & 1 \end{bmatrix} \longrightarrow \begin{aligned} R_6^0(q_1, \ldots, q_6) &= R \\ o_6^0(q_1, \ldots, q_6) &= o \end{aligned}$$

Kinematic Decoupling

- where $\boldsymbol{o}$ and $\boldsymbol{R}$ are the desired position and orientation of the tool frame, expressed with respect to the world coordinate system.
- Thus, we are given $\boldsymbol{o}$ and $\boldsymbol{R}$, and the inverse kinematics problem is to solve for $q_1, \ldots, q_6$.
- The motion of the final three joints about these axes will not change the position of o_c, and thus the position of the wrist center is a function of only the first three joint variables.
- The origin of the tool frame (whose desired coordinates are given by o) is simply obtained by a translation of distance d_6 along z_5 from o_c.
- In our case, z_5 and z_6 are the same axis, and the third column of R expresses the direction of z_6 with respect to the base frame.
- Therefore, we have

$$o = o_c^0 + d_6 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \longrightarrow o_c^0 = o - d_6 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$R_6^0(q_1, \ldots, q_6) = R$$
$$o_6^0(q_1, \ldots, q_6) = o$$



Kinematic Decoupling

- Thus, to have the end effector of the robot at the point with coordinates given by o and with the orientation of the end effector given by $R = (r_{ij})$, it is necessary and sufficient that the wrist center o_c have coordinates given by

$$o_c^0 = o - d_6 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

and that the orientation of the frame $o_6x_6y_6z_6$ with respect to the base be given by R.

- If the components of the end effector's position o are denoted o_x, o_y, o_z and the components of the wrist center o_c^0 are denoted x_c , y_c , z_c then:

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} o_x - d_6 r_{13} \\ o_y - d_6 r_{23} \\ o_z - d_6 r_{33} \end{bmatrix}$$

- We may find the values of the first three joint variables.
- This determines the orientation transformation R_3^0 which depends only on these first three joint variables.

Kinematic Decoupling

- We can determine the orientation of the end effector relative to the frame $o_3x_3y_3z_3$ from the expression

$$R = R_3^0 R_6^3 \longrightarrow R_6^3 = (R_3^0)^{-1} R = (R_3^0)^T R$$

- Note that the righthand side of above equation is completely known since $\boldsymbol{R}$ is given and $\boldsymbol{R_3^0}$ can be calculated once the first three joint variables are known.

Inverse Position: A Geometric Approach

- For the most common kinematic arrangements that we consider, we can use a geometric approach to find the variables q_1, q_2, q_3 corresponding to o_c^0.
- Since most six-DOF manipulator designs are kinematically simple, usually consisting of one of the five basic configurations with a spherical wrist, the geometric approach is simple and effective.
- The complexity of the inverse kinematics problem increases with the number of nonzero DH parameters. For most manipulators, many of the a_i, d_i are zero, the α_i are 0 or $\pi/2$, etc.
- The general idea of the geometric approach is to solve for joint variable q_i by projecting the manipulator onto the $x_{i-1} - y_{i-1}$ plane and solving a simple trigonometry problem.
- For example, to solve for θ_1, we project the arm onto the $x_0 - y_0$ plane and use trigonometry to find θ_1.

Inverse Position: A Geometric Approach

Spherical Configuration (RRP)

$$Atan2(x, y) = \tan^{-1}\left(\frac{y}{x}\right)$$

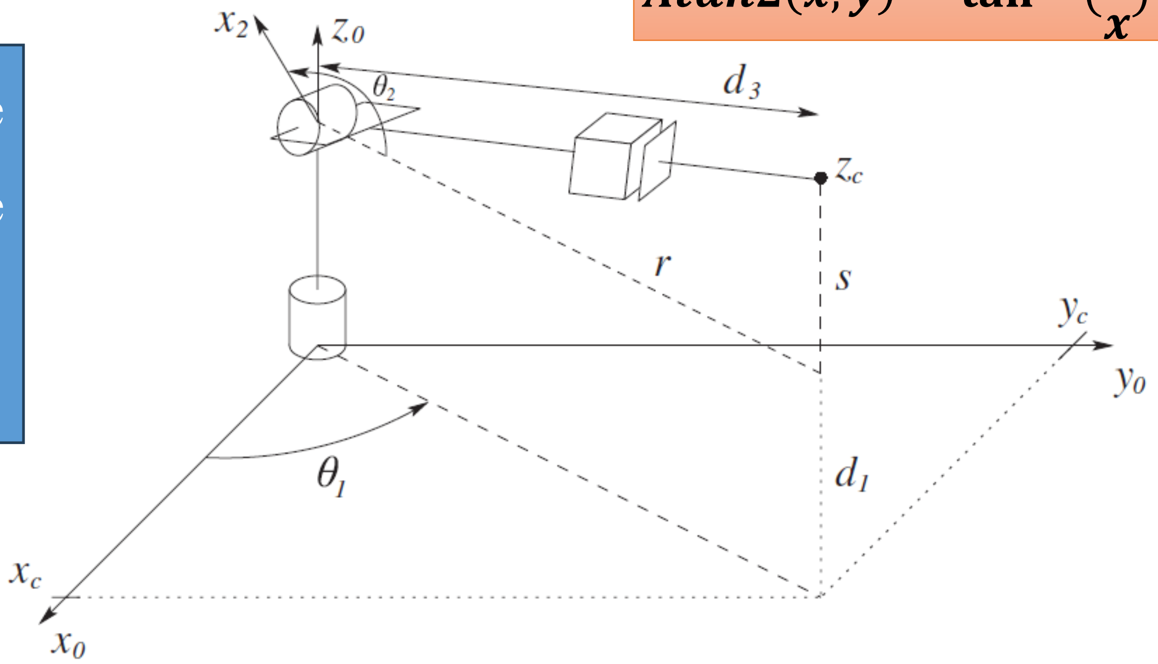
- We first solve the inverse position kinematics for a three degree of freedom spherical manipulator with the components of $o_c = o_c^0$ denoted by x_c, y_c, z_c.
- Projecting o_c onto the $x_0 - y_0$ plane, we see that

$$\theta_1 = \text{Atan2}(x_c, y_c)$$

$$\theta_2 = \text{Atan2}(r, s) + \frac{\pi}{2}$$

where $r^2 = x_c^2 + y_c^2$ and $s = z_c - d_1$.

$$d_3 = \sqrt{r^2 + s^2} = \sqrt{x_c^2 + y_c^2 + (z_c - d_1)^2}$$

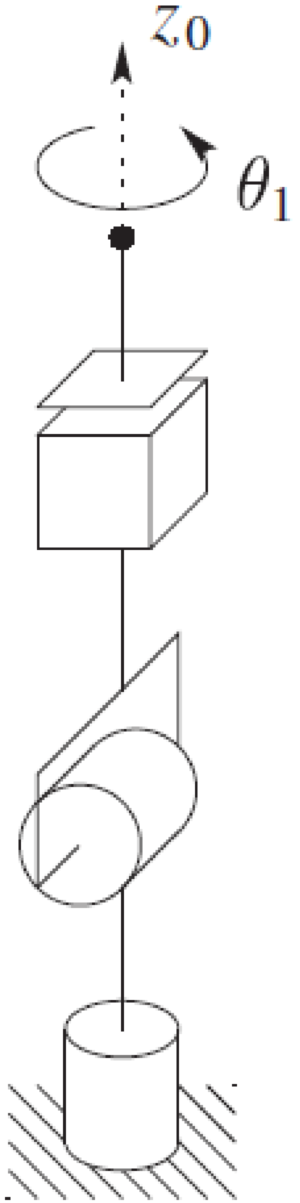


A second valid solution for θ_1 is → $\theta_1 = \pi + \text{Atan2}(x_c, y_c)$ → This will lead to a different solution for θ_2.

Inverse Position: A Geometric Approach

Spherical Configuration (RRP)

- These solutions for θ_1, are valid unless $x_c = y_c = 0$.
- In this case, Equation of θ_1 is undefined and the manipulator is in a singular configuration, in which the wrist center o_c intersects z_0 as shown in Figure.
- In this configuration any value of θ_1 leaves o_c fixed.
- There are thus infinitely many solutions for θ_1 when o_c intersects z_0.



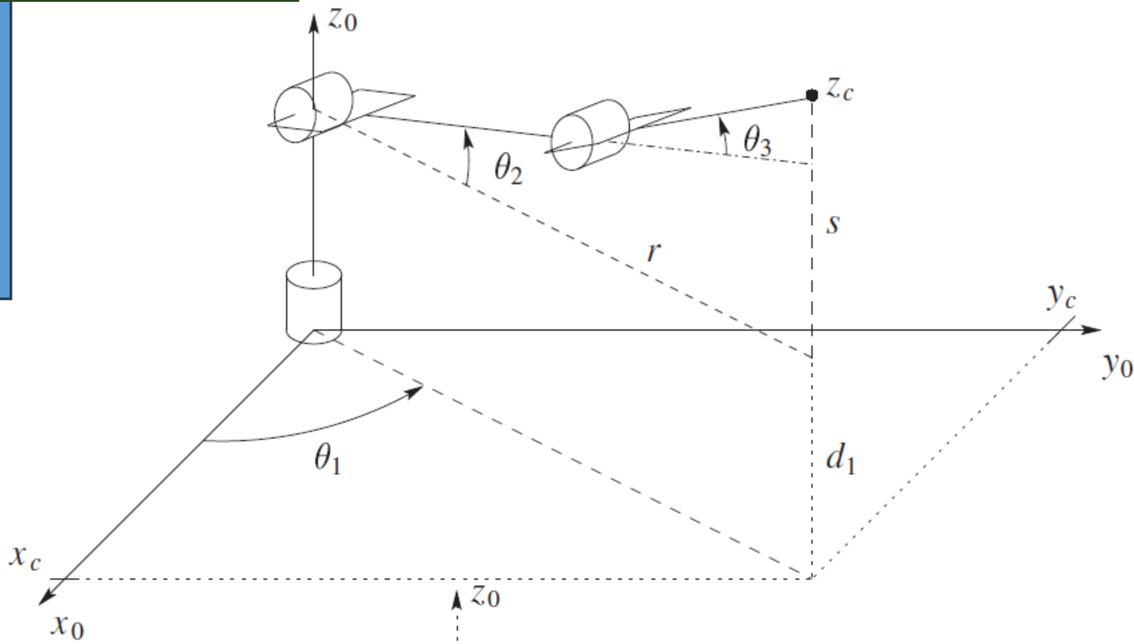
- The negative square root solution for d_3 is disregarded and thus in this case we obtain two solutions to the inverse position kinematics as long as the wrist center does not intersect z_0.

Inverse Position: A Geometric Approach

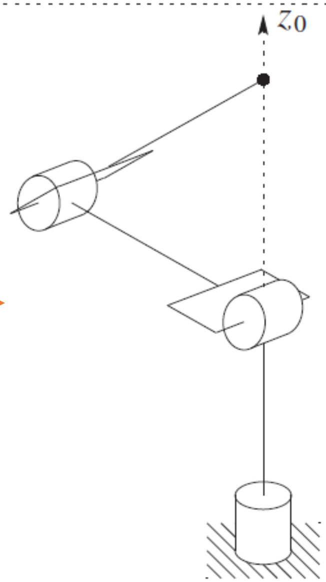
Articulated Configuration (Elbow Manipulator)(RRR)

As in the case of the spherical manipulator, the first joint variable is the base rotation and there are two possible solutions provided x_c and y_c are not both zero.

$$\theta_1 = \mathrm{Atan2}(x_c, y_c)$$
$$\theta_1 = \pi + \mathrm{Atan2}(x_c, y_c)$$



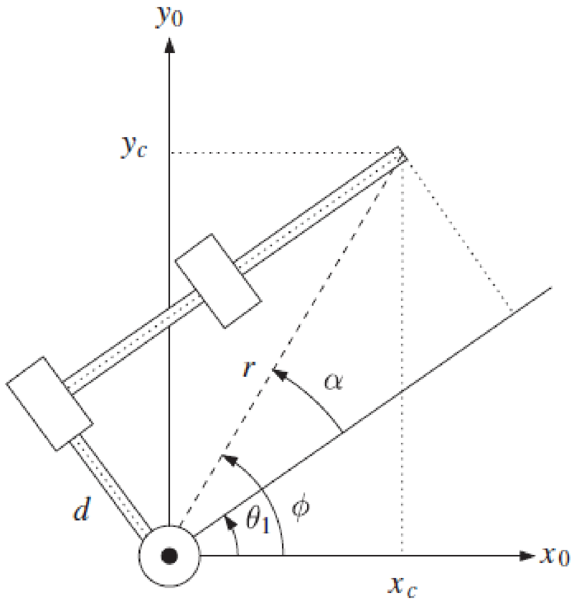
If both x_c and y_c are zero, the configuration is singular as before and θ_1 may take on any value.

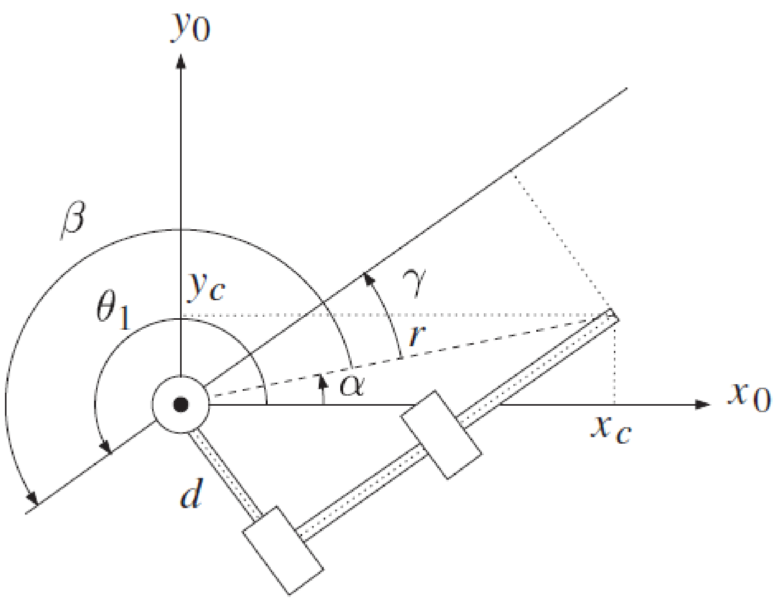


Inverse Position: A Geometric Approach

Articulated Configuration (Elbow Manipulator)(RRR)

- If there is an offset $d \neq 0$ then the wrist center cannot intersect z_0.
- In this case, depending on how the DH parameters have assigned, we will have $d_2 = d$ or $d_3 = d$, and there will, in general, be only two solutions for θ_1 .
- These correspond to the so-called left arm and right arm configurations.





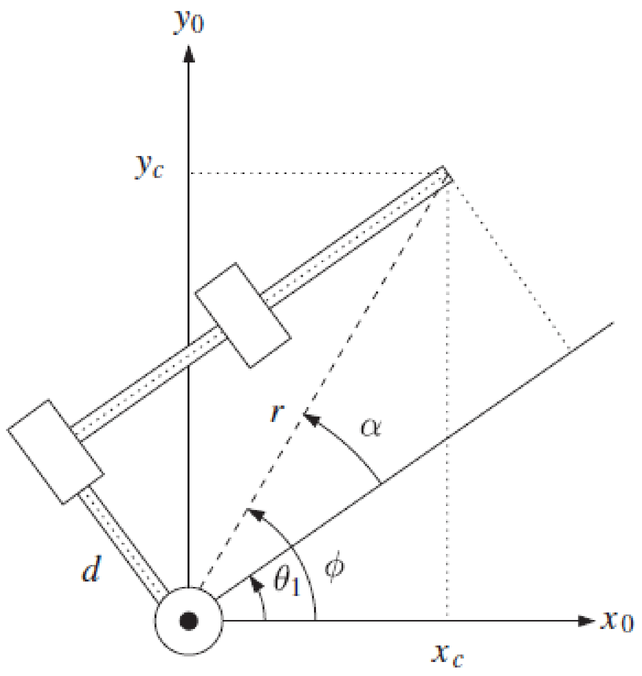
Inverse Position: A Geometric Approach

Articulated Configuration (Elbow Manipulator)(RRR)

For the left arm configuration

$$\theta_1 = \phi - \alpha$$

$$\begin{aligned} \phi &= \text{Atan2}(x_c, y_c) \\ \alpha &= \text{Atan2}\left(\sqrt{r^2 - d^2}, d\right) \\ &= \text{Atan2}\left(\sqrt{x_c^2 + y_c^2 - d^2}, d\right) \end{aligned}$$

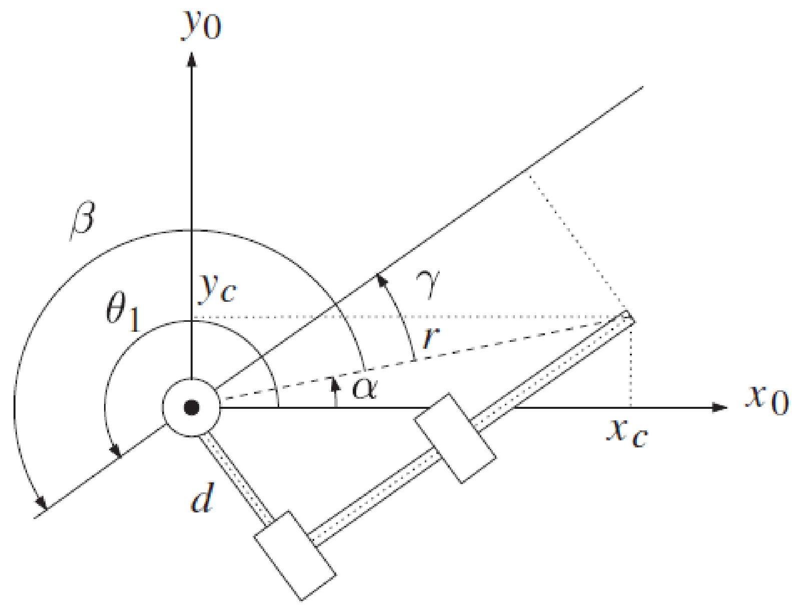


The second solution, given by the right arm configuration

$$\theta_1 = \text{Atan2}(x_c, y_c) + \text{Atan2}\left(-\sqrt{r^2 - d^2}, -d\right)$$

$$\begin{aligned} \theta_1 &= \alpha + \beta \\ \alpha &= \text{Atan2}(x_c, y_c) \\ \beta &= \gamma + \pi \\ \gamma &= \text{Atan2}\left(\sqrt{r^2 - d^2}, d\right) \end{aligned}$$

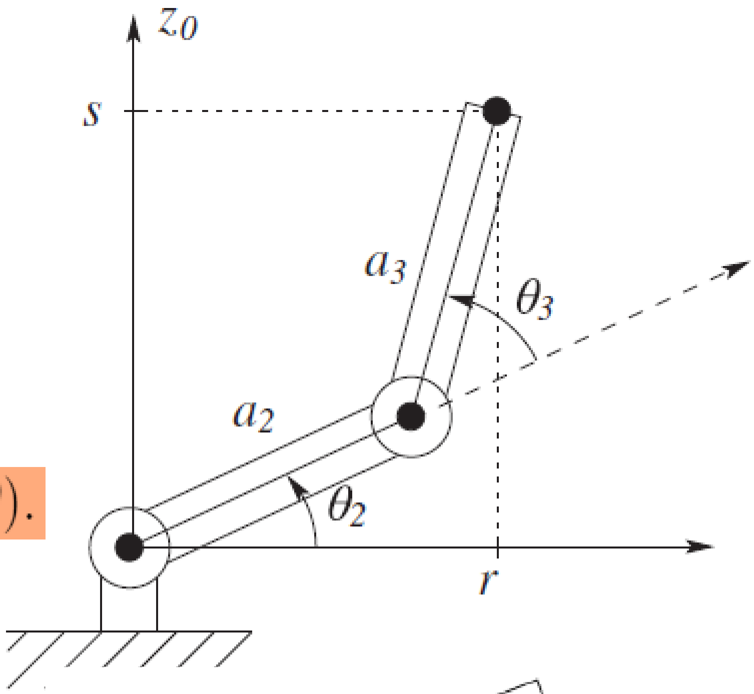
$$\longrightarrow \beta = \text{Atan2}\left(-\sqrt{r^2 - d^2}, -d\right)$$



Inverse Position: A Geometric Approach

Articulated Configuration (Elbow Manipulator)(RRR)

- To find the angles θ_2 ; θ_3 for the elbow manipulator given θ_1 , we consider the plane formed by the second and third links.
- Since the motion of second and third links is planar, the solution is analogous to that of the two-link manipulator.



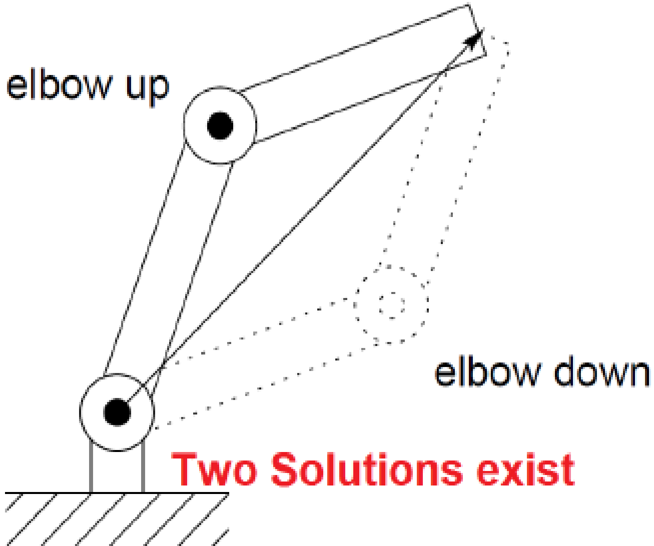
Apply the law of cosines

$\cos(\theta+\pi) = -\cos(\theta)$ and $\sin(\theta+\pi) = -\sin(\theta)$.

$$\cos\theta_3 = \frac{r^2+s^2-a_2^2-a_3^2}{2a_2a_3} = \frac{x_c^2+y_c^2-d^2+(z_c-d_1)^2-a_2^2-a_3^2}{2a_2a_3} := D$$

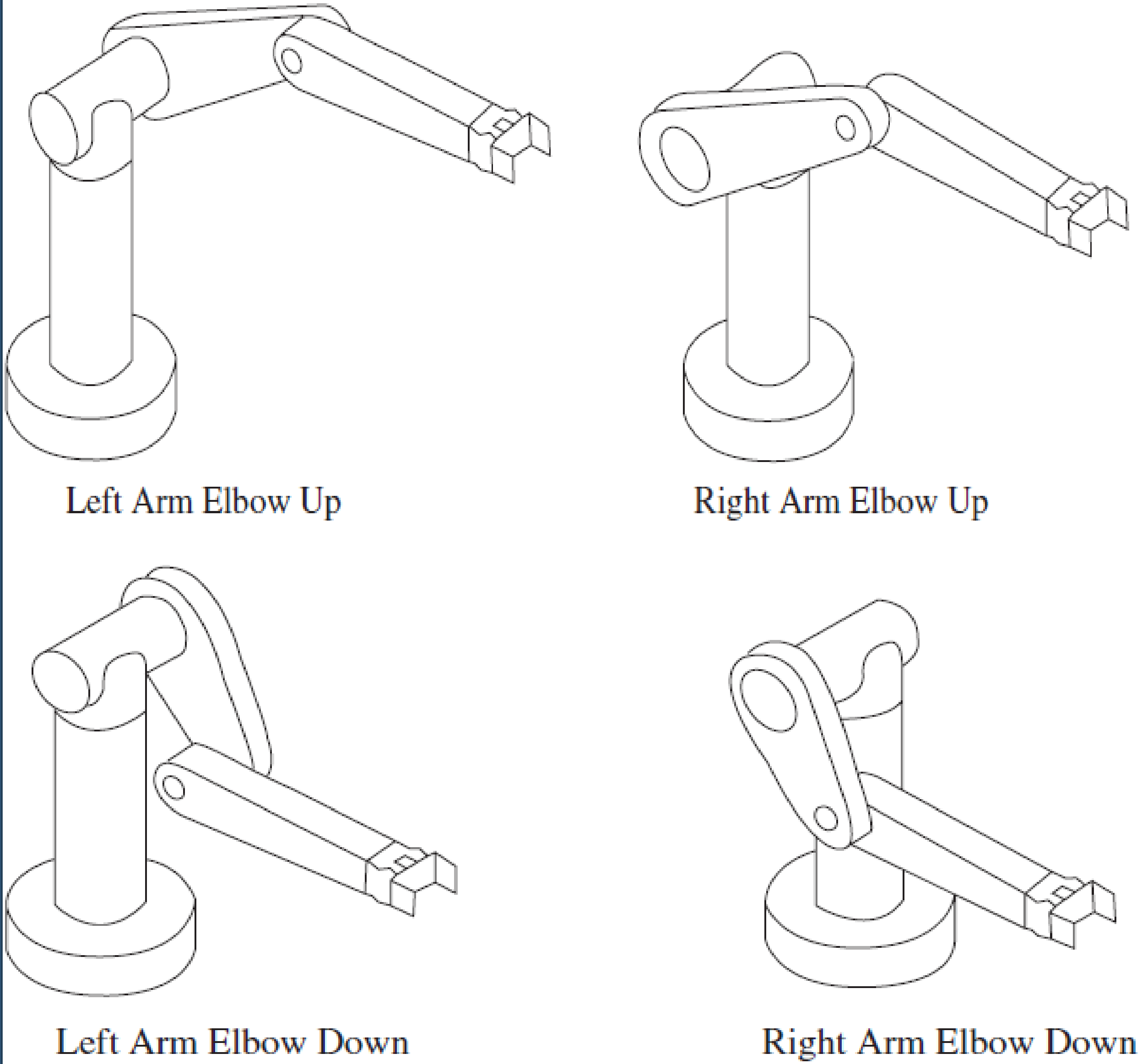
$$\theta_3 = \text{Atan2}\left(D, \pm\sqrt{1-D^2}\right)$$

$$\theta_2 = \text{Atan2}(r,s) - \text{Atan2}(a_2+a_3c_3, a_3s_3) = \text{Atan2}\left(\sqrt{x_c^2+y_c^2-d^2}, z_c-d_1\right) - \text{Atan2}(a_2+a_3c_3, a_3s_3)$$



Inverse Position: A Geometric Approach

- An example of an elbow manipulator with offsets is the PUMA shown in Figure.
- There are four solutions to the inverse position kinematics as shown.
- These correspond to the situations left arm-elbow up, left arm-elbow down, right-arm elbow up and right arm-elbow down.
- We will see that there are two solutions for the wrist orientation thus giving a total of eight solutions of the inverse kinematics for the PUMA manipulator.



Inverse Orientation: An Algebraic Approach

- We used a geometric approach to solve the inverse position problem.
- This gives the values of the first three joint variables corresponding to a given position of the wrist center.
- The inverse orientation problem is now one of finding the values of the final three joint variables corresponding to a given orientation with respect to the frame $o_3x_3y_3z_3$.
- For a spherical wrist, this can be interpreted as the problem of finding a set of Euler angles corresponding to a given rotation matrix $\boldsymbol{R}$.
- In particular, we solve for the three Euler angles, φ, θ, ψ , and then use the mapping

$$\theta_4 = \phi, \theta_5 = \theta, \theta_6 = \psi$$

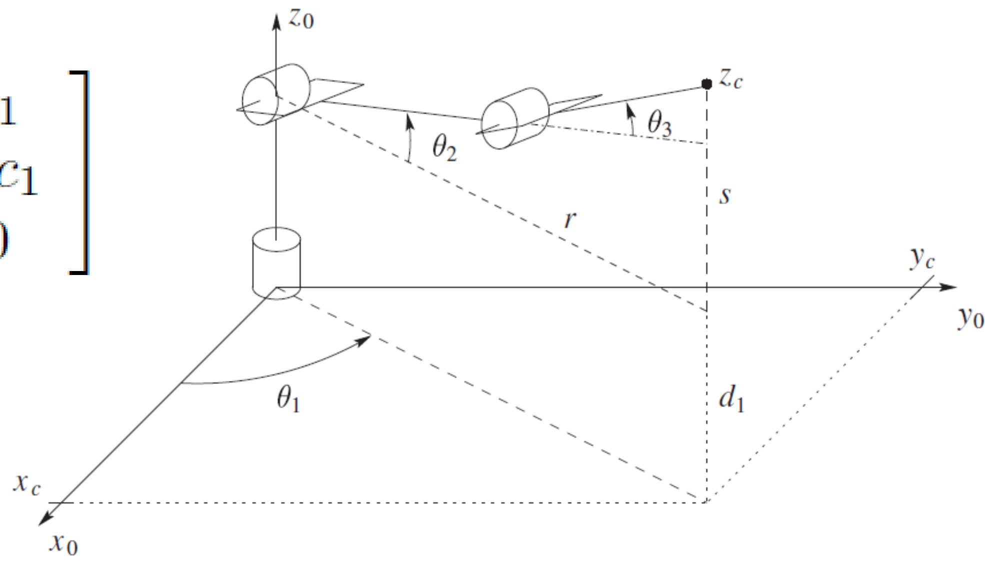
$$\begin{aligned} R_{ZYZ} &= R_{z,\phi}R_{y,\theta}R_{z,\psi} \\ &= \begin{bmatrix} c_\phi & -s_\phi & 0 \\ s_\phi & c_\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_\theta & 0 & s_\theta \\ 0 & 1 & 0 \\ -s_\theta & 0 & c_\theta \end{bmatrix} \begin{bmatrix} c_\psi & -s_\psi & 0 \\ s_\psi & c_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} c_\phi c_\theta c_\psi - s_\phi s_\psi & -c_\phi c_\theta s_\psi - s_\phi c_\psi & c_\phi s_\theta \\ s_\phi c_\theta c_\psi + c_\phi s_\psi & -s_\phi c_\theta s_\psi + c_\phi c_\psi & s_\phi s_\theta \\ -s_\theta c_\psi & s_\theta s_\psi & c_\theta \end{bmatrix} \end{aligned}$$

Inverse Orientation: An Algebraic Approach

Articulated Manipulator with Spherical Wrist

Link	a_i	α_i	d_i	θ_i
1	0	90	d_1	θ_1
2	a_2	0	0	θ_2
3	a_3	0	0	θ_3

$$R_3^0 = \begin{bmatrix} c_1c_{23} & -c_1s_{23} & s_1 \\ s_1c_{23} & -s_1s_{23} & -c_1 \\ s_{23} & c_{23} & 0 \end{bmatrix}$$



$$R_6^3 = \begin{bmatrix} c_4c_5c_6 - s_4s_6 & -c_4c_5s_6 - s_4c_6 & c_4s_5 \\ s_4c_5c_6 + c_4s_6 & -s_4c_5s_6 + c_4c_6 & s_4s_5 \\ -s_5c_6 & s_5s_6 & c_5 \end{bmatrix}$$

$$R_6^3 = (R_3^0)^T R$$

The three equations given by the third column are given by

There are other solutions

$$c_4s_5 = c_1c_{23}r_{13} + s_1c_{23}r_{23} + s_{23}r_{33}$$
$$s_4s_5 = -c_1s_{23}r_{13} - s_1s_{23}r_{23} + c_{23}r_{33}$$
$$c_5 = s_1r_{13} - c_1r_{23}$$

$$\theta_5 = \text{Atan2}\left(s_1r_{13} - c_1r_{23}, \pm\sqrt{1 - (s_1r_{13} - c_1r_{23})^2}\right)$$
$$\theta_4 = \text{Atan2}(c_1c_{23}r_{13} + s_1c_{23}r_{23} + s_{23}r_{33}, -c_1s_{23}r_{13} - s_1s_{23}r_{23} + c_{23}r_{33})$$
$$\theta_6 = \text{Atan2}(-s_1r_{11} + c_1r_{21}, s_1r_{12} - c_1r_{22})$$

Inverse Kinematics

Elbow Manipulator / Complete Solution

Given

$$o = \begin{bmatrix} o_x \\ o_y \\ o_z \end{bmatrix}, \quad R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \longrightarrow \begin{aligned} x_c &= o_x - d_6 r_{13} \\ y_c &= o_y - d_6 r_{23} \\ z_c &= o_z - d_6 r_{33} \end{aligned}$$

$$\theta_1 = \text{Atan2}(x_c, y_c)$$

$$\theta_2 = \text{Atan2}\left(\sqrt{x_c^2 + y_c^2 - d^2}, z_c - d_1\right) - \text{Atan2}(a_2 + a_3 c_3, a_3 s_3)$$

$$\theta_3 = \text{Atan2}\left(D, \pm\sqrt{1 - D^2}\right),$$

$$\textit{with } D = \frac{x_c^2 + y_c^2 - d^2 + (z_c - d_1)^2 - a_2^2 - a_3^2}{2a_2 a_3}$$

$$\theta_4 = \text{Atan2}(c_1 c_{23} r_{13} + s_1 c_{23} r_{23} + s_{23} r_{33}, -c_1 s_{23} r_{13} - s_1 s_{23} r_{23} + c_{23} r_{33})$$

$$\theta_5 = \text{Atan2}\left(s_1 r_{13} - c_1 r_{23}, \pm\sqrt{1 - (s_1 r_{13} - c_1 r_{23})^2}\right)$$

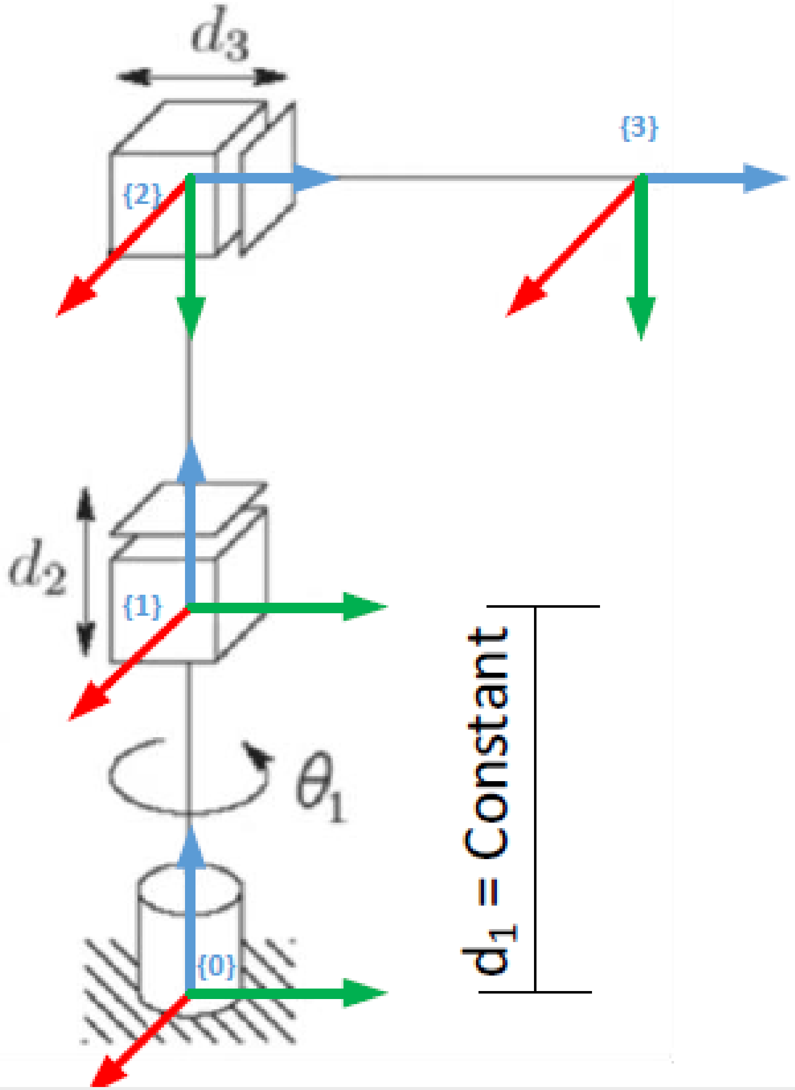
$$\theta_6 = \text{Atan2}(-s_1 r_{11} + c_1 r_{21}, s_1 r_{12} - c_1 r_{22})$$

There are possible other solutions

Additional Examples on Algebraic Approach

Example: RPP Robot

$$T_3^0 = A_1 A_2 A_3 \quad = \begin{bmatrix} c_1 & 0 & -s_1 & -s_1 d_3 \\ s_1 & 0 & c_1 & c_1 d_3 \\ 0 & -1 & 0 & d_1 + d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Forward Kinematics:

$$x = -d_3 sin\theta_1$$

$$y = d_3 cos\theta_1$$

$$z = d_1 + d_2$$

Additional Examples on Algebraic Approach

Example: RPP Robot

$$x = -d_3 sin\theta_1$$

$$y = d_3 cos\theta_1$$

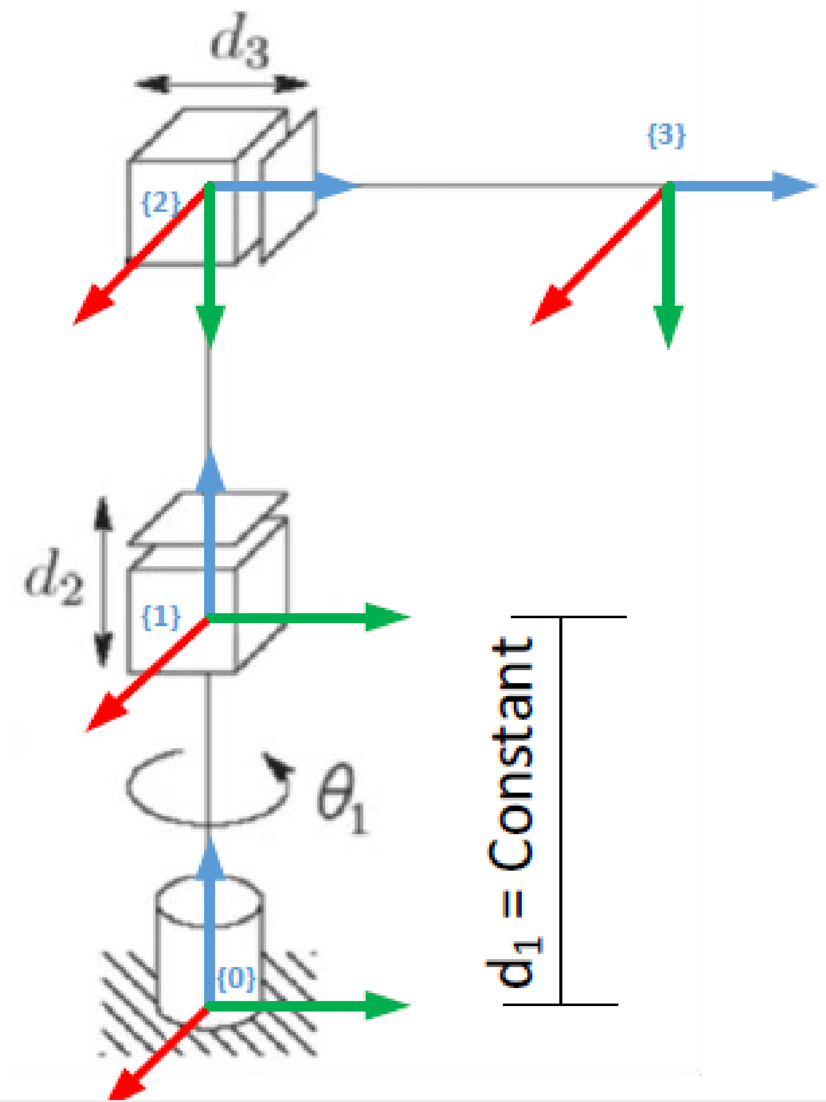
$$z = d_1 + d_2$$

Inverse Kinematics:

$$\theta_1 = tan^{-1}\frac{-x}{y}$$

$$d_2 = z - d_1$$

$$d_3 = \sqrt{x^2 + y^2}$$



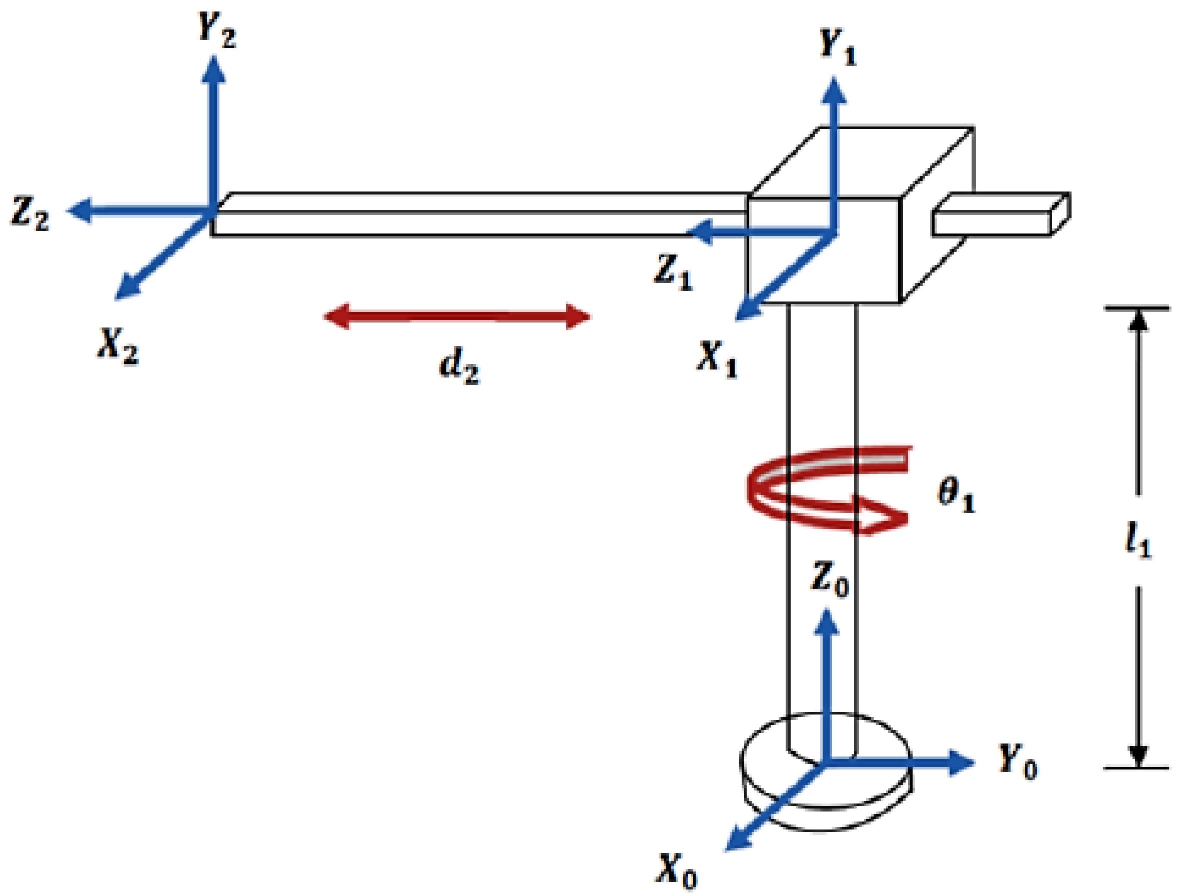
Additional Examples on Algebraic Approach

Example: RP Robot

$$A_2^0 = A_1^0 * A_2^1$$

$$A_1^0 = \begin{bmatrix} C\theta_1 & 0 & S\theta_1 & 0 \\ S\theta_1 & 0 & -C\theta_1 & 0 \\ 0 & 1 & 0 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad A_2^1 = \begin{bmatrix} 1 & 0 & S\theta_1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2^0 = \begin{bmatrix} C\theta_1 & 0 & S\theta_1 & d_2 S\theta_1 \\ S\theta_1 & 0 & -C\theta_1 & -d_2 C\theta_1 \\ 0 & 1 & 0 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



$$x = d_2 sin\theta_1$$

$$y = -d_2 cos\theta_1$$

$$z = l_1$$

Additional Examples on Algebraic Approach

Example: RP Robot

$$x = d_2 sin\theta_1$$

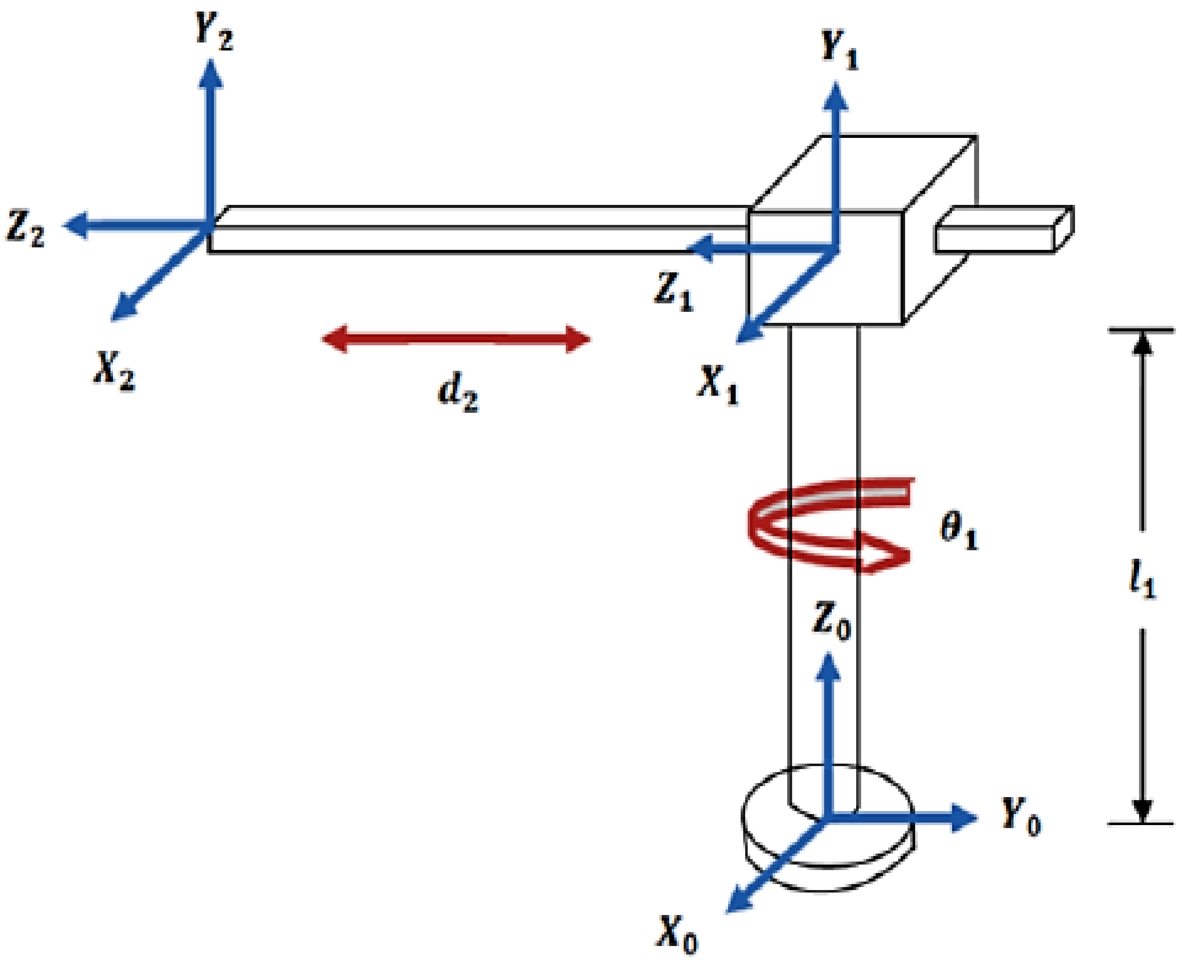
$$y = -d_2 cos\theta_1$$

$$z = l_1$$

Inverse Kinematics:

$$\theta_1 = tan^{-1}\frac{-x}{y}$$

$$d_2 = \sqrt{x^2 + y^2}$$



MATLAB: Robotics System Toolbox

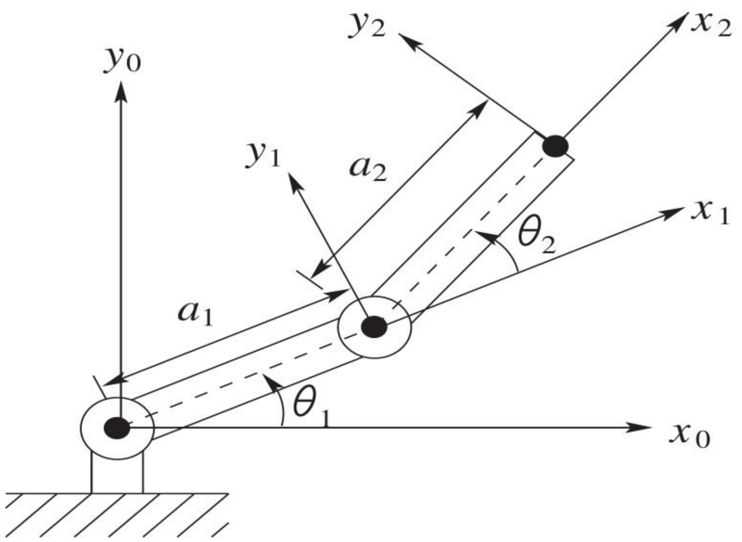
Inverse Kinematics

```
ik = inverseKinematics('RigidBodyTree',robot);
weights = [0.25 0.25 0.25 1 1 1];
initialguess = robot.homeConfiguration;
[configSoln,solnInfo] = ik('endeffector',T,weights,initialguess);
theta1=rad2deg(configSoln(1).JointPosition)
theta2=rad2deg(configSoln(2).JointPosition)
```

Result

theta1 = 60.0000

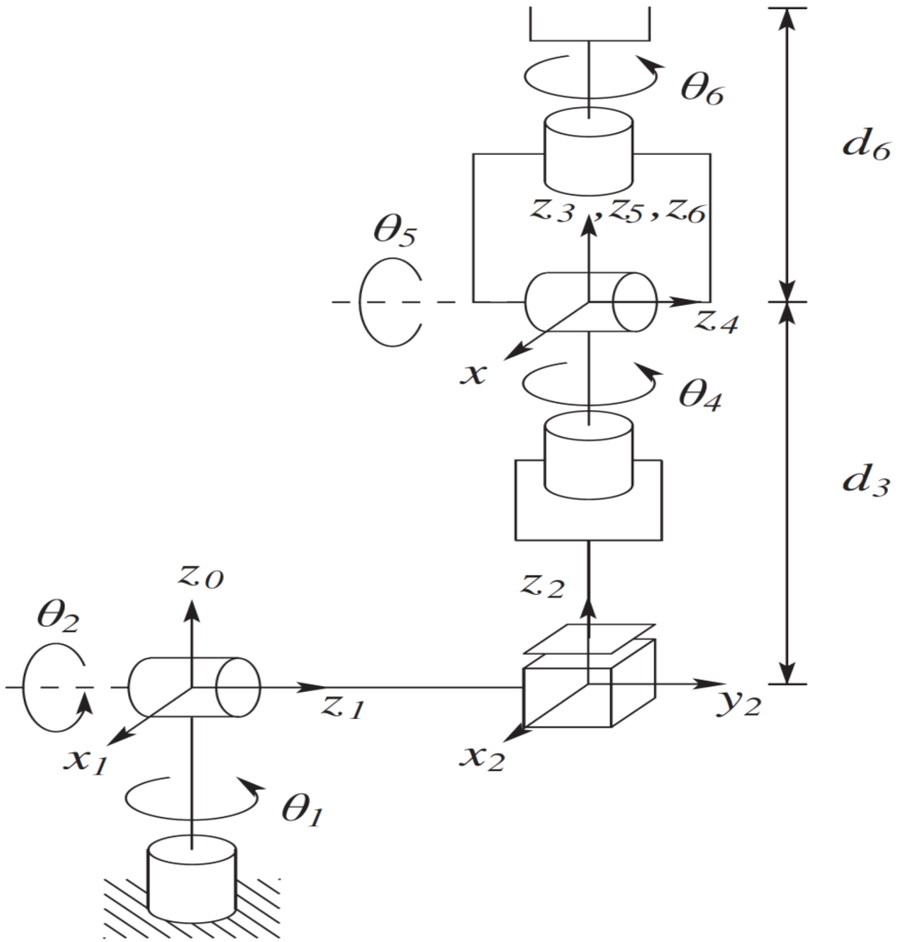
theta2 = 30.0000



MATLAB: Robotics System Toolbox

Inverse Kinematics

Example: Stanford Manipulator

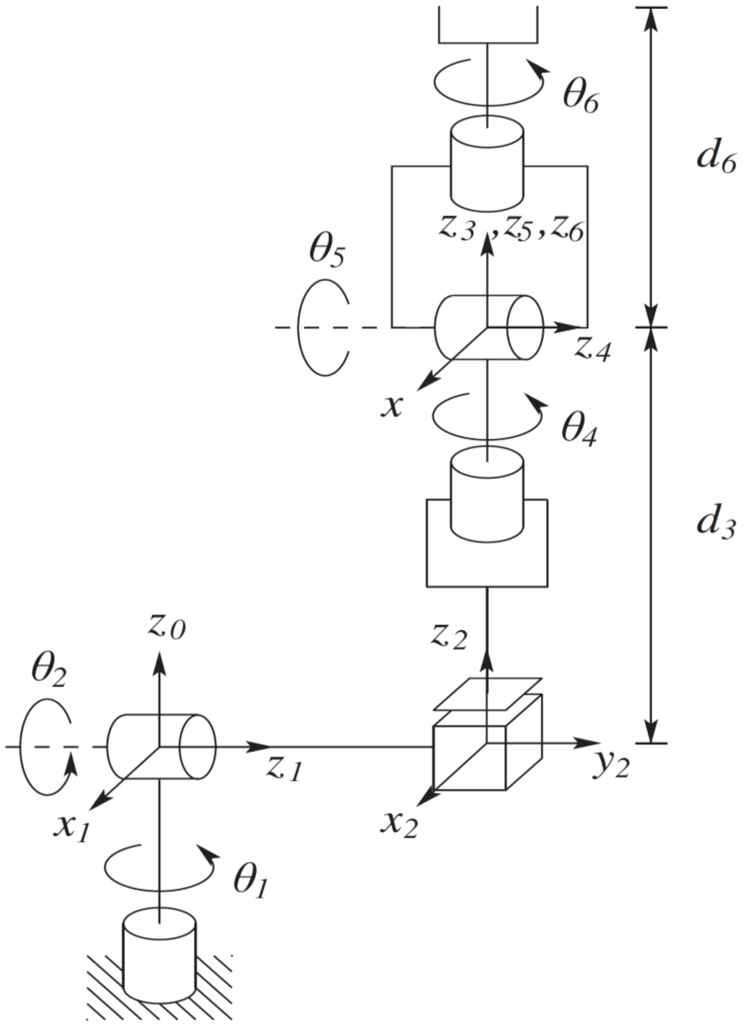


Link	d_i	a_i	α_i	θ_i
1	0	0	-90	θ_1
2	d_2	0	$+90$	θ_2
3	d_3	0	0	0
4	0	0	-90	θ_4
5	0	0	$+90$	θ_5
6	d_6	0	0	θ_6

MATLAB: Robotics System Toolbox

Example: Stanford Manipulator

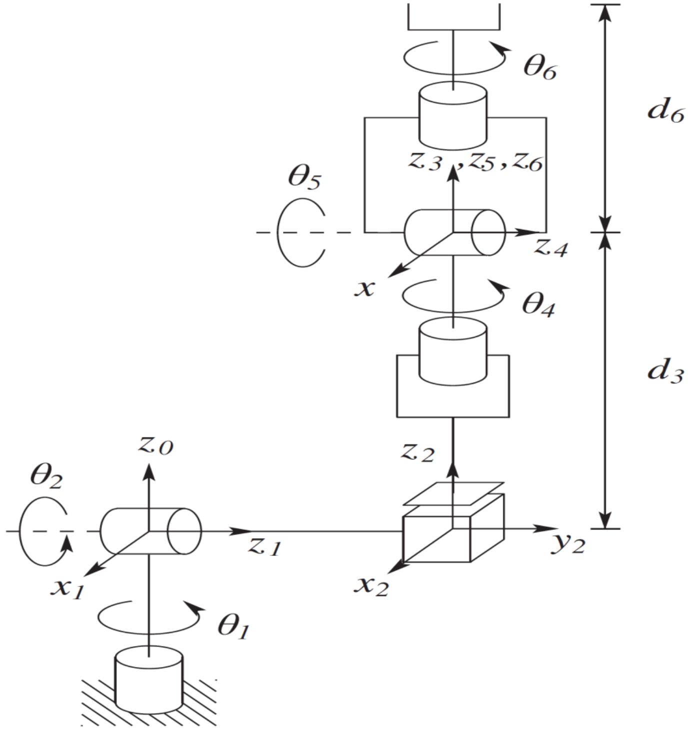
```
robot = rigidBodyTree;
body1 = rigidBody('body1');
body2 = rigidBody('body2');
body3 = rigidBody('body3');
body4 = rigidBody('body4');
body5 = rigidBody('body5');
body6 = rigidBody('body6');
bodyEndEffector = rigidBody('endeffector');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
jnt1 = rigidBodyJoint('jnt1','revolute');
jnt2 = rigidBodyJoint('jnt2','revolute');
jnt3 = rigidBodyJoint('jnt3','prismatic');
jnt4 = rigidBodyJoint('jnt4','revolute');
jnt5 = rigidBodyJoint('jnt5','revolute');
jnt6 = rigidBodyJoint('jnt6','revolute');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```



MATLAB: Robotics System Toolbox

Example: Stanford Manipulator

```
jnt1.HomePosition = deg2rad(0);
jnt2.HomePosition = deg2rad(0);
jnt3.HomePosition = 0.5;
jnt4.HomePosition = deg2rad(0);
jnt5.HomePosition = deg2rad(0);
jnt6.HomePosition = deg2rad(0);
%%%%%%%%%%%%%%%[a alpha d theta]
dh1=[0 deg2rad(-90) 0 0];
dh2=[0 deg2rad(90) 1 0];
dh3=[0 0 0 0];
dh4=[0 deg2rad(-90) 0 0];
dh5=[0 deg2rad(90) 0 0];
dh6=[0 0 1 0];
%%%%%%%%%%%%%%%%%%%%%%%%%%
```

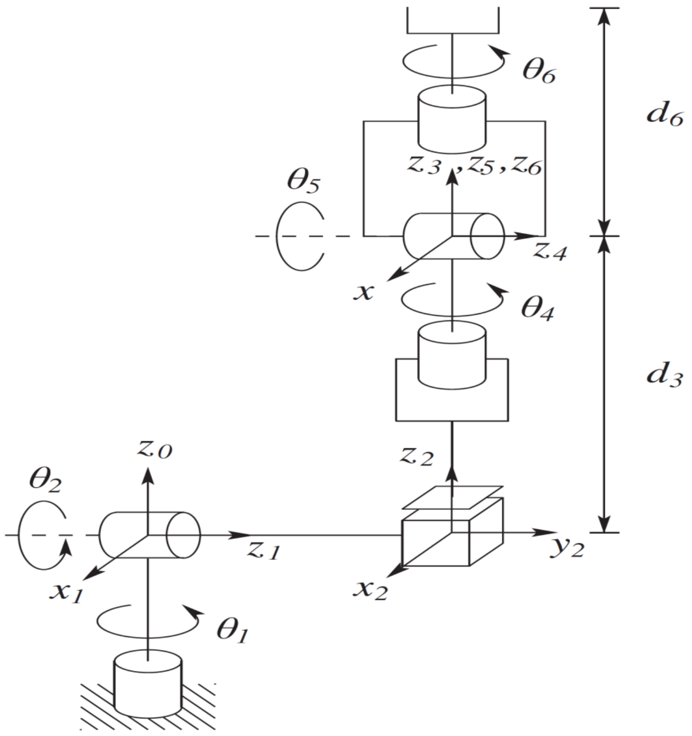


Link	d_i	a_i	α_i	θ_i
1	0	0	-90	θ_1
2	d_2	0	$+90$	θ_2
3	d_3	0	0	0
4	0	0	-90	θ_4
5	0	0	$+90$	θ_5
6	d_6	0	0	θ_6

MATLAB: Robotics System Toolbox

Example: Stanford Manipulator

```
setFixedTransform(jnt1,dh1,'dh');
setFixedTransform(jnt2,dh2,'dh');
setFixedTransform(jnt3,dh3,'dh');
setFixedTransform(jnt4,dh4,'dh');
setFixedTransform(jnt5,dh5,'dh');
setFixedTransform(jnt6,dh6,'dh');
%%%%%%%%%%%%%%%%%%%%%%%
body1.Joint = jnt1;
body2.Joint = jnt2;
body3.Joint = jnt3;
body4.Joint = jnt4;
body5.Joint = jnt5;
body6.Joint = jnt6;
%%%%%%%%%%%%%%%%%%%%%%%
```

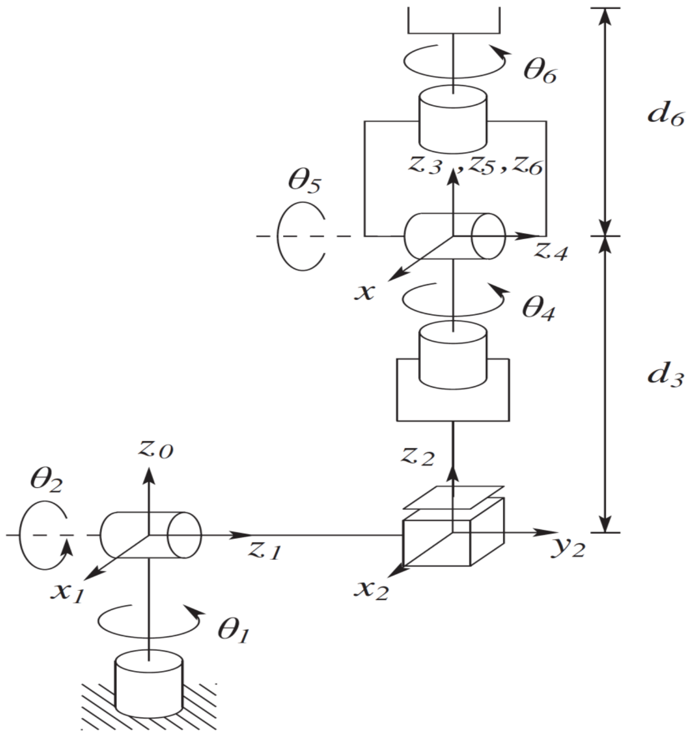


Link	d_i	a_i	α_i	θ_i
1	0	0	-90	θ_1
2	d_2	0	$+90$	θ_2
3	d_3	0	0	0
4	0	0	-90	θ_4
5	0	0	$+90$	θ_5
6	d_6	0	0	θ_6

MATLAB: Robotics System Toolbox

Example: Stanford Manipulator

```
addBody(robot,body1,'base');
addBody(robot,body2,'body1');
addBody(robot,body3,'body2');
addBody(robot,body4,'body3');
addBody(robot,body5,'body4');
addBody(robot,body6,'body5');
addBody(robot,bodyEndEffector,'body6');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
config = homeConfiguration(robot)
config(1).JointPosition = deg2rad(60)
config(2).JointPosition = deg2rad(0)
config(3).JointPosition = 1
config(4).JointPosition = deg2rad(0)
config(5).JointPosition = deg2rad(60)
config(6).JointPosition = deg2rad(0)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

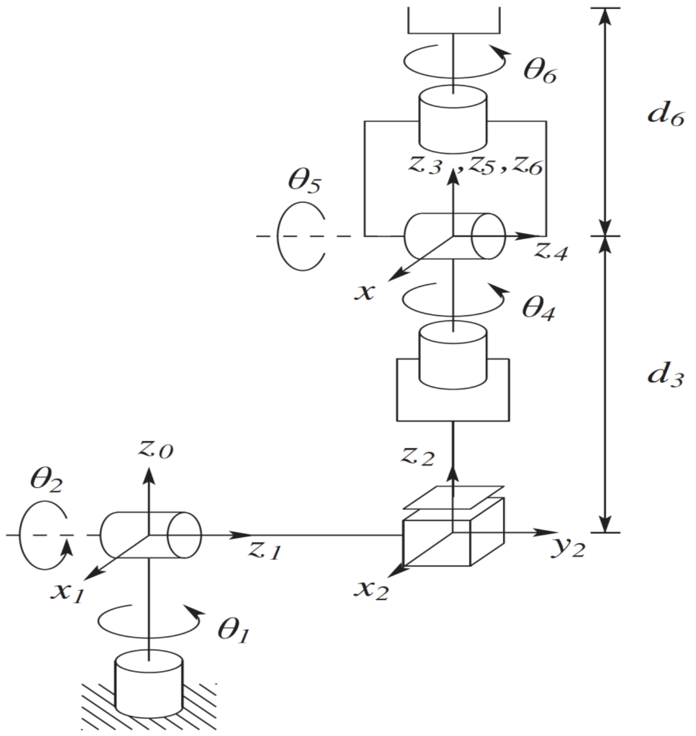


Link	d_i	a_i	α_i	θ_i
1	0	0	-90	θ_1
2	d_2	0	$+90$	θ_2
3	d_3	0	0	0
4	0	0	-90	θ_4
5	0	0	$+90$	θ_5
6	d_6	0	0	θ_6

MATLAB: Robotics System Toolbox

Example: Stanford Manipulator

```
T = getTransform(robot,config,'endeffector','base')
showdetails(robot)
show(robot)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
ik = inverseKinematics('RigidBodyTree',robot);
weights = [0.01 0.01 0.01 0.1 0.1 0.1];
initialguess = robot.homeConfiguration;
[configSoln,solnInfo] = ik('endeffector',T,weights,initialguess)
theta1=rad2deg(configSoln(1).JointPosition)
theta2=rad2deg(configSoln(2).JointPosition)
d3=(configSoln(3).JointPosition)
theta4=rad2deg(configSoln(4).JointPosition)
theta5=rad2deg(configSoln(5).JointPosition)
theta6=rad2deg(configSoln(6).JointPosition)
```



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

Thank You

Any Questions ??