University of Pisa
Master of Science in Computer Science
Course of Robotics (ROB)
A.Y. 2016/17

Robot mechanics and kinematics

Cecilia Laschi
The BioRobotics Institute
Scuola Superiore Sant'Anna, Pisa

cecilia.laschi@santannapisa.it
http://didawiki.cli.di.unipi.it/doku.php/magistraleinformatica/rob/start

Robot mechanics and kinematics

- Introduction to robot mechanics
    - Definition of degree of freedom (DOF)
    - Definition of robot manipulator
    - Joint types
    - Manipulator types
- Definitions of joint space and Cartesian space
    - Robot position in joint space
    - Robot position in Cartesian space
    - Definition of workspace
- Direct and inverse kinematics
    - Kinematics transformations
    - Concept of kinematic redundancy
    - Concept of kinematic singularity
    - Recall of transformation matrices
- Denavit-Hartenberg representation
    - Algorithm
    - Examples

References: Bajd, Mihelj, Lenarcic, Stanovnik, Munih, *Robotics*, Springer, 2010: Chapters 1-2

Degree of Freedom (DOF)

1 DOF

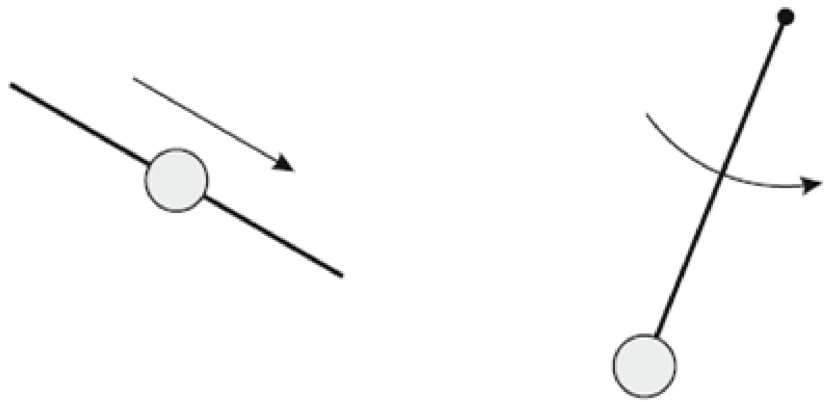
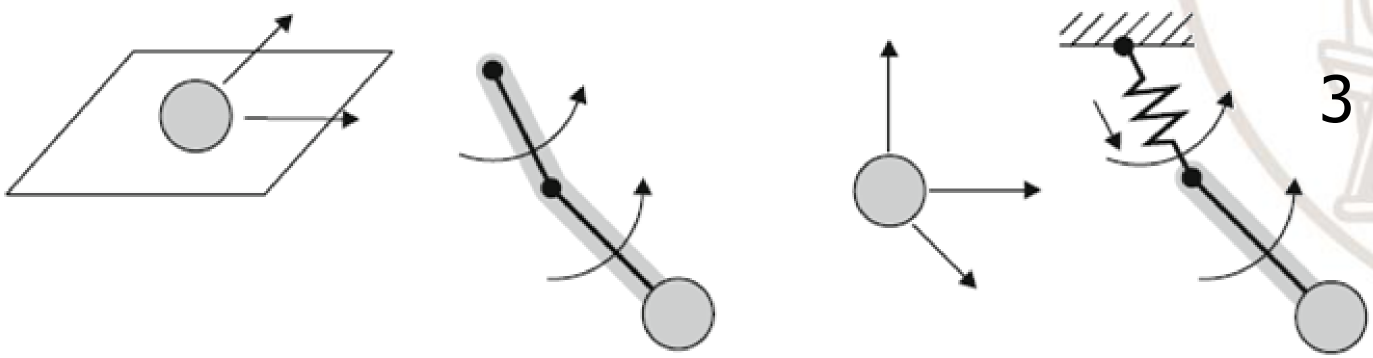


Fig. 1.1 Two examples of systems with one degree of freedom: mass particle on a wire (left) and rigid pendulum in a plane (right)

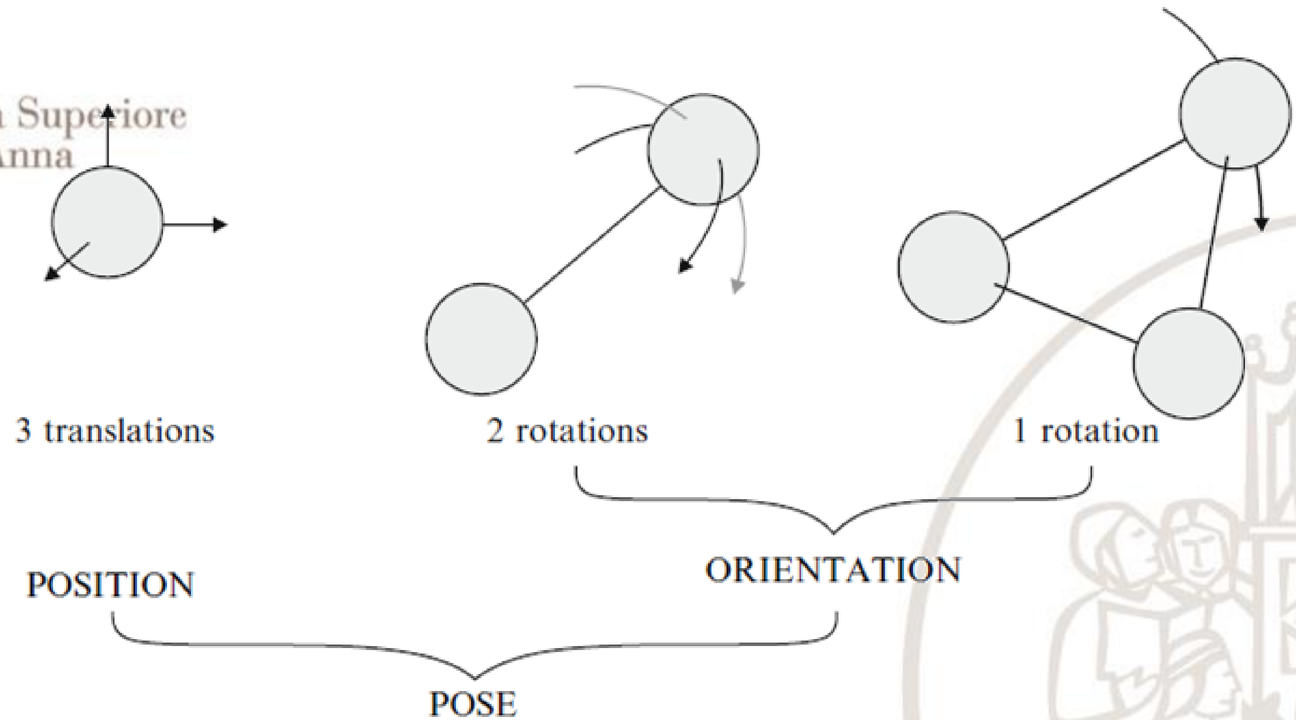
2 DOFs



3 DOFs

Fig. 1.2 Examples with two (left) and three degrees of freedom (right)

DOFs of a rigid body



A single mass particle has three degrees of freedom, described by three rectangular displacements along a line called translations (T).
We add another mass particle to the first one in such a way that there is constant distance between them. The second particle is restricted to move on the surface of a sphere surrounding the first particle.
Its position on the sphere can be described by two circles reminding us of meridians and latitudes on a globe.
The displacement along a circular line is called rotation (R).
The third mass particle is added in such a way that the distances with respect to the first two particles are kept constant. In this way the third particle may move along a circle, a kind of equator, around the axis determined by the first two particles.
A rigid body therefore has six degrees of freedom: three translations and three rotations. The first three degrees of freedom describe the position of the body, while the other three degrees of freedom determine its orientation.
The term pose is used to include both position and orientation.

Robot manipulator

- Definition: **open kinematic chain**
- Sequence of rigid segments, or **links**, connected through revolute or translational **joints**, actuated by a **motor**
- One extremity is connected to a support **base**, the other one is free and equipped with a tool, named **end effector**

Joints and DOFs

- **Joint** = set of two surfaces that can slide, keeping contact to one another

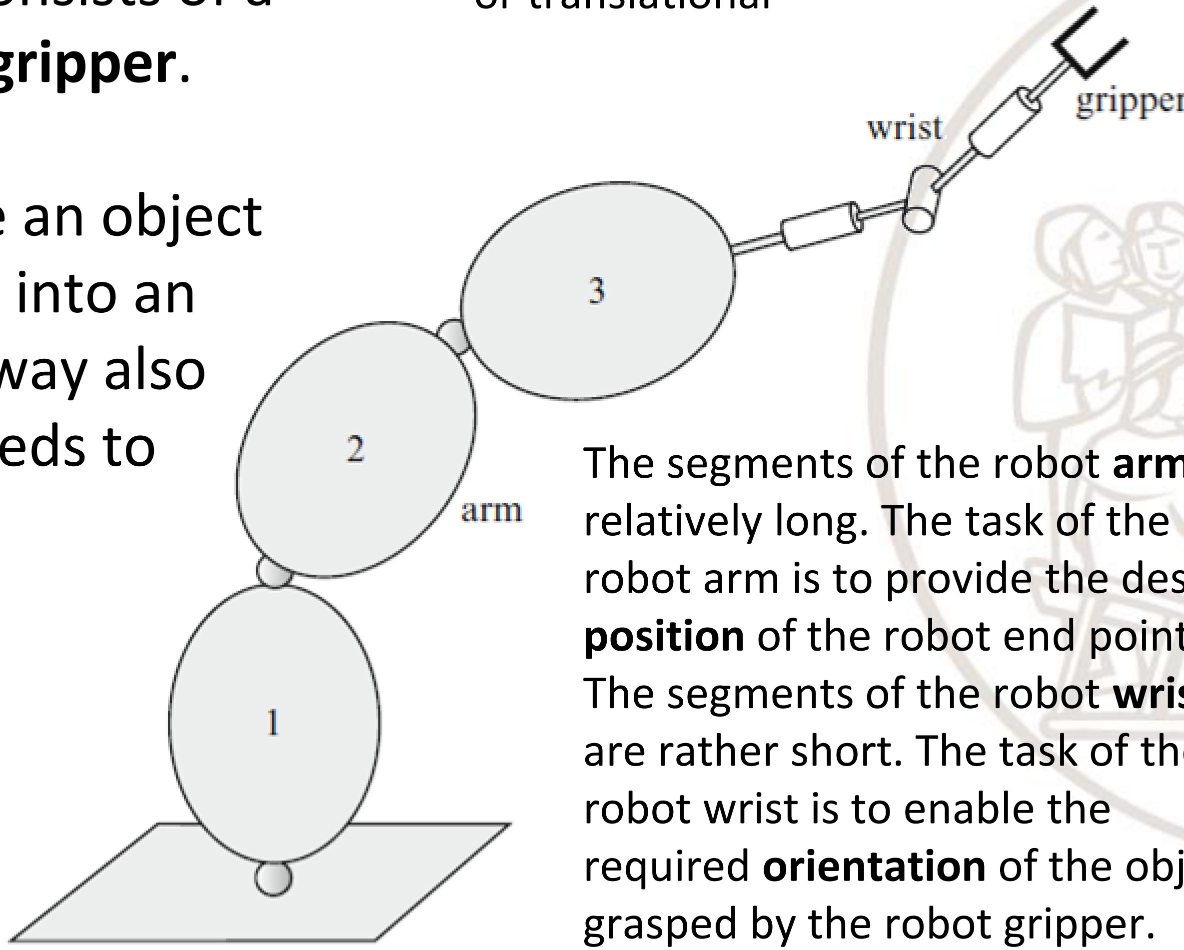
- Couple joint-link = robot **degree of freedom (DOF)**

- **Link 0** = support base and origin of the reference coordinate frame for robot motion

Robot manipulator

A robot manipulator consists of a robot **arm**, **wrist**, and **gripper**. The task of the robot manipulator is to place an object grasped by the gripper into an arbitrary **pose**. In this way also the industrial robot needs to have **six** degrees of freedom.

Chain of 3 links
2 adjacent links are connected by 1 joint
Each joint gives 1 DOF, either rotational or translational



The segments of the robot **arm** are relatively long. The task of the robot arm is to provide the desired **position** of the robot end point. The segments of the robot **wrist** are rather short. The task of the robot wrist is to enable the required **orientation** of the object grasped by the robot gripper.

Joint types

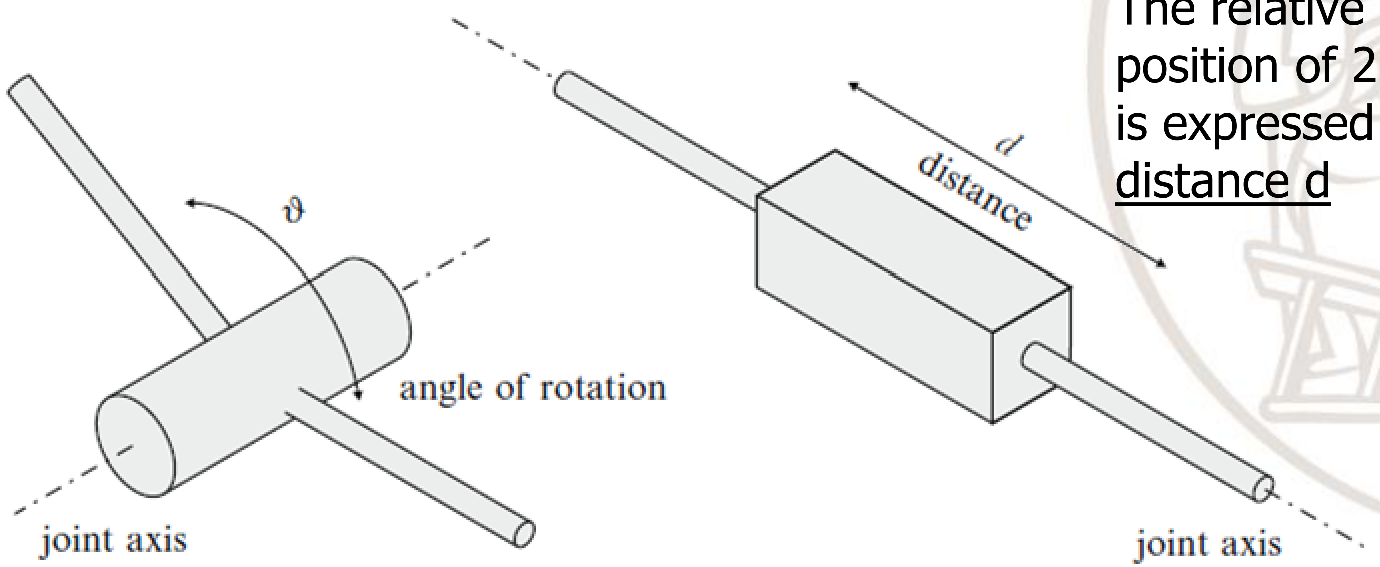
THE BIOROBOTICS INSTITUTE

Scuola Superiore Sant'Anna

Rotational Joint (revolute)

Translational Joint (prismatic)

The relative position of 2 links is expressed by an <u>angle θ</u>



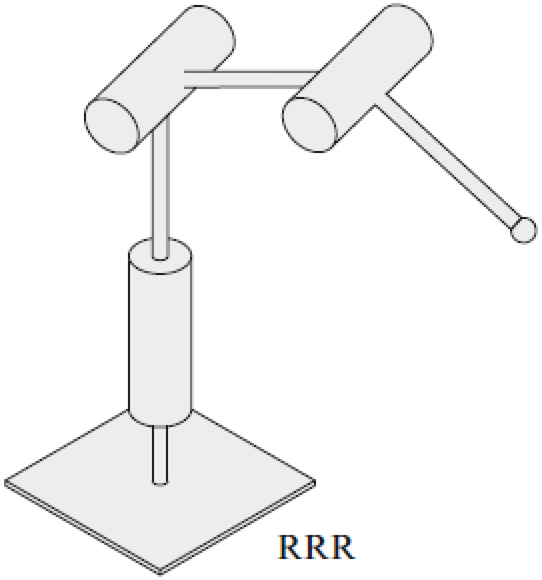
The relative position of 2 links is expressed by a <u>distance d</u>

Manipulator types

Fundamental categories:

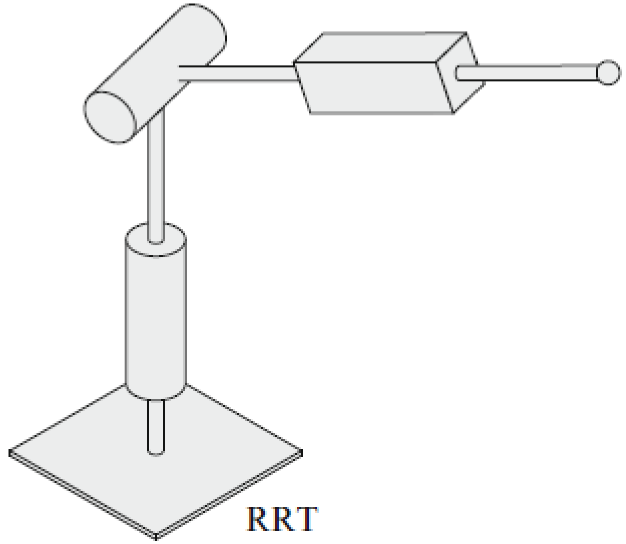
- Rotational (3 or more rotational joints) – RRR (also named *anthropomorphic*)
- Spherical (2 rotational joints and 1 translational joint) – RRT
- SCARA (2 rotational joints and 1 translational joint) – RRT (with 3 parallel axes)
- Cilindrical (1 rotational joint and 2 translational joints) – RTT
- Cartesian (3 translational joints) – TTT

Anthropomorfic



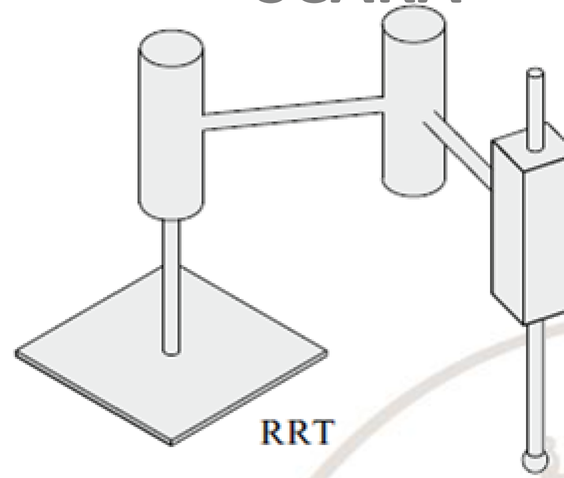
RRR

Spherical



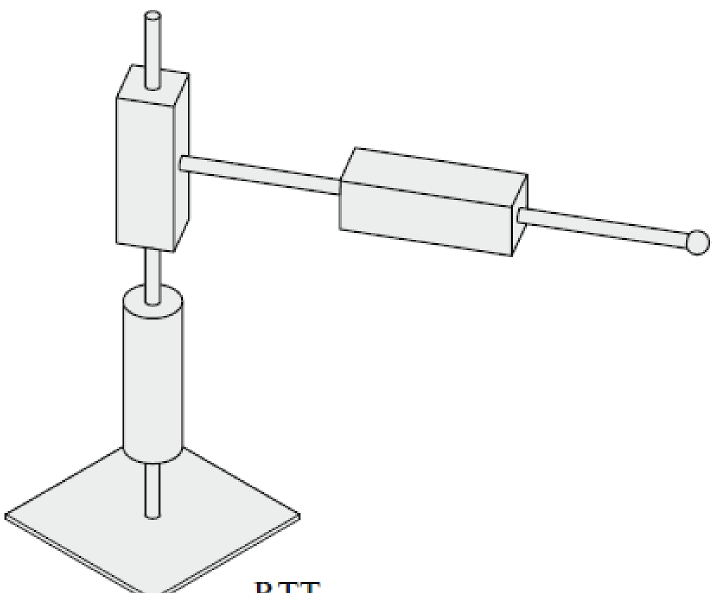
RRT

SCARA



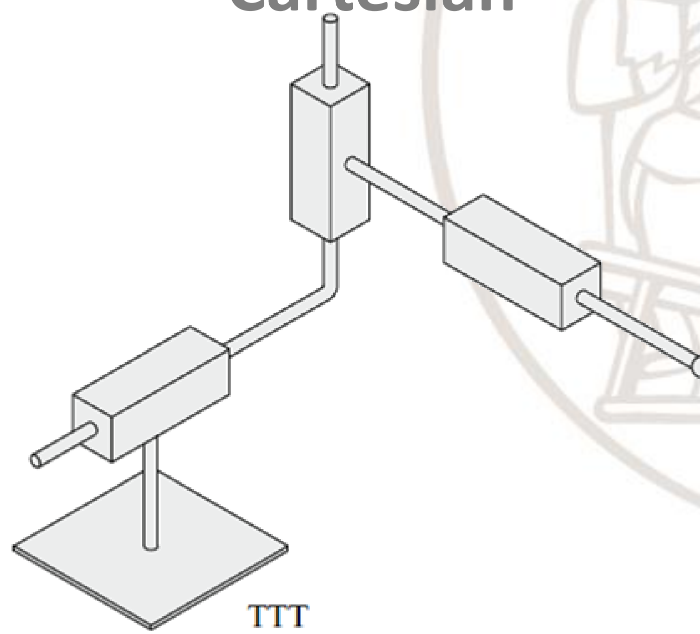
RRT

Cilindrical



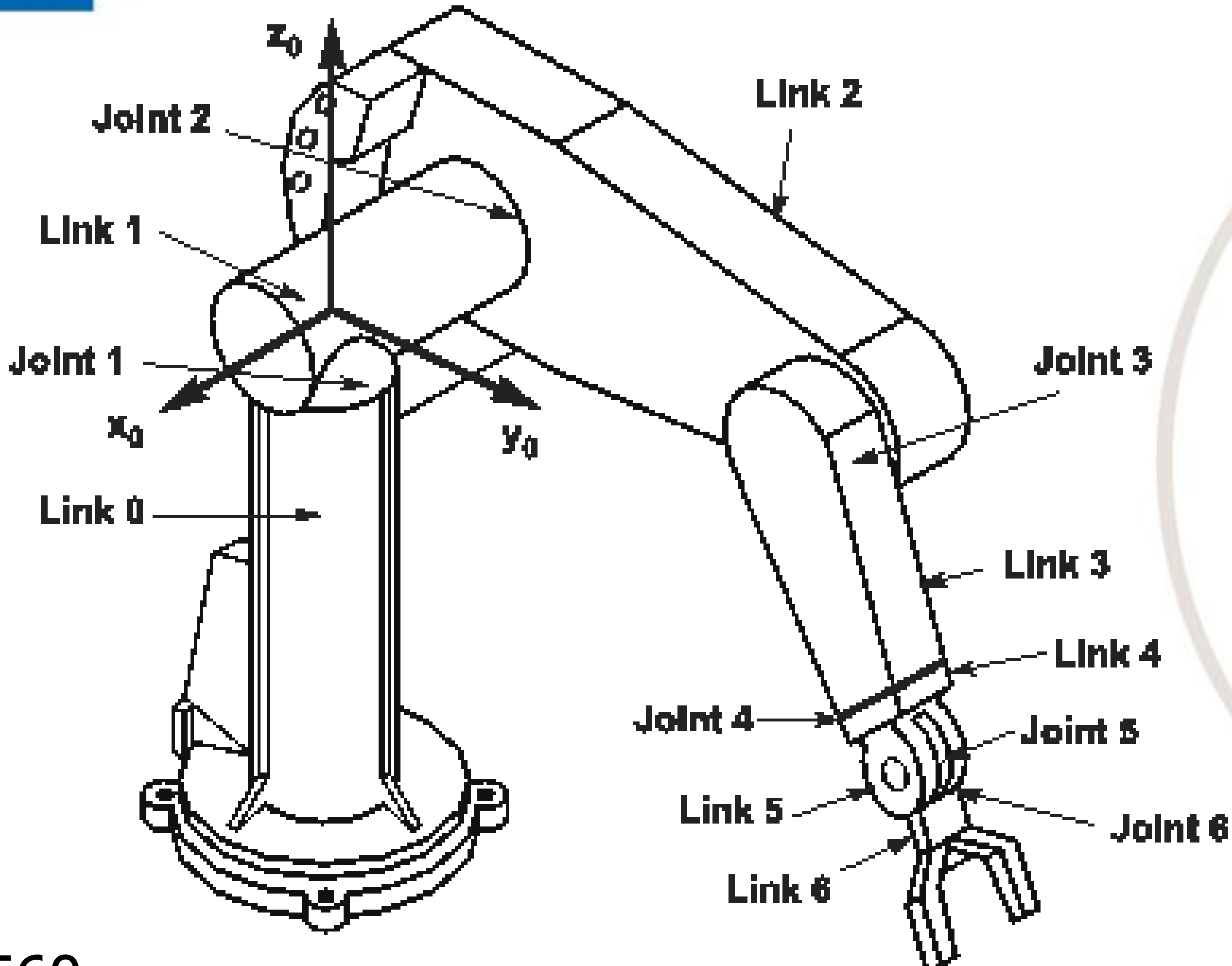
RTT

Cartesian



TTT

Robot manipulator



PUMA 560

Joint space and Cartesian space

- **Joint space** (or configuration space) is the space in which the q vector of joint variables are defined.
  Its dimension is indicated with N
  (N = number of joints in the robot).

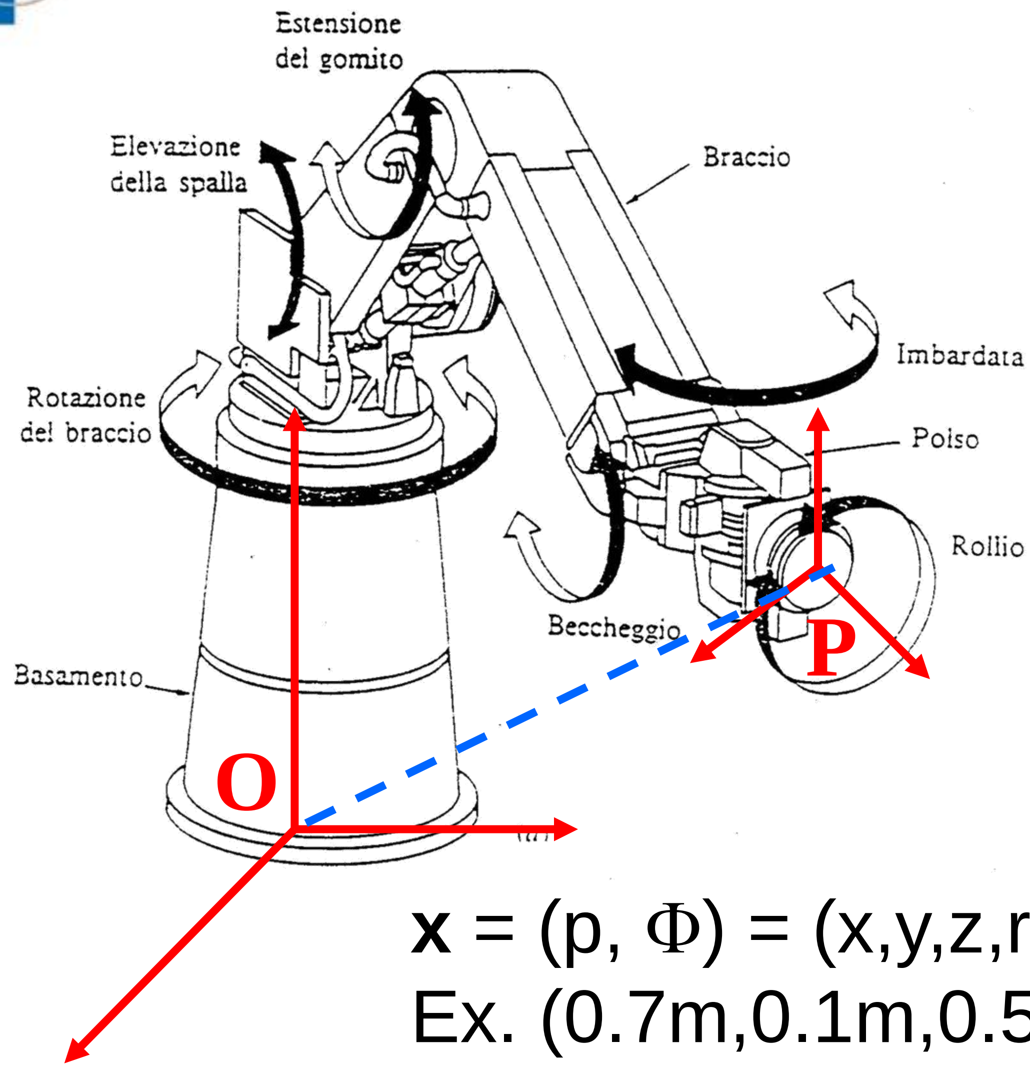
- **Cartesian space** (or operational space) is the space in which the $x = (p, \Phi)^T$ vector of the end-effector position is defined.
  Its dimension is indicated with M (M=6).

Robot position in joint space and in Cartesian space

- **q** is the vector of the robot position in joint space.
  It contains the joint variables,
  it has dimension N x 1,
  it is expressed in degrees.

- **x** = $(p, \Phi)^T$ is the vector of the robot position in Cartesian space.
  It contains:
  - **p**, vector of Cartesian coordinates of the end effector, which has dimension 3x1 (x,y,z coordinates).
  - **Φ**, vector of orientation of the end effector, which has dimension 3x1 (roll, pitch, yaw angles).

Robot manipulator



$\mathbf{x}$ = (p, Φ) = (x,y,z,roll,pitch,yaw)

Ex. (0.7m,0.1m,0.5m,10°,-45°,5°)

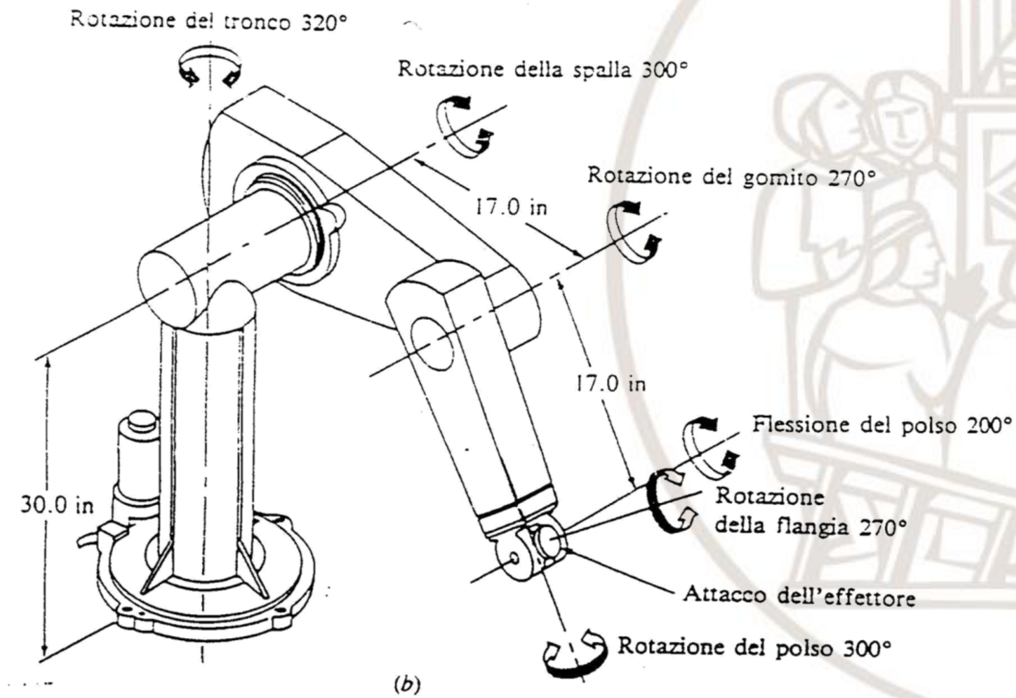
Robot manipulator

Tipically:

Main subgroups =
Supporting structure + wrist

The supporting structure tunes the position of the end effector

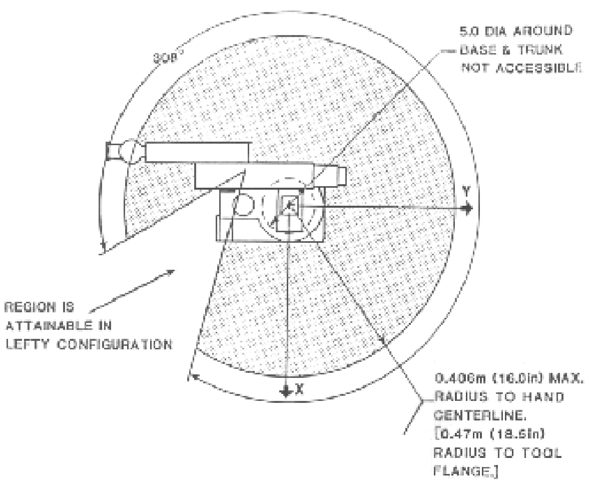
The wrist tunes the orientation of the end effector

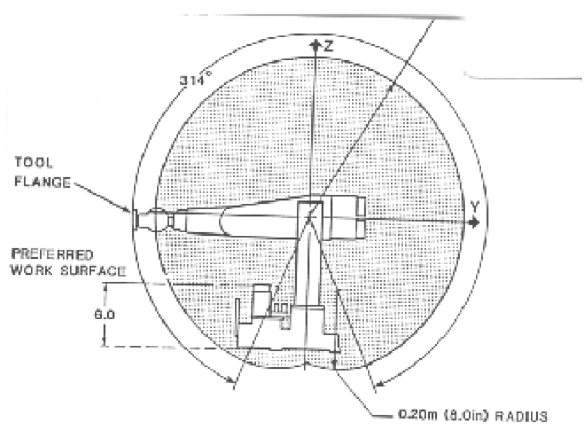


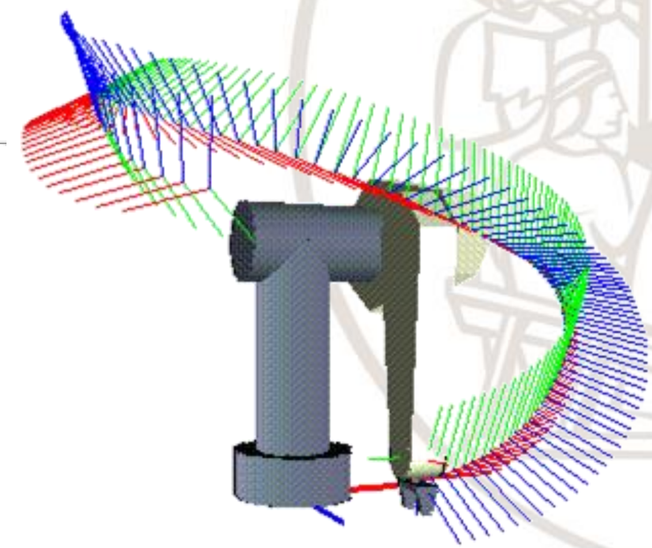
(b)

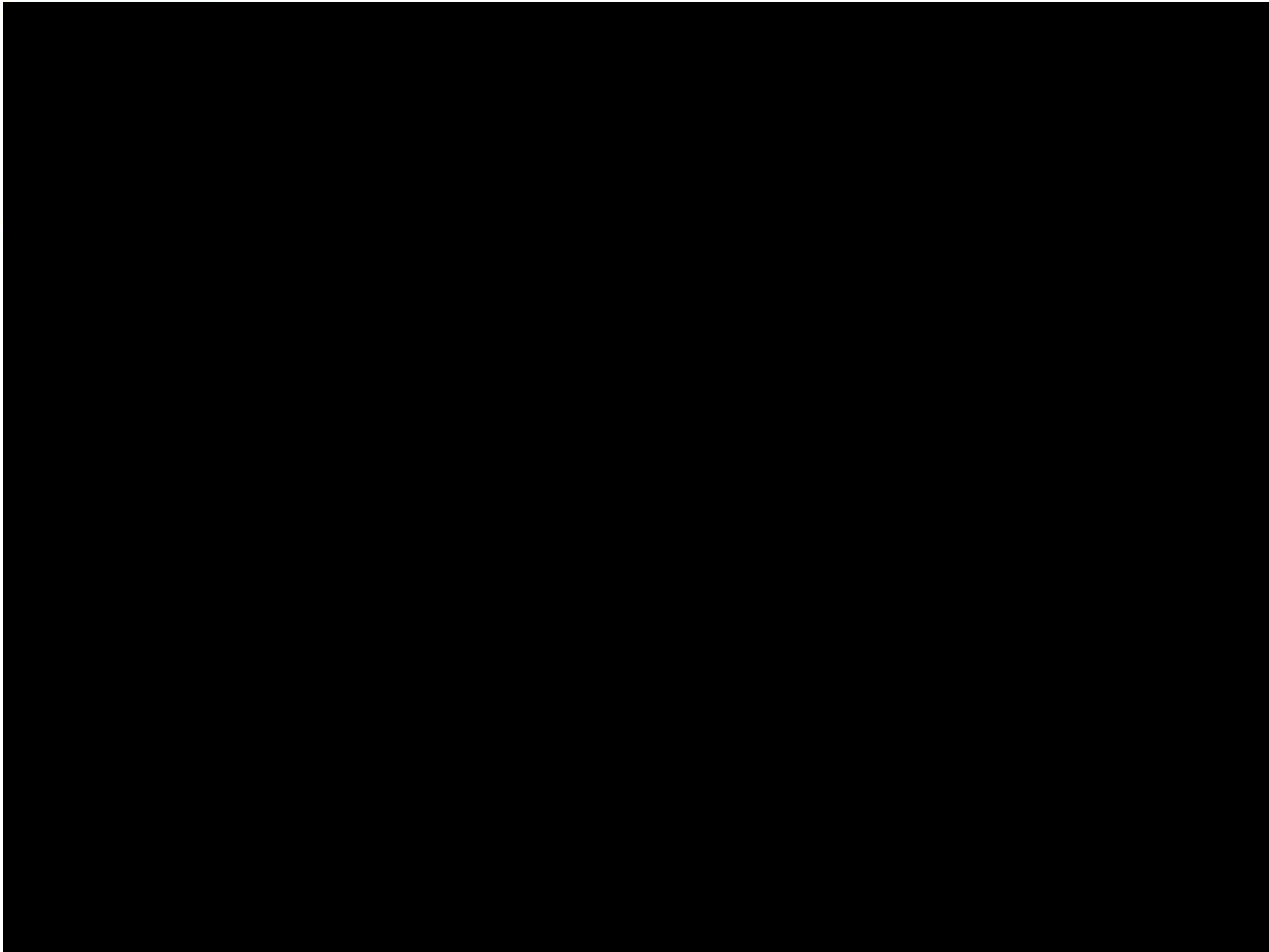
Workspace

Robot workspace = region described by the origin of the end effector when the robot joints execute all possible motions









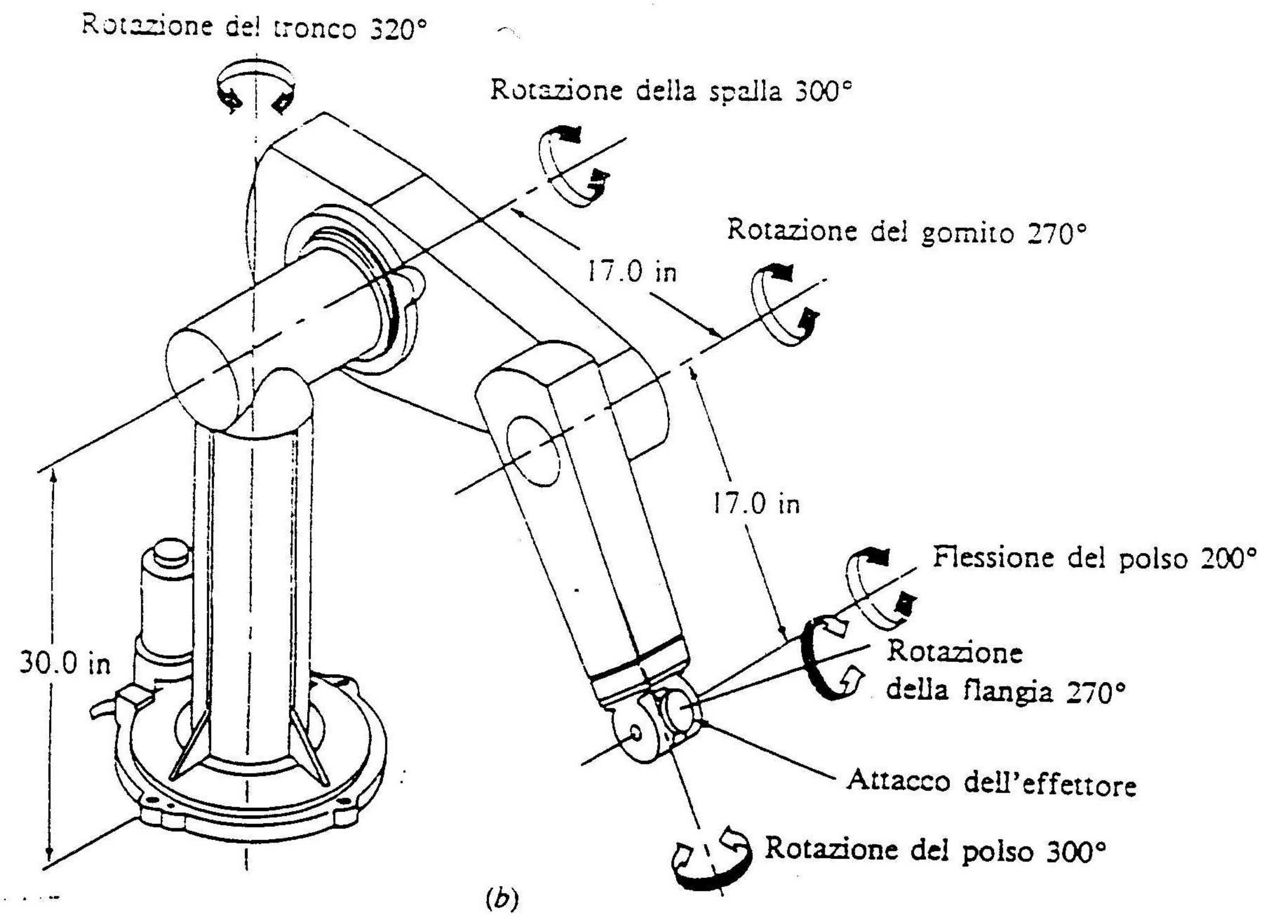
Workspace

- **Reachable workspace** = region of the space that the end effector can reach with *at least one* orientation.

- **Dextrous workspace** = region of the space that the end effector can reach with *more than one* orientation.

Workspace

It depends on
- Link lengths
- Joint ranges of motion



(b)

Robot arm kinematics

- Analytical study of the geometry of the arm motion, with respect to a steady Cartesian reference frame, without considering forces and torques which generate motion (actuation, inertia, friction, gravity, etc.).

- Analytical description of the relations between joint positions and the robot end effector position and orientation.

Kinematics transformations

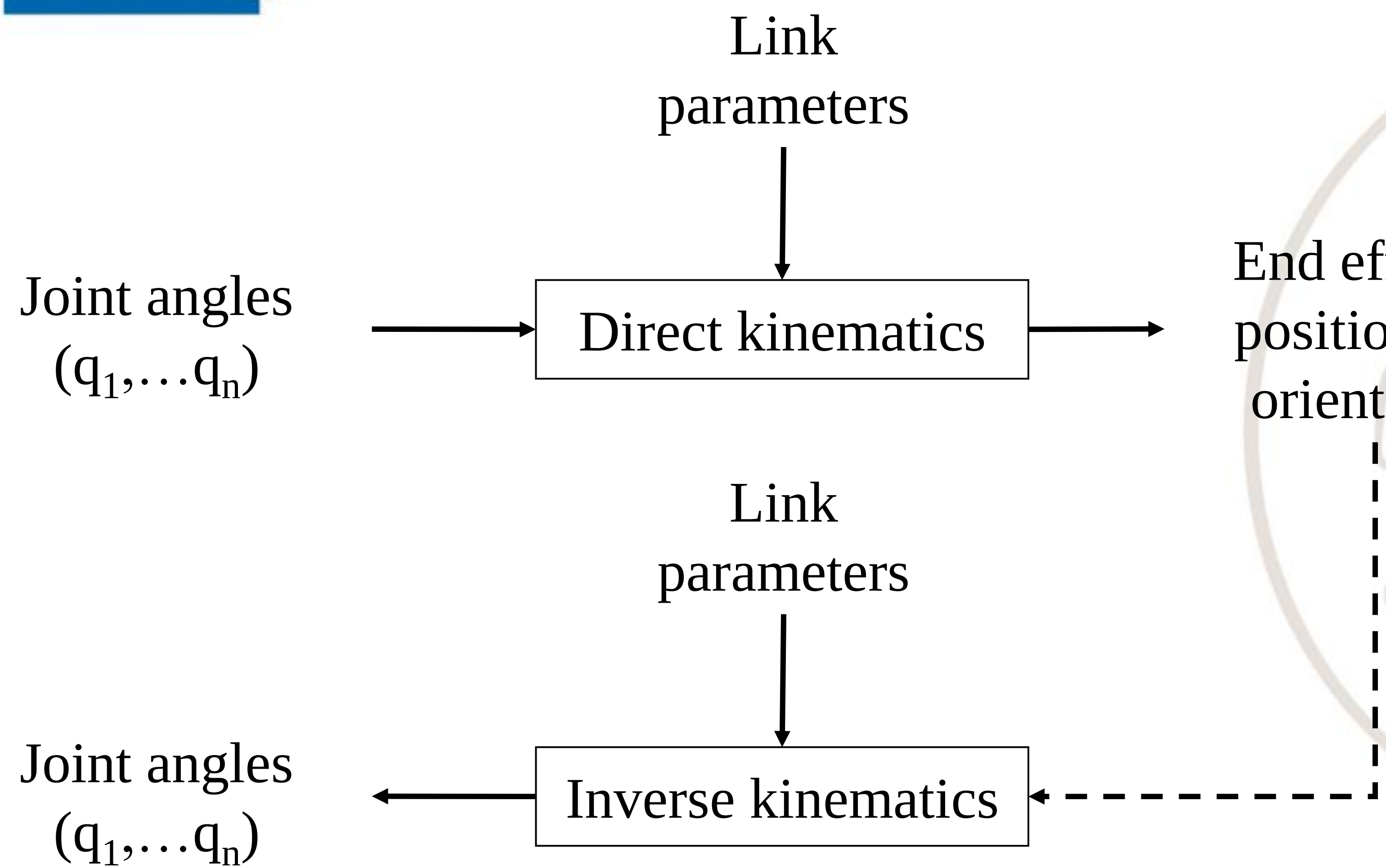
Direct kinematics:

- Computing the end-effector position in the Cartesian space, given the robot position in the joint space

Inverse kinematics:

- Computing the joint position for obtaining a desired position of the end effector in the Cartesian space

Direct and inverse kinematics



Direct kinematics problem

- For a given robot arm, given the vector of joint angles q and given the link geometric parameters, find the position and orientation of the end effector, with respect to a reference coordinate frame
- Find the vectorial non-linear function

$$\mathbf{x} = K(q) \qquad \textit{x unknown, q known}$$

$$\textit{Ex. PUMA } (x,y,z,\textit{roll},\textit{pitch},\textit{yaw}) = K(q_1,\ldots,q_6)$$

Inverse kinematics problem

THE BIOROBOTICS INSTITUTE

Scuola Superiore Sant'Anna

- For a given robot arm, given a desired position and orientation of the end effector, with respect to a reference coordinate frame, find the corresponding joint variables
- Find the vectorial non-linear function

$$\mathbf{q} = K^{-1}(x) \qquad \textit{q unknown, x known}$$

Ex. PUMA $(q_1,\ldots,q_6) = K^{-1}\,(x,y,z,roll,pitch,yaw)$

Kinematics redundancy

Number of degrees of freedom higher than the number of variables needed for characterizing a task ⇔ The operational space size is smaller that the joint space size

The number of redundancy degrees is R=N-M

Advantages: multiple solutions

Disadvantages: computing and control complexity

Inverse kinematics problem

THE BIOROBOTICS INSTITUTE

Scuola Superiore Sant'Anna

- The equations to solve are generally non linear
- It is not always possible to find an analytical solution
- There can be multiple solutions
- There can be infinite solutions (redundant robots)
- There may not be possible solutions, for given arm kinematic structures
- The existence of a solution is guaranteed if the desired position and the orientation belong to the robot dextrous workspace

Recall of transformation matrices

Matrices for translations and rotations of reference coordinate frames

Rotation matrices

A rotation matrix operates on a position vector in a 3D space.

A rotation matrix transforms the coordinates of the vector expressed in a reference system OUVW in the coordinates expressed in a reference system OXYZ.

OXYZ is the reference system in the 3D space.

OUVW is the reference system of the rigid body which moves together with it.

Rotation matrices

THE BIOROBOTICS INSTITUTE

Scuola Superiore Sant'Anna

$$p_{xyz} = Rp_{uvw}$$

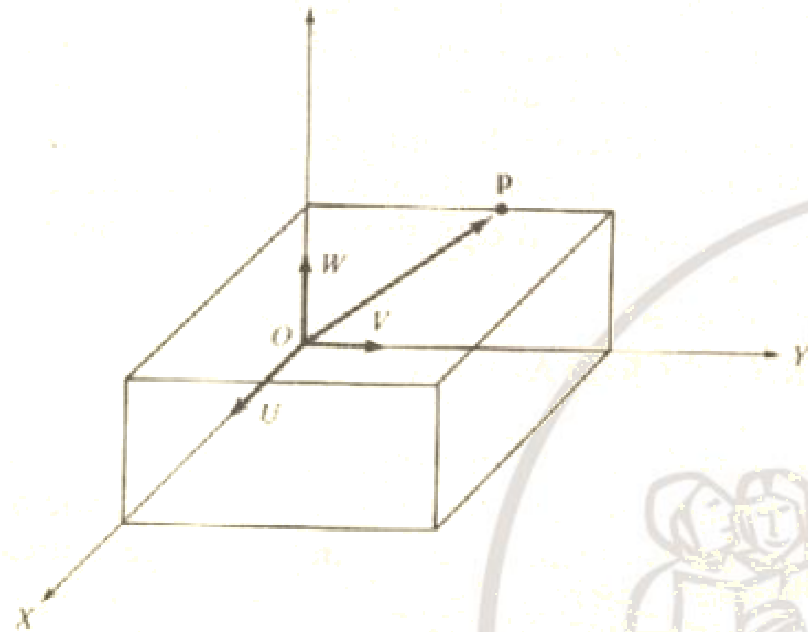


Figura 2.2 Sistemi di coordinate di riferimento e solidali al corpo.

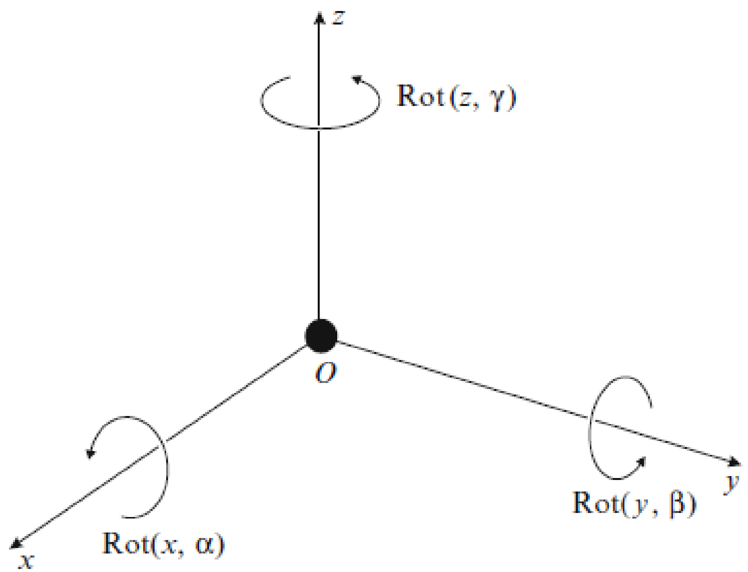
is the relation transforming
the coordinates of the vector p_{uvw} expressed in the reference system OUVW
in the coordinates of the vector p_{xyz} expressed in the reference system OXYZ.

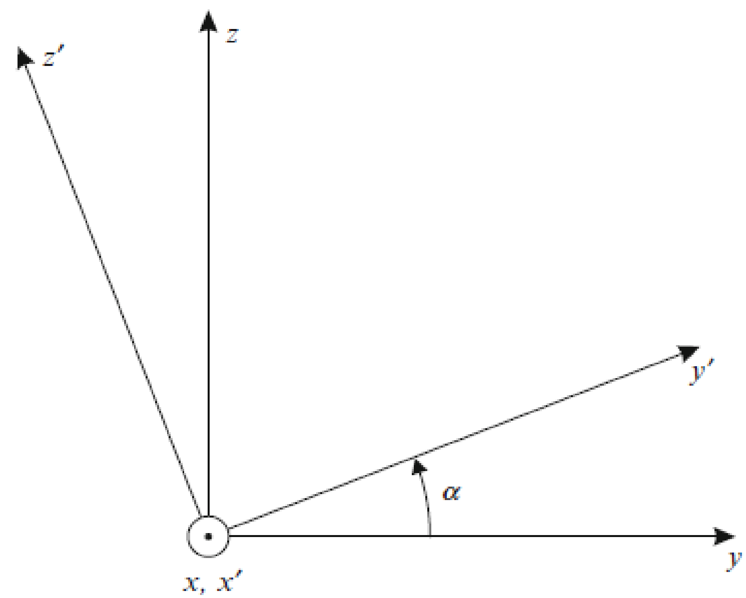
R is the 3x3 rotation matrix between the two frames OUVW and OXYZ

THE BIOROBOTICS INSTITUTE

Scuola Superiore Sant'Anna

Rotation matrices





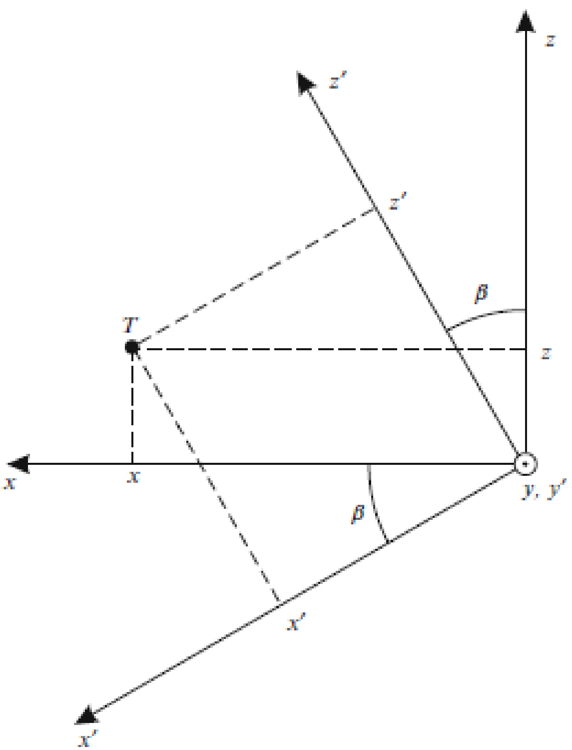
$$Rot(x,\alpha) = \begin{array}{c} \\ \\ \end{array} \begin{array}{cccccc} & x' & y' & z' & & \\ & \left[\begin{matrix} \cos 0^\circ & \cos 90^\circ & \cos 90^\circ & 0 \\ \cos 90^\circ & \cos\alpha & \cos(90^\circ+\alpha) & 0 \\ \cos 90^\circ & \cos(90^\circ-\alpha) & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{matrix}\right] & & & & \begin{matrix} x \\ y \\ z \\ \end{matrix} \end{array} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

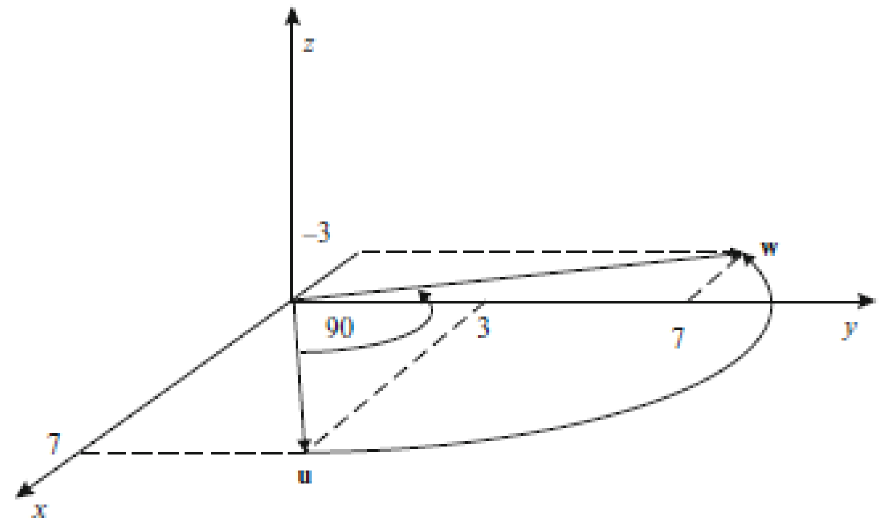
The angle between the x' and the x axes is 0°, therefore we have $\cos 0^\circ$ in the intersection of the x' column and the x row. The angle between the x' and the y axes is 90°, we put $\cos 90^\circ$ in the corresponding intersection. The angle between the y' and the y axes is α, the corresponding matrix element is $\cos\alpha$.

Rotation matrices

THE BIOROBOTICS INSTITUTE

Scuola Superiore Sant'Anna





$$
Rot(y,\beta)=\begin{array}{cc}\begin{bmatrix}\cos\beta & 0 & \sin\beta & 0\\ 0 & 1 & 0 & 0\\ -\sin\beta & 0 & \cos\beta & 0\\ 0 & 0 & 0 & 1\end{bmatrix} & \begin{array}{c} x\\ y\\ z\\ \ \end{array}\end{array}
$$

$$
Rot(z,\gamma)=\begin{bmatrix}\cos\gamma & -\sin\gamma & 0 & 0\\ \sin\gamma & \cos\gamma & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}
$$

$$
\mathbf{w}=\begin{bmatrix}0 & -1 & 0 & 0\\ 1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}7\\ 3\\ 0\\ 1\end{bmatrix}=\begin{bmatrix}-3\\ 7\\ 0\\ 1\end{bmatrix}
$$

Fundamental rotation matrices

Rotation around the X axis

$$R_{x,\,\alpha} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

Rotation around the Y axis

$$R_{y,\,\phi} = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & \\ -\sin\phi & 0 & \cos\phi \end{bmatrix}$$

Rotation around the Z axis

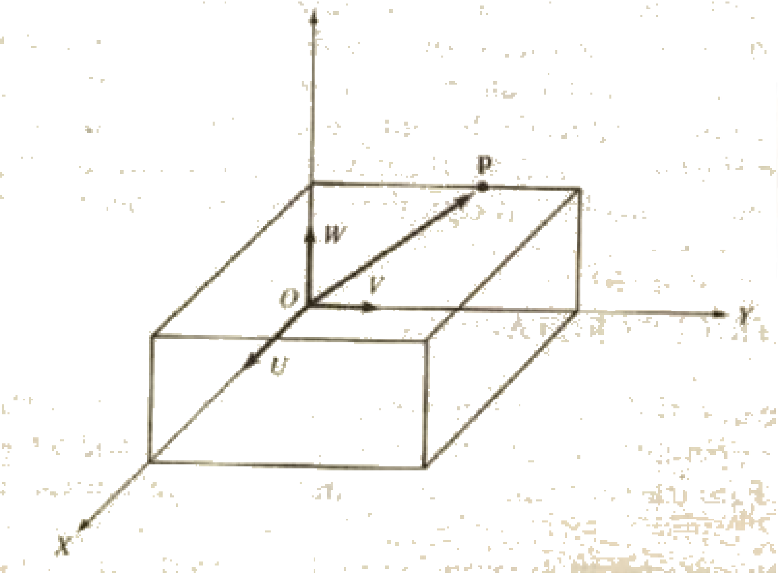
$$R_{z,\,\theta} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Composed rotation matrices

- The fundamental rotation matrices can be multiplied to represent a sequence of rotations around the main axes of the reference frame:

$$R = R_{x,\alpha} R_{y,\phi} R_{z,\theta}$$

$$p_{xyz} = Rp_{uvw}$$



- Please note: matrix product is not commutative

Homogeneous coordinates

THE BIOROBOTICS INSTITUTE

Scuola Superiore Sant'Anna

Representation of a position vector of size N with a vector of size (N+1)

$P = (p_x, p_y, p_z)^T$ $\quad$ $P\hat{} = (wp_x, wp_y, wp_z, w)^T$

w = scaling factor

In robotics w = 1.

Unified representation of translation, rotation, perspective and scaling.

Homogeneous rotation matrices

Rotation around the X axis

$$R_{x,\alpha} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotation around the Y axis

$$R_{y,\phi} = \begin{bmatrix} \cos\phi & 0 & \sin\phi & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\phi & 0 & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

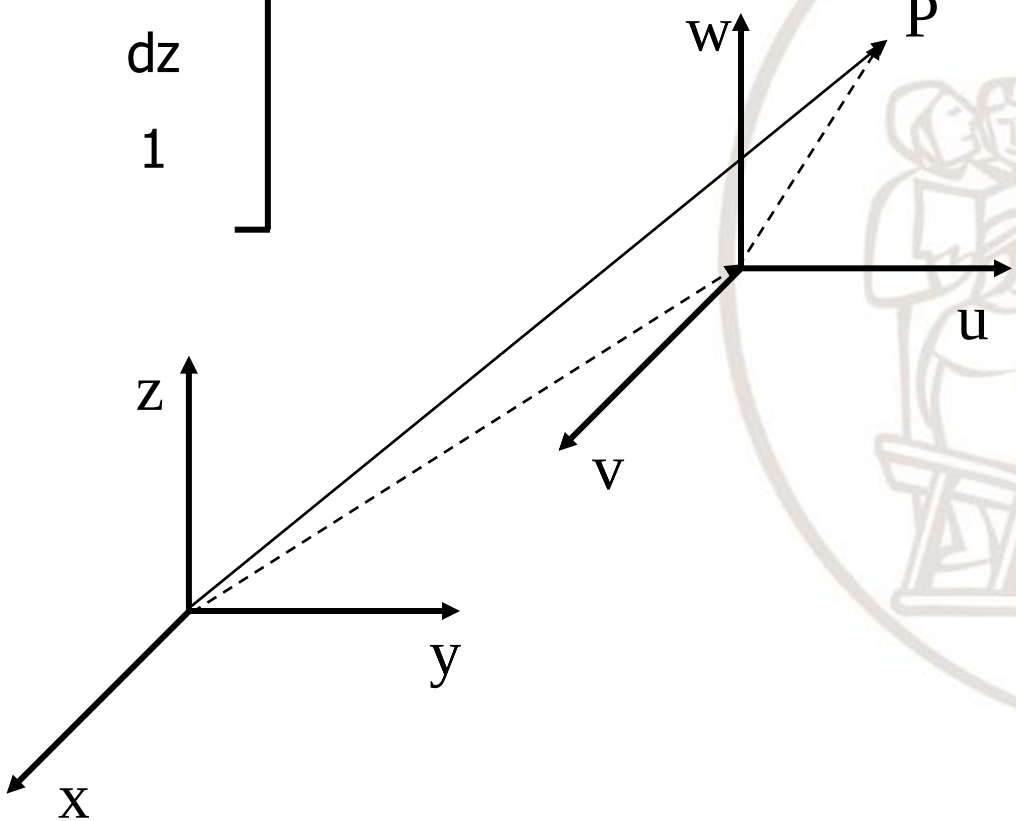
Rotation around the Z axis

$$R_{z,\theta} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Fundamental homogeneous translation matrix

$$T_{tran} = \begin{bmatrix} 1 & 0 & 0 & dx \\ 0 & 1 & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

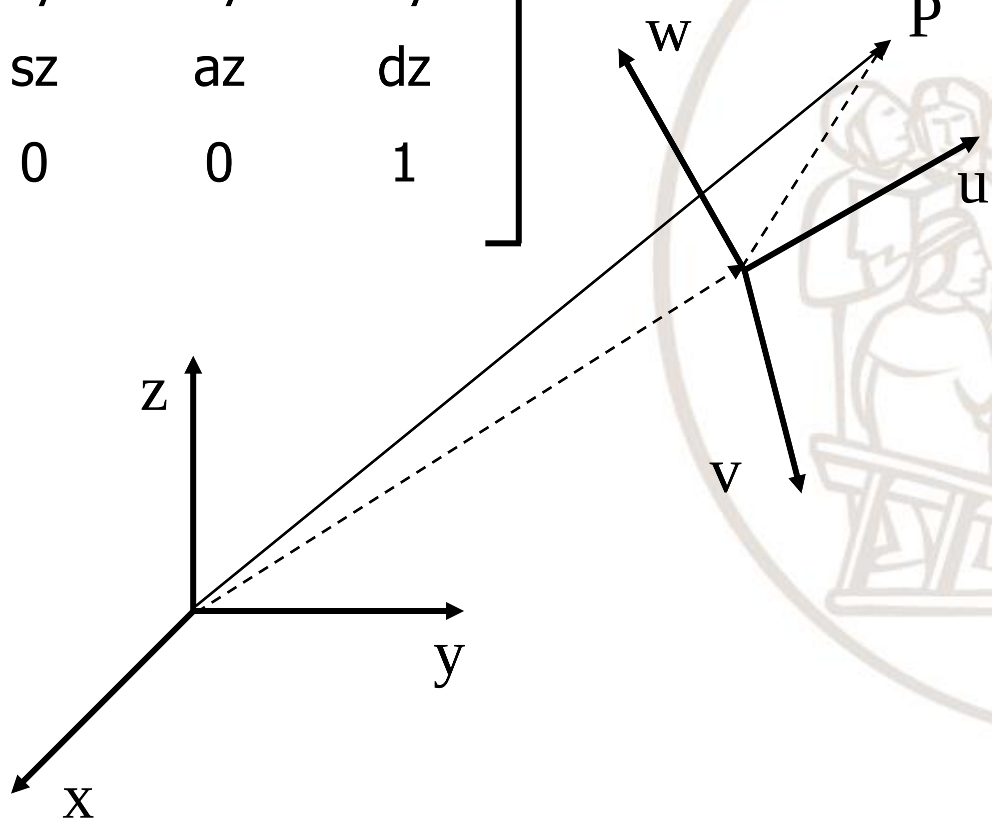
$$P_{xyz} = T_{tran} \, P_{vuw}$$



Homogeneous transformation matrix: rotation and translation

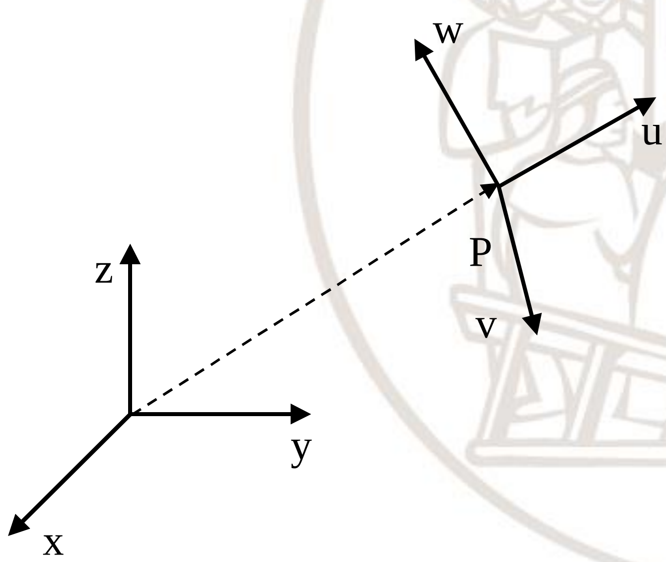
$$T=\left[\begin{array}{c:c} R_{3x3} & p_{3x1} \\ \hdashline f_{1x3} & 1_{1x1} \end{array}\right]=\begin{bmatrix} nx & sx & ax & dx \\ ny & sy & ay & dy \\ nz & sz & az & dz \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$p_{xyz} = T\, p_{vuw}$$



Geometric interpretation of tranformation matrices

$$T = \begin{bmatrix} nx & sx & ax & dx \\ ny & sy & ay & dy \\ nz & sz & az & dz \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{vmatrix} \mathbf{n} & \mathbf{s} & \mathbf{a} & \mathbf{p} \\ 0 & 0 & 0 & 1 \end{vmatrix}$$



p = origin of OUVW with respect to OXYZ

n,s,a representation of the orientation of the frame OUVW with respect to OXYZ

Composite homogeneous tranformation matrices

Homogeneous matrices for rotation and translation can be multiplied to obtain a composite matrix (T)

$$T = T^0{}_1 T^1{}_2 \;....\; T^{n-1}{}_n$$

$$p^0 = T^0{}_1 T^1{}_2 \;....\; T^{n-1}{}_n\, p^n = T\, p^n$$

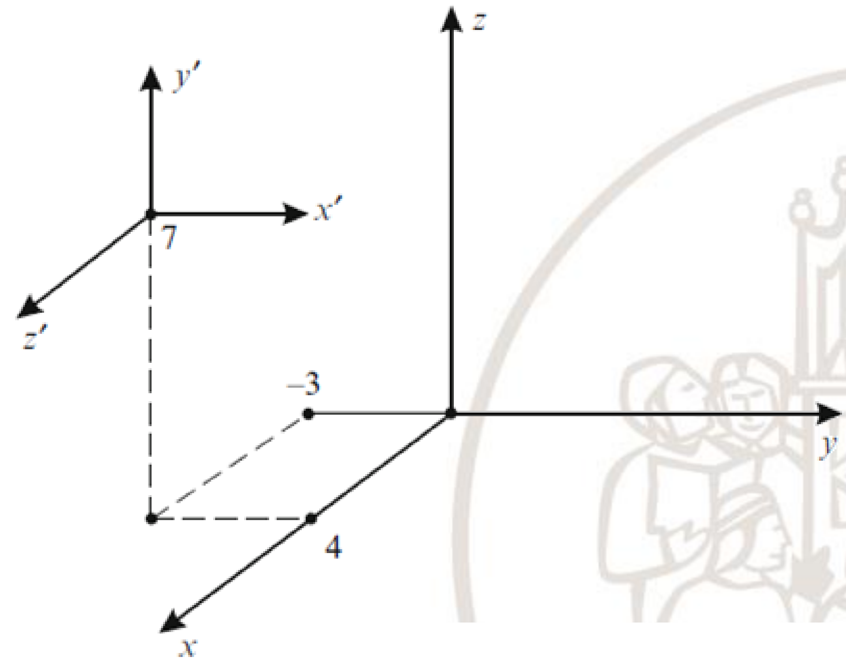
Example of transformation of a reference frame

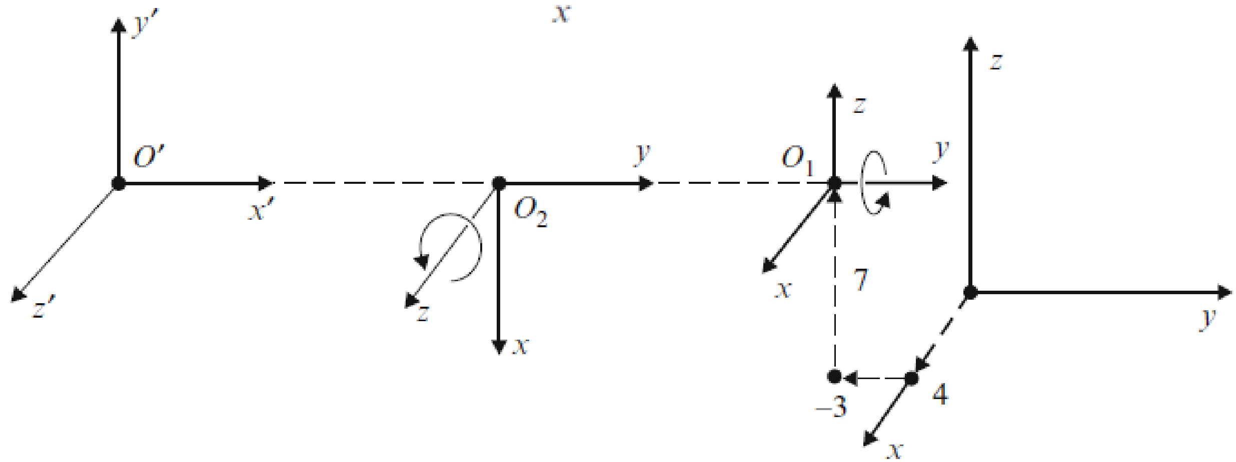
$$\mathbf{H} = Trans(4,-3,7)Rot(y,90^\circ)Rot(z,90^\circ)$$

$$= \begin{bmatrix} 1 & 0 & 0 & 4 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & 1 & 7 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

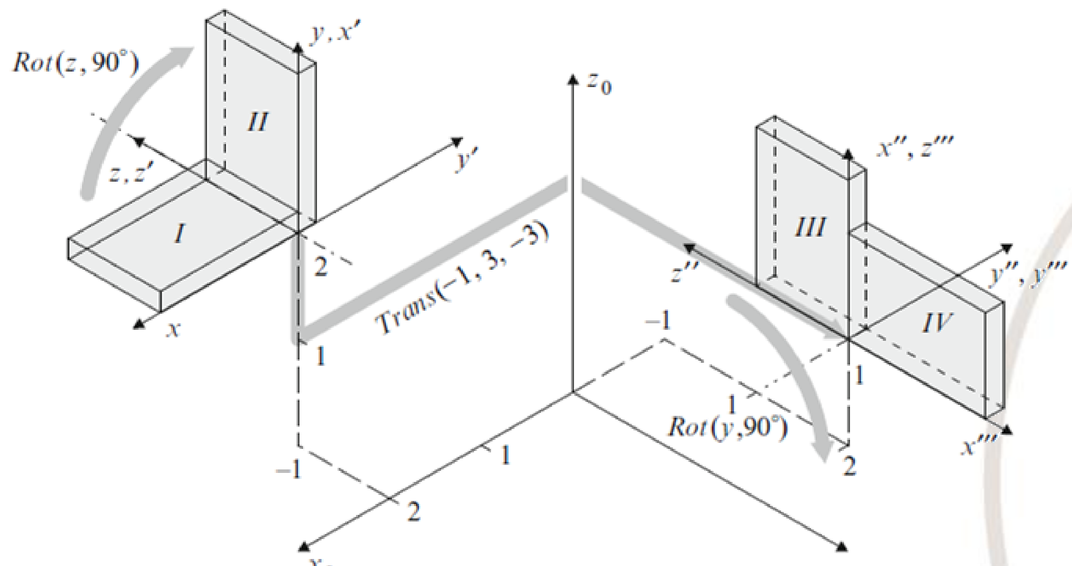
$$= \begin{bmatrix} 0 & 0 & 1 & 4 \\ 1 & 0 & 0 & -3 \\ 0 & 1 & 0 & 7 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

$$\begin{array}{cc} \begin{array}{cccc} x' & y' & z' & \end{array} & \\ \begin{bmatrix} \begin{bmatrix}0\end{bmatrix} & \begin{bmatrix}0\end{bmatrix} & \begin{bmatrix}1\end{bmatrix} & 4 \\ 1 & 0 & 0 & -3 \\ \begin{bmatrix}0\end{bmatrix} & \begin{bmatrix}1\end{bmatrix} & \begin{bmatrix}0\end{bmatrix} & 7 \\ 0 & 0 & 0 & 1 \end{bmatrix} & \begin{array}{c} x \\ y \\ z \\ \end{array} \end{array}$$





Example of transformation of an object position



$$\mathbf{D} = Rot(z,90^\circ)\cdot Trans(-1,3,-3)\cdot Rot(y'',90^\circ)$$

$$= \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & -3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & -1 & 0 & -3 \\ 0 & 0 & 1 & -1 \\ -1 & 0 & 0 & -3 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

$$\mathbf{H}_3 = \mathbf{H}\cdot\mathbf{D} = \begin{bmatrix} 1 & 0 & 0 & 2 \\ 0 & 0 & -1 & -1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 & -3 \\ 0 & 0 & 1 & -1 \\ -1 & 0 & 0 & -3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

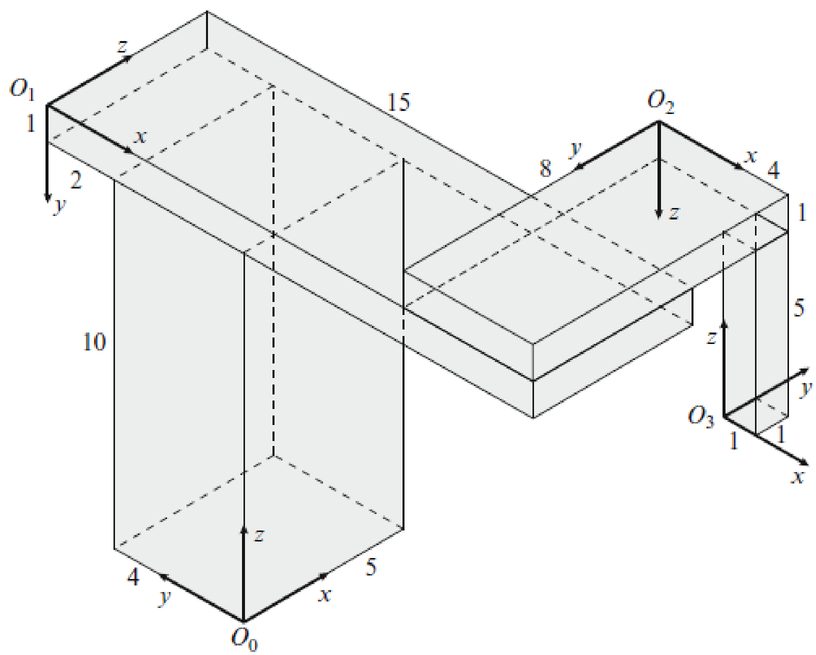
$$= \begin{array}{cc} \begin{array}{cccc} x''' & y''' & z''' & \end{array} & \\ \begin{bmatrix} 0 & -1 & 0 & -1 \\ 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} & \begin{array}{c} x_0 \\ y_0 \\ z_0 \\ \end{array} \end{array}.$$

THE BIOROBOTICS INSTITUTE

Scuola Superiore Sant'Anna

Generic manipulator model

$$ {}^{0}\mathbf{H}_3 = {}^{0}\mathbf{H}_1\,{}^{1}\mathbf{H}_2\,{}^{2}\mathbf{H}_3. $$



$$
{}^{0}\mathbf{H}_1 = \begin{array}{c}
\overbrace{\begin{array}{ccc} x & y & z \end{array}}^{O_1} \\
\left[\begin{array}{cccc} 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 6 \\ 0 & -1 & 0 & 11 \\ 0 & 0 & 0 & 1 \end{array}\right]
\end{array}
\left.\begin{array}{c} x \\ y \\ z \\ \end{array}\right\} O_0
$$

$$
{}^{1}\mathbf{H}_2 = \begin{bmatrix} 1 & 0 & 0 & 11 \\ 0 & 0 & 1 & -1 \\ 0 & -1 & 0 & 8 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{2}\mathbf{H}_3 = \begin{bmatrix} 1 & 0 & 0 & 3 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 6 \\ 0 & 0 & 0 & 1 \end{bmatrix}.
$$

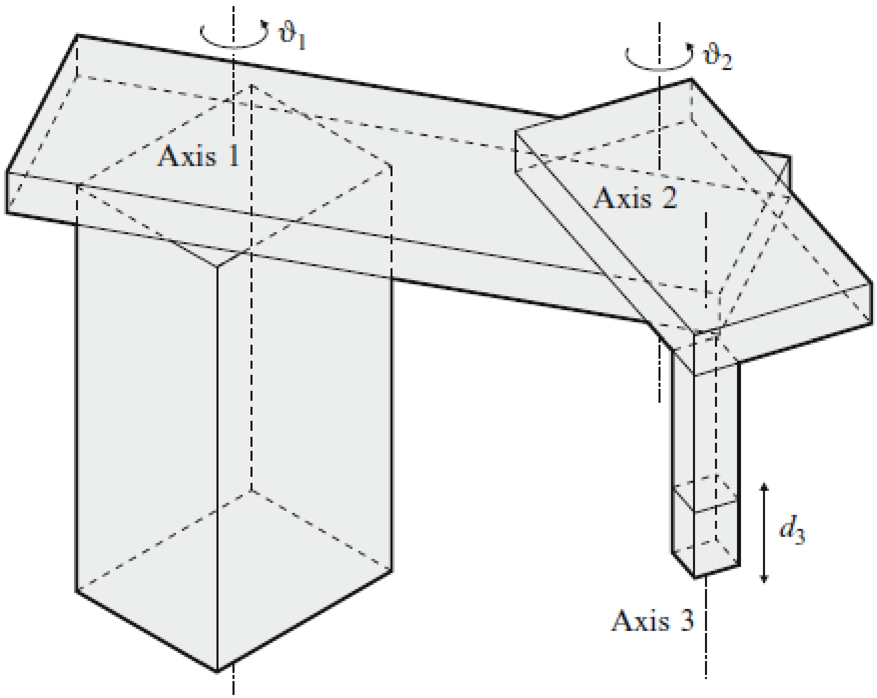
$$
{}^{0}\mathbf{H}_3 = \begin{bmatrix} 0 & 1 & 0 & 7 \\ -1 & 0 & 0 & -8 \\ 0 & 0 & 1 & 6 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

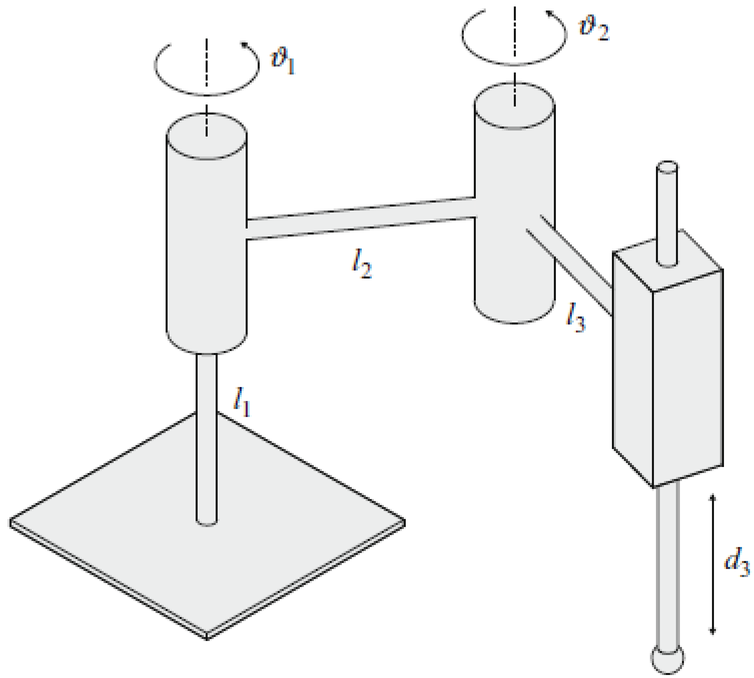
Our final goal is the geometrical model of a robot manipulator. A geometrical robot model is given by the description of the pose of the last segment of the robot (end effector) expressed in the reference (base) frame. The knowledge how to describe the pose of an object by the use of homogenous transformation matrices will be first applied to the process of assembly. For this purpose a mechanical assembly consisting of four blocks will be considered.

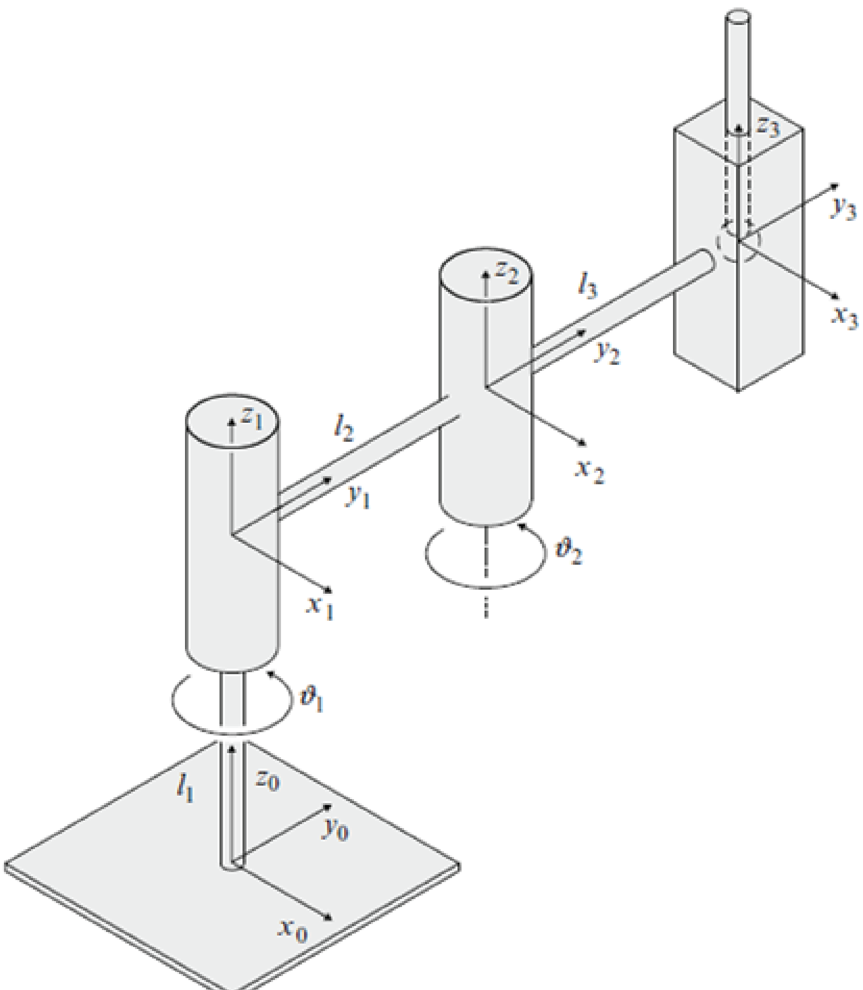
A plate with dimensions (5×15×1) is placed over a block (5×4×10). Another plate (8×4×1) is positioned perpendicularly to the first one, holding another small block (1×1×5).

A frame is attached to each of the four blocks. Our task will be to calculate the pose of the O_3 frame with respect to the reference frame O_0.

Geometric manipulator model







$$ {}^0\mathbf{H}_3 = ({}^0\mathbf{H}_1\mathbf{D}_1)\cdot({}^1\mathbf{H}_2\mathbf{D}_2)\cdot({}^2\mathbf{H}_3\mathbf{D}_3). $$

In equation (2.24) the matrices ${}^0\mathbf{H}_1$, ${}^1\mathbf{H}_2$, and ${}^2\mathbf{H}_3$ describe the pose of each joint frame with respect to the preceding frame in the same way as in the case of assembly of the blocs. From Figure 2.11 it is evident that the $\mathbf{D}_1$ matrix represents a rotation around the positive z_1 axis. The following product of two matrices describes the pose and the displacement in the first joint

$$ {}^0\mathbf{H}_1\mathbf{D}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c1 & -s1 & 0 & 0 \\ s1 & c1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c1 & -s1 & 0 & 0 \\ s1 & c1 & 0 & 0 \\ 0 & 0 & 1 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}. $$

In the above matrices the following shorter notation was used: $\sin\vartheta_1 = s1$ and $\cos\vartheta_1 = c1$.

In the second joint there is a rotation around the z_2 axis

$$ {}^1\mathbf{H}_2\mathbf{D}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & l_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c2 & -s2 & 0 & 0 \\ s2 & c2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c2 & -s2 & 0 & 0 \\ s2 & c2 & 0 & l_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. $$

In the last joint there is translation along the z_3 axis

$$ {}^2\mathbf{H}_3\mathbf{D}_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & l_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & l_3 \\ 0 & 0 & 1 & -d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}. $$

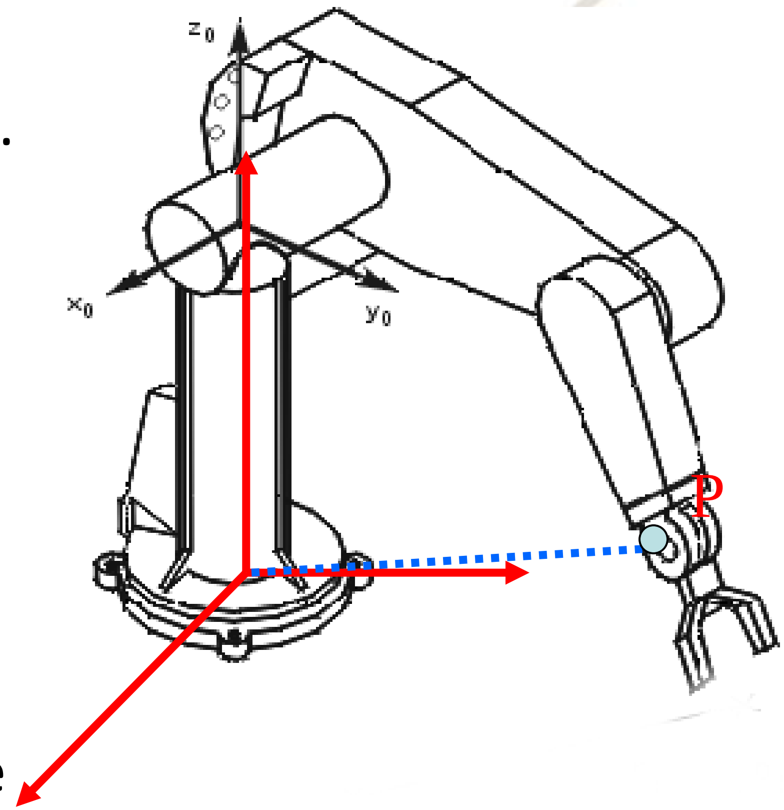
The geometrical model of the SCARA robot manipulator is obtained by postmultiplication of the three matrices derived above

$$ {}^0\mathbf{H}_3 = \begin{bmatrix} c12 & -s12 & 0 & -l_3s12 - l_2s1 \\ s12 & c12 & 0 & l_3c12 + l_2c1 \\ 0 & 0 & 1 & l_1 - d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}. $$

When multiplying the three matrices the following abbreviation was introduced $c12 = \cos(\vartheta_1 + \vartheta_2) = c1c2 - s1s2$ and $s12 = \sin(\vartheta_1 + \vartheta_2) = s1c2 + c1s2$.

Direct kinematics
Denavit-Hartenberg (D-H) representation

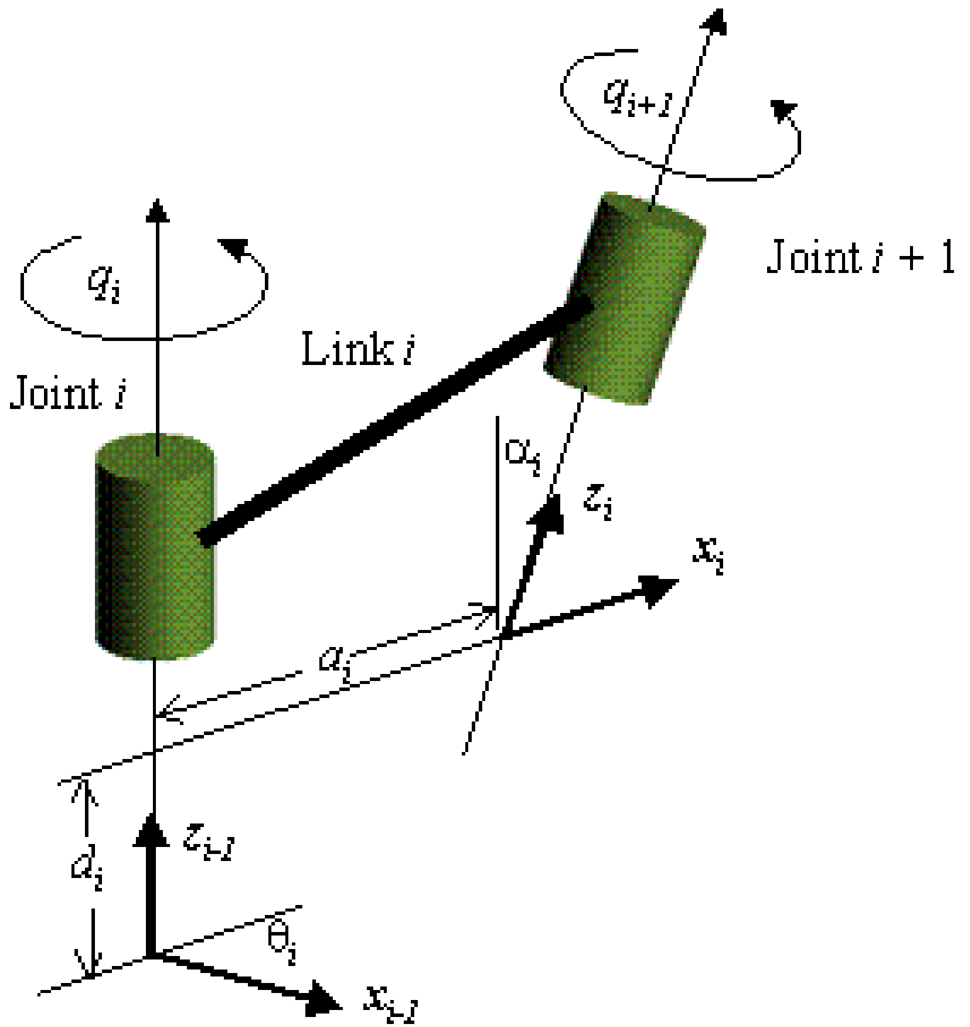
- Matrix-based method for describing the relations (rotations and translations) between adjacent links.
- D-H representation consists of homogeneous 4x4 transformation matrices, which represent each link reference frame with respect to the previous link.
- Through a sequence of transformations, the position of the end effector can be expressed in the base frame coordinates



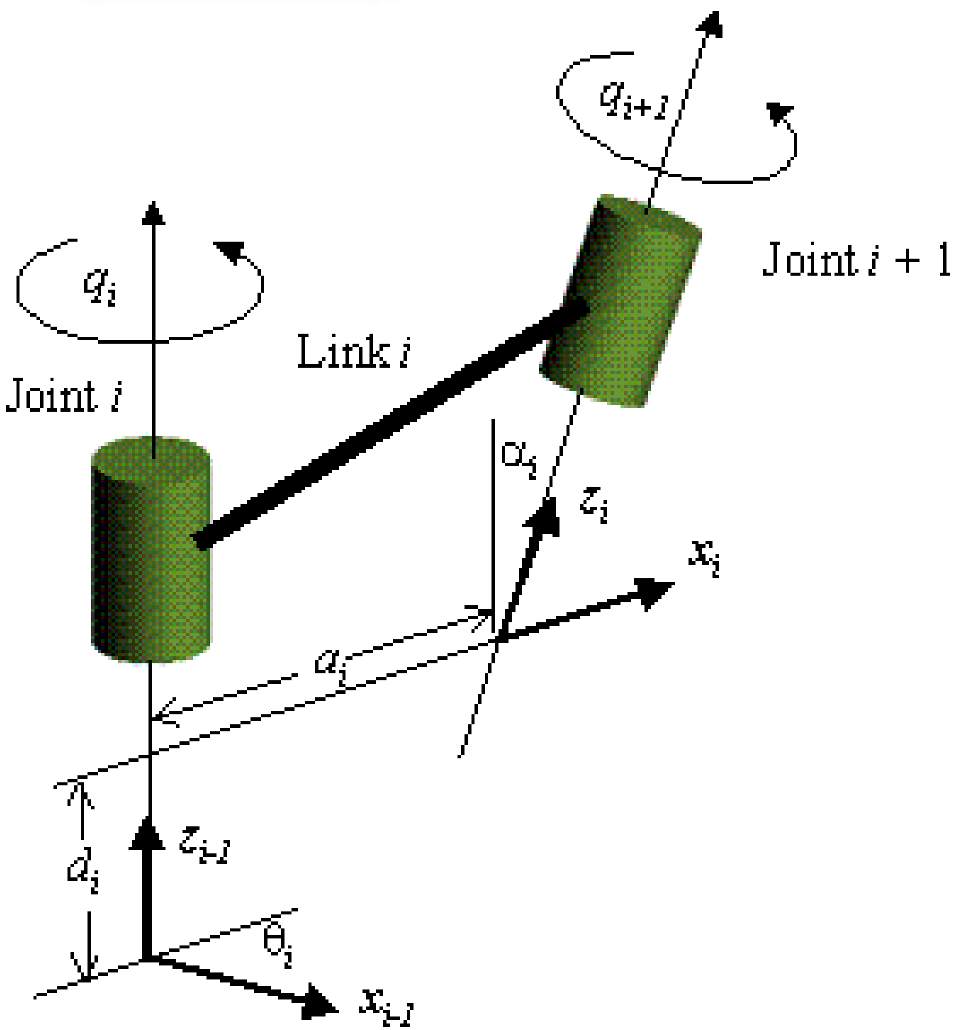
Link coordinate frames and their geometric parameters

- 4 geometric parameters are associated to each link:
  - 2 of them describe the relative position of adjacent link (joint parameters)
  - 2 of them describe the link structure
- The homogeneous transformation matrices depend on such geometric parameters, of which **only one is unknown**

Link coordinate frames and their geometric parameters

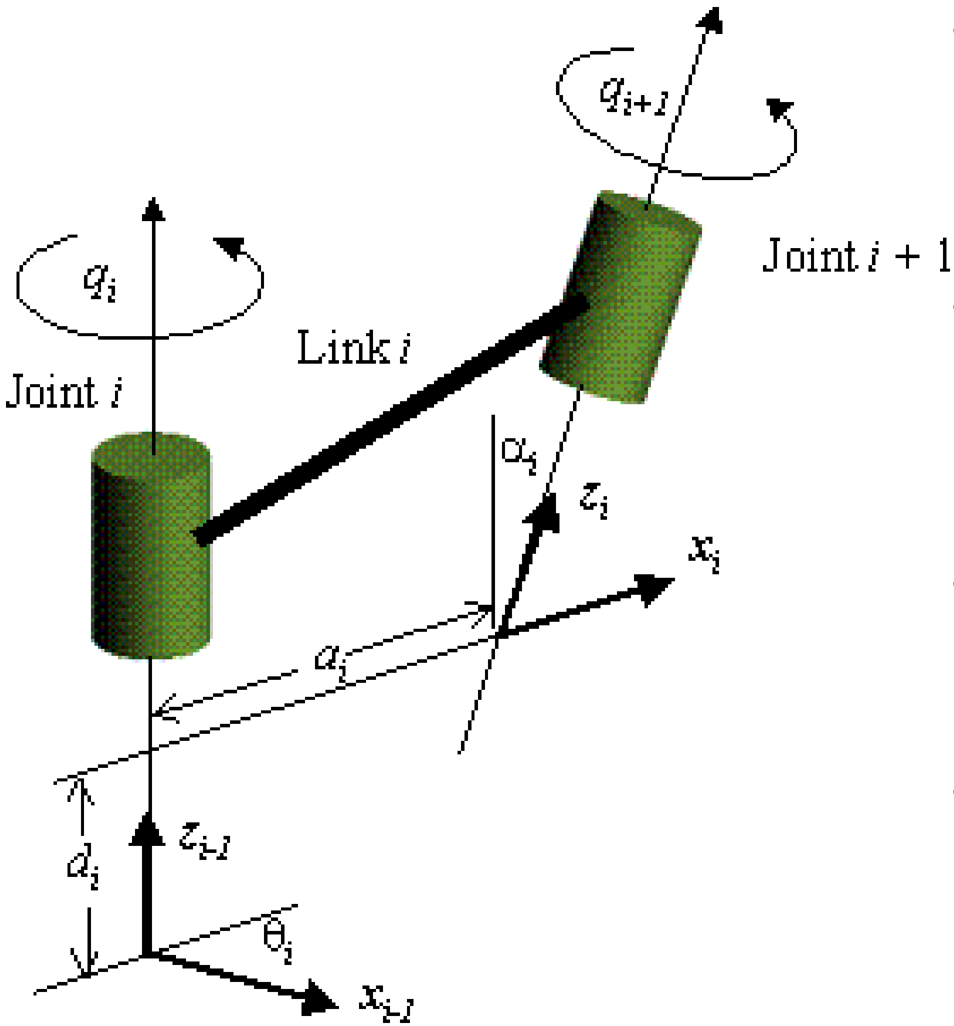


Link coordinate frames and their geometric parameters



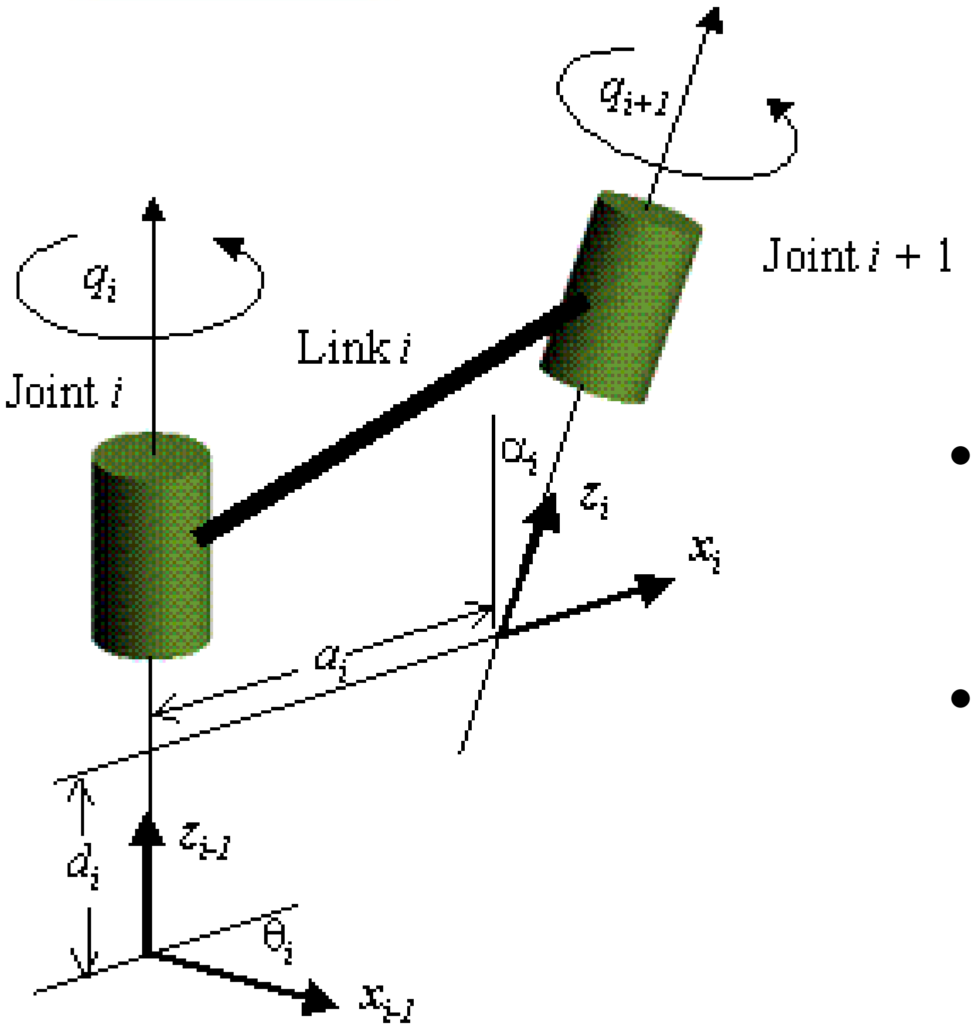
- The joint rotation axis is defined at the connection between the 2 links that the joint connects.
- For each axes, 2 normal lines are defined, one for each link.
- 4 parameters are associated to each link: 2 describe the asjacent links relative position (joint parameters) and 2 describe the link structure.

Link coordinate frames and their geometric parameters



- From the kinematics viewpoint, a link keeps a fixed configuration between 2 joints (link structure).
- The structure of link i can be characterized through the length and the angle of the rotation axis of joint i.
- a_i = minimum distance along the common normal line between the two joint axes
- α_i = angle between the two joint axes on a plane normal to a_i

Link coordinate frames and their geometric parameters



- the position of the i-th link with respect to the (i-1)-th link can be expressed by measuring the distance and the angle between 2 adjcent links
- d_i = distance between normal lines, along the i-th joint axis
- θ_i = angle between two normal lines, on a plane normal to the axis

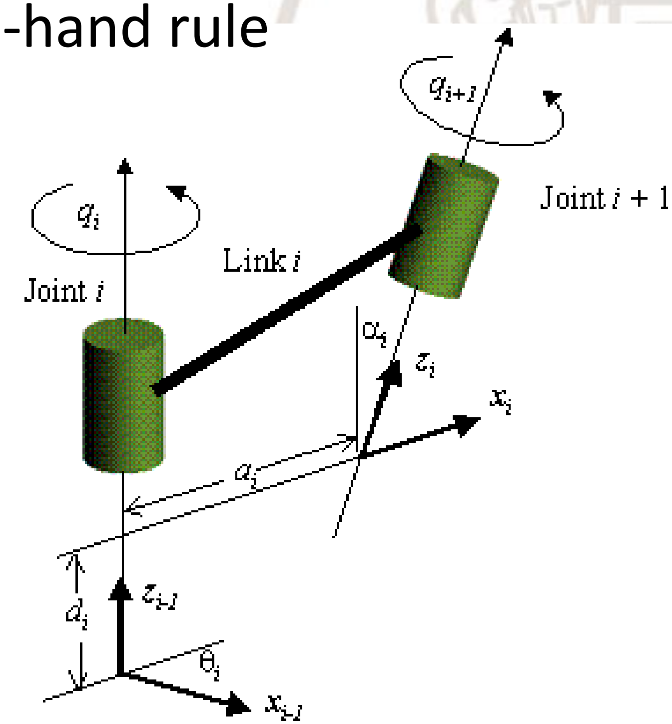
Denavit-Hartenberg (D-H) representation

THE BIOROBOTICS INSTITUTE

Scuola Superiore Sant'Anna

For a 6-DOF arm = 7 coordinate frames
z_{i-1} axis = motion axis of joint i
z_i axis = motion axis of joint i+1
x_i axis = normal to z_{i-1} axis and z_i axis
y_i axis = completes the frame with the right-hand rule

The end-effector position expressed in the end-effector frame can be expresses in the base frame, through a sequence of transformations.



Denavit-Hartenberg (D-H) representation

Algorithm:

1. Fix a base coordinate frame (0)
2. For each joint (1 a 5, for a 6-DOF robot), set:
   the joint axis,
   the origin of the coordinate frame,
   the x axis,
   the y axis.
3. Fix the end-effector coordinate frame.
4. For each joint and for each link, set:
   the joint parameters
   the link parameters.

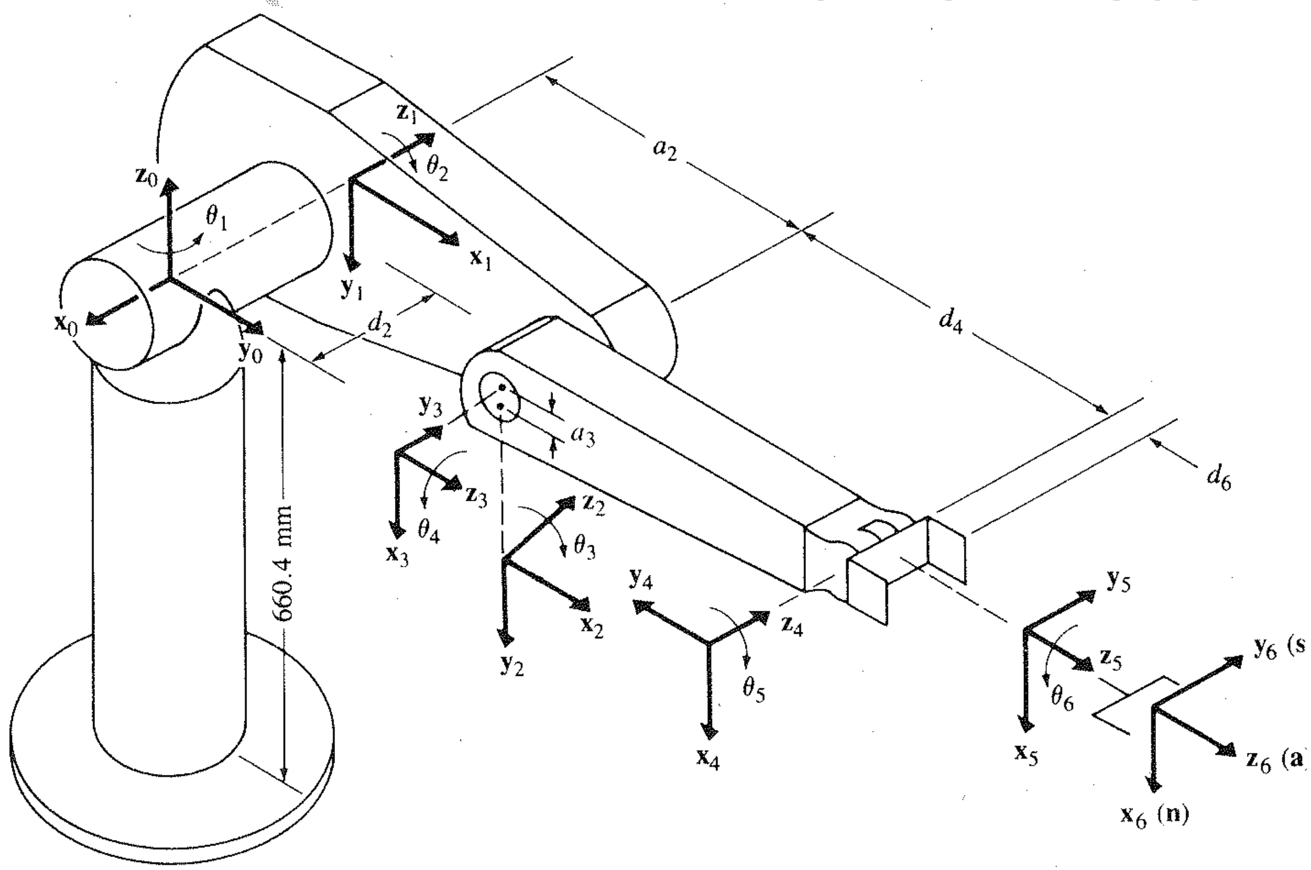
Denavit-Hartenberg (D-H) representation

THE BIOROBOTICS INSTITUTE

Scuola Superiore Sant'Anna

D-H for PUMA 560

THE BIOROBOTICS INSTITUTE

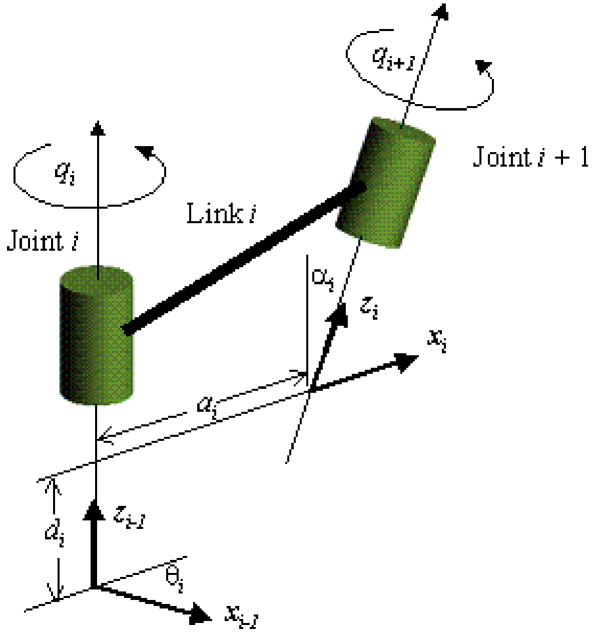


Parametri delle coordinate dei link per il braccio PUMA					
Giunto i	θ_i	α_i	a_i	d_i	Escursione del giunto
1	90	−90	0	0	−160 to +160
2	0	0	431.8 mm	149.09 mm	−225 to 45
3	90	90	−20.32 mm	0	−45 to 225
4	0	−90	0	433.07 mm	−110 to 170
5	0	90	0	0	−100 to 100
6	0	0	0	56.25 mm	−266 to 266

Denavit-Hartenberg (D-H) representation

- Once fixed the coordinate frames for each link, a homogenous transformation matrix can be built, describing the relations between adjacent frames.

- The matrix is built through rotations and translations:
  - Rotate around x_i for an angle α_i, in order to allineate the z axes
  - Translate of a_i along x_i
  - Translate of d_i along z_{i-1} in order to overlap the 2 origins
  - Rotate around z_{i-1} for an angle θ_i, in order to allineate the x axes



Denavit-Hartenberg (D-H) representation

- The D-H transformation can be expressed with a homogeneous transformation matrix:

$$^{i-1}A_i = T_{z,\theta}\, T_{z,d}\, T_{x,a}\, T_{x,\alpha}$$

$$r_{i-1} = {}^{i-1}A_i\, p_i = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \cos\theta_i & -a_i \sin\alpha_i \\ 0 & \sin\alpha_i & \cos\alpha_i & -d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Denavit-Hartenberg (D-H) representation

The D-H representation only depends on the 4 parameters associated to each link, which completely describe all joints, either revolute or prismatic.

For a **revolute joint**, d_i , a_i , α_i are the joint parameters, constant for a given robot. **Only θ_i varies.**

For a **giunto prismatic**, θ_i , a_i , α_i are the joint parameters, constant for a given robot. **Only d_i varies**

Denavit-Hartenberg (D-H) representation

The homogeneous matrix T describing the n-th frame with respect to the base frame is the product of the sequence of transformation matrices ${}^{i-1}A_i$, expressed as:

$${}^{0}T_n = {}^{0}A_1 \, {}^{1}A_2 \, \ldots\ldots.. \, {}^{n-1}A_n$$

$${}^{0}T_n = \begin{bmatrix} X_i & Y_i & Z_i & p_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad {}^{0}T_n = \begin{bmatrix} {}^{0}R_n & {}^{0}p_n \\ 0 & 1 \end{bmatrix}$$

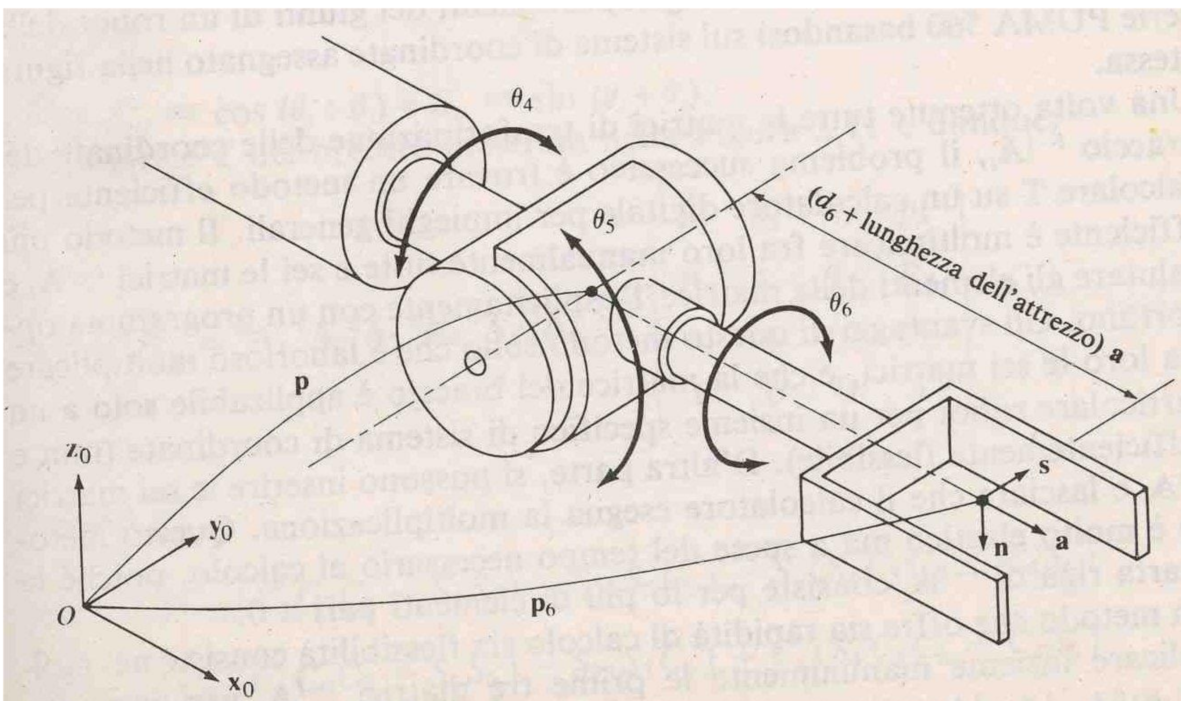
where [X_i Y_i Z_i] is the matrix describing the orientation of the n-th frame with respect to the base frame

P_i is the position vector pointing from the origin of the base frame to the origin of the n-th frame

R is the matrix describing the roll, pitch and yaw angles

Denavit-Hartenberg (D-H) representation

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



Denavit-Hartenberg (D-H) representation

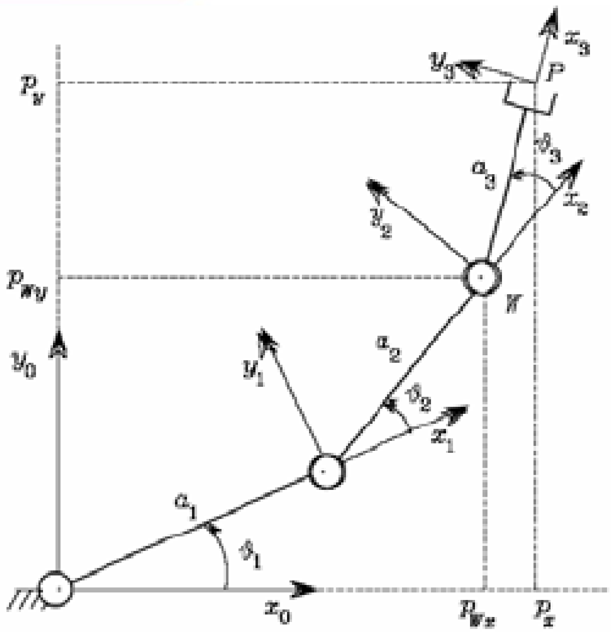
The direct kinematics of a 6-link manipulator can be solved by calculating $T = {}^0A_6$ by multiplying the 6 matrices

For revolute-joints manipulators, the parameters to set for finding the end-effector final position in the Cartesian space are the joint angles $\theta_i = q_i$

For a given $q = (q_0, q_1, q_2, q_{3,} q_{4,} q_5)$ it is possible to find (x,y,z,roll, pitch, yaw)

$$\mathbf{x} = K(q) = T(q)$$

Planar 3-link manipulator



	a_i	α_i	d_i	ϑ_i
1	a_1	0	0	ϑ_1
2	a_2	0	0	ϑ_2
3	a_3	0	0	ϑ_3

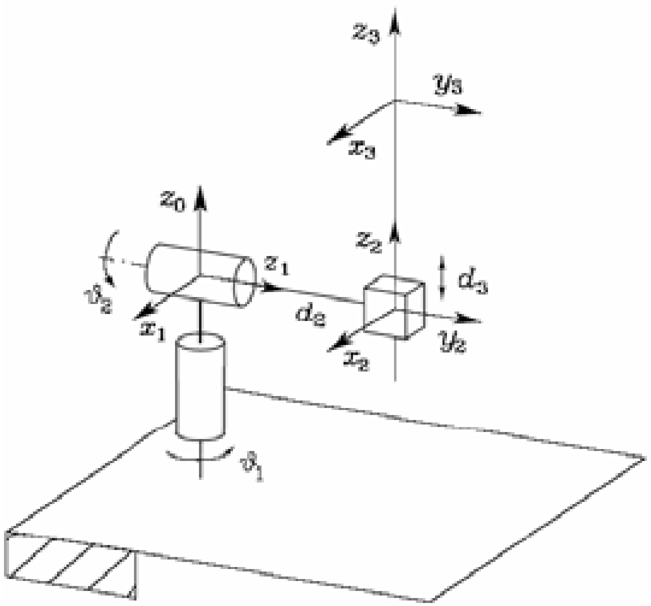
$$\boldsymbol{T}_3^0 = \boldsymbol{A}_1^0 \boldsymbol{A}_2^1 \boldsymbol{A}_3^2 = \begin{bmatrix} c_{123} & -s_{123} & 0 & a_1c_1 + a_2c_{12} + a_3c_{123} \\ s_{123} & c_{123} & 0 & a_1s_1 + a_2s_{12} + a_3s_{123} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La terna utensile non coincide con la terna 3

Spherical manipulator

THE BIOROBOTICS INSTITUTE

Scuola Superiore Sant'Anna



	a_i	α_i	d_i	ϑ_i
1	0	$-\pi/2$	0	ϑ_1
2	0	$\pi/2$	d_2	ϑ_2
3	0	0	d_3	0

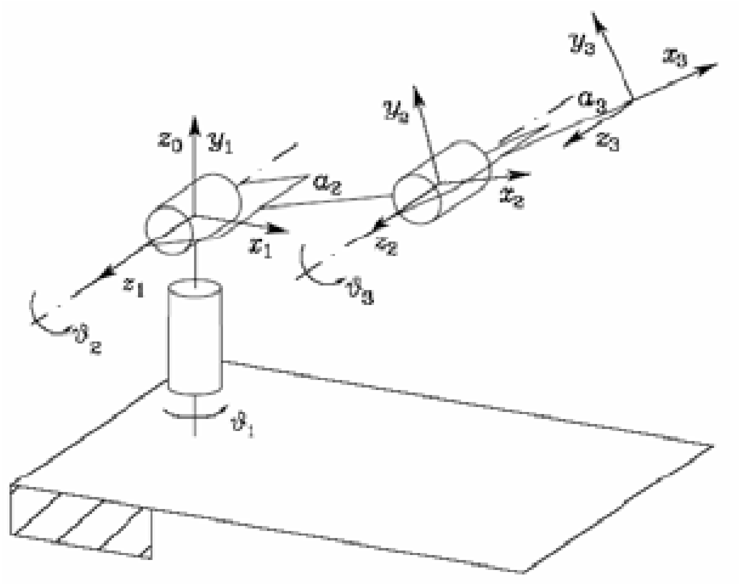
$$T_3^0 = A_1^0 A_2^1 A_3^2 = \begin{bmatrix} c_1 c_2 & -s_1 & c_1 s_2 & c_1 s_2 d_3 - s_1 d_2 \\ s_1 c_2 & c_1 & s_1 s_2 & s_1 s_2 d_3 + c_1 d_2 \\ -s_2 & 0 & c_2 & c_2 d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La terna utensile coincide con la terna 3

Anthropomorphic manipulator

THE BIOROBOTICS INSTITUTE

Scuola Superiore Sant'Anna



	a_i	α_i	d_i	ϑ_i
1	0	$\pi/2$	0	ϑ_1
2	a_2	0	0	ϑ_2
3	a_3	0	0	ϑ_3

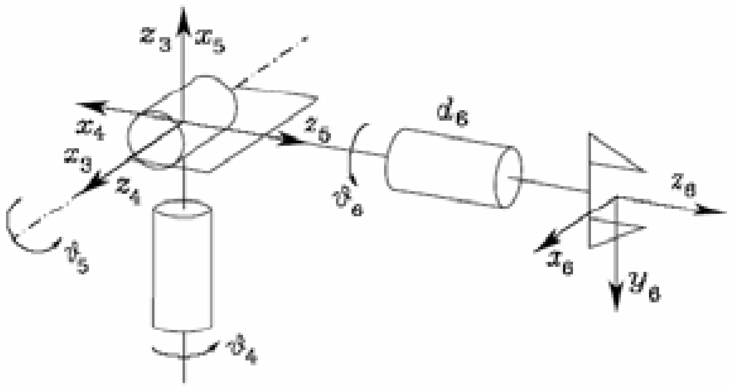
$$\boldsymbol{T}_3^0 = \boldsymbol{A}_1^0 \boldsymbol{A}_2^1 \boldsymbol{A}_3^2 = \begin{bmatrix} c_1 c_{23} & -c_1 s_{23} & s_1 & c_1(a_2 c_2 + a_3 c_{23}) \\ s_1 c_{23} & -s_1 s_{23} & -c_1 & s_1(a_2 c_2 + a_3 c_{23}) \\ s_{23} & c_{23} & 0 & a_2 s_2 + a_3 s_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La terna utensile non coincide con la terna 3

Spherical wrist

THE BIOROBOTICS INSTITUTE

Scuola Superiore Sant'Anna



	a_i	α_i	d_i	ϑ_i
4	0	$-\pi/2$	0	ϑ_4
5	0	$\pi/2$	0	ϑ_5
6	0	0	d_6	ϑ_6

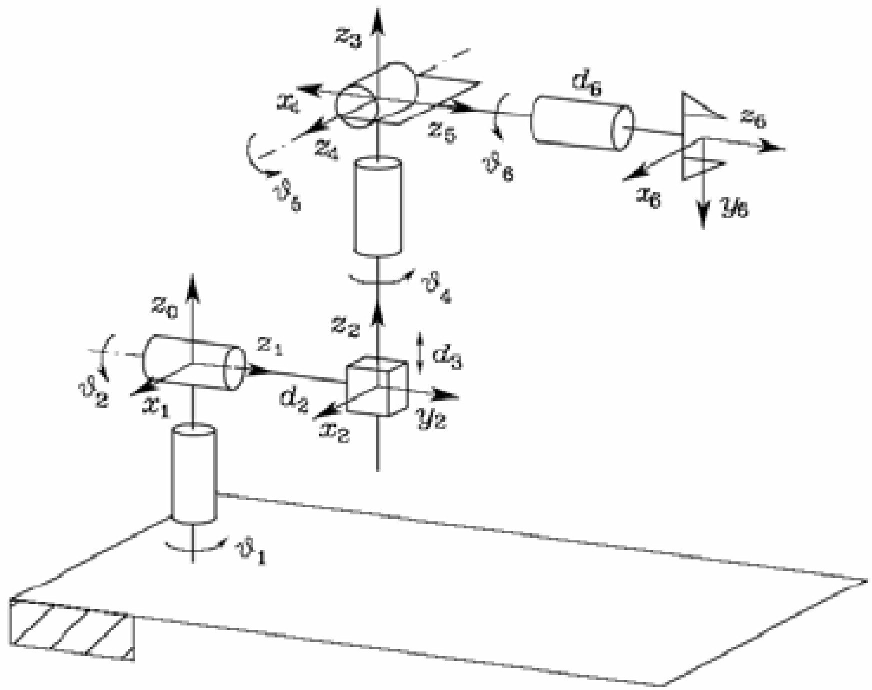
ϑ_4, ϑ_5, ϑ_6 sono gli angoli di Eulero ZYZ della terna 6 rispetto alla 3

$$\boldsymbol{T}_6^3 = \boldsymbol{A}_4^3 \boldsymbol{A}_5^4 \boldsymbol{A}_6^5 = \begin{bmatrix} c_4c_5c_6 - s_4s_6 & -c_4c_5s_6 - s_4c_6 & c_4s_5 & c_4s_5d_6 \\ s_4c_5c_6 + c_4s_6 & -s_4c_5s_6 + c_4c_6 & s_4s_5 & s_4s_5d_6 \\ -s_5c_6 & s_5s_6 & c_5 & c_5d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La terna utensile coincide con la terna 6

Stanford manipulator

Il manipolatore di Stanford è un manipolatore sferico con polso sferico



	a_i	α_i	d_i	ϑ_i
1	0	$-\pi/2$	0	ϑ_1
2	0	$\pi/2$	d_2	ϑ_2
3	0	0	d_3	0
4	0	$-\pi/2$	0	ϑ_4
5	0	$\pi/2$	0	ϑ_5
6	0	0	d_6	ϑ_6

$$T_6^0 = T_3^0 T_6^3$$

Già calcolata per il polso sferico

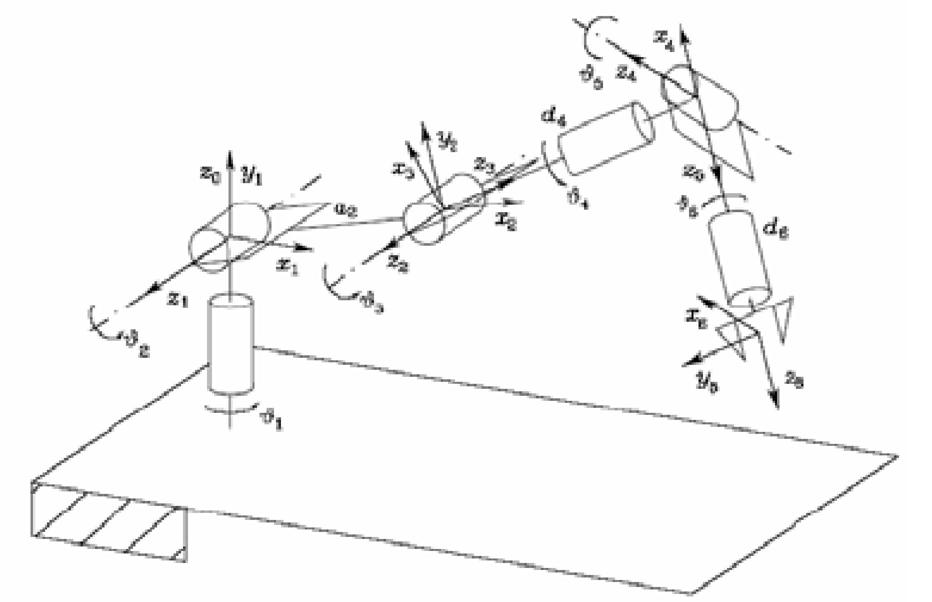
Già calcolata per il manipolatore sferico

Anthropomorphic manipulator with spherical wrist

THE BIOROBOTICS INSTITUTE

Scuola Superiore Sant'Anna

Montiamo un polso sferico sul manipolatore antropomorfo



	a_i	α_i	d_i	ϑ_i
1	0	$\pi/2$	0	ϑ_1
2	a_2	0	0	ϑ_2
3	0	$\pi/2$	0	ϑ_3
4	0	$-\pi/2$	d_4	ϑ_4
5	0	$\pi/2$	0	ϑ_5
6	0	0	d_6	ϑ_6

La terna 3 del manipolatore antropomorfo non era orientata correttamente per il successivo polso sferico, per cui per calcolare la cinematica diretta occorre rifare i conti (non basta semplicemente moltiplicare le due matrici di trasformazione parziali)

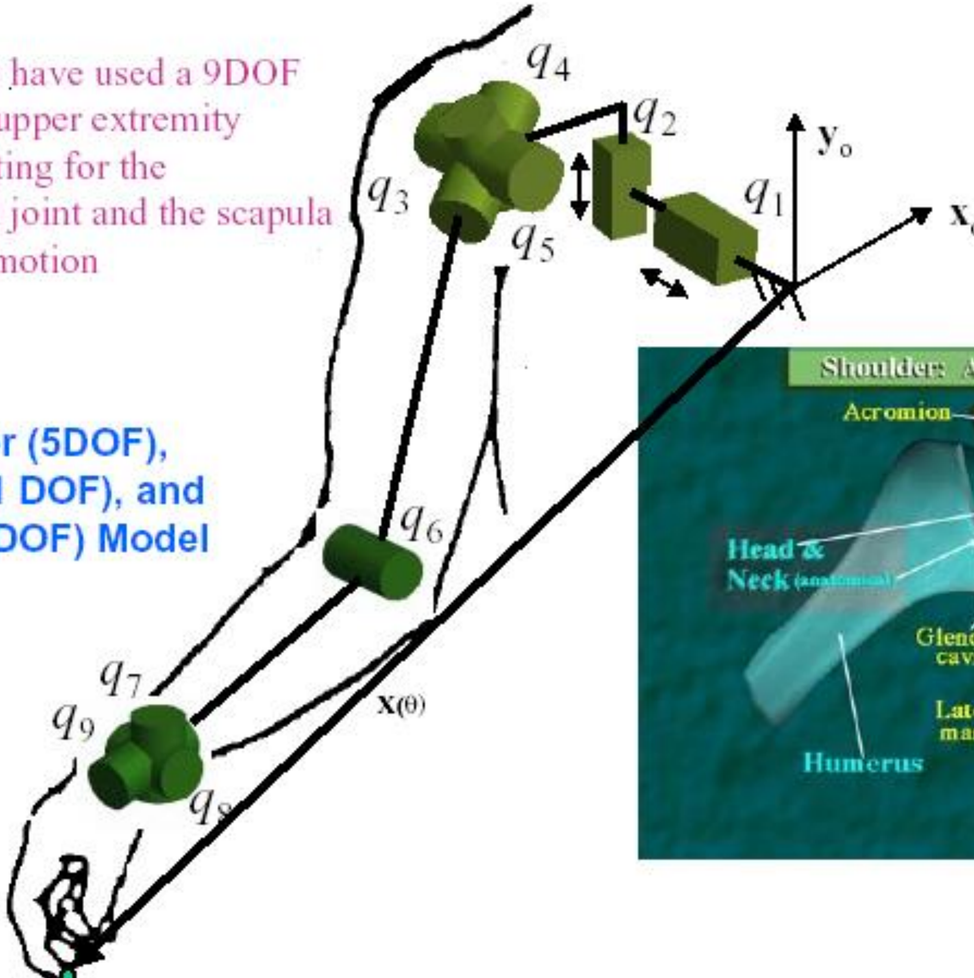
Kinematic model of the human arm

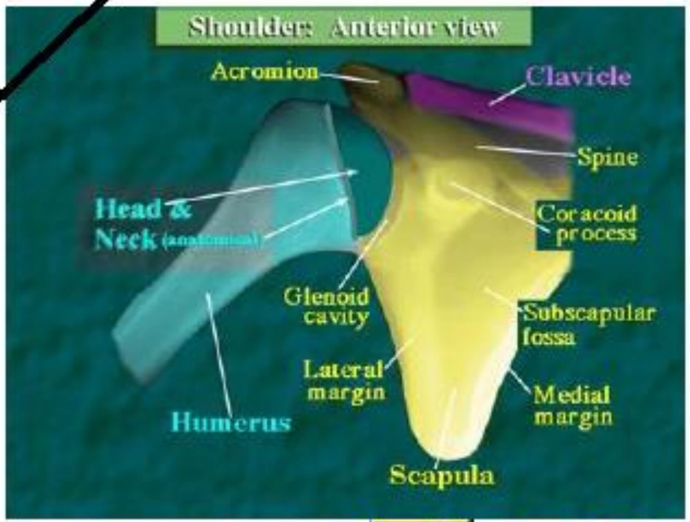
THE BIOROBOTICS INSTITUTE

Scuola Superiore Sant'Anna

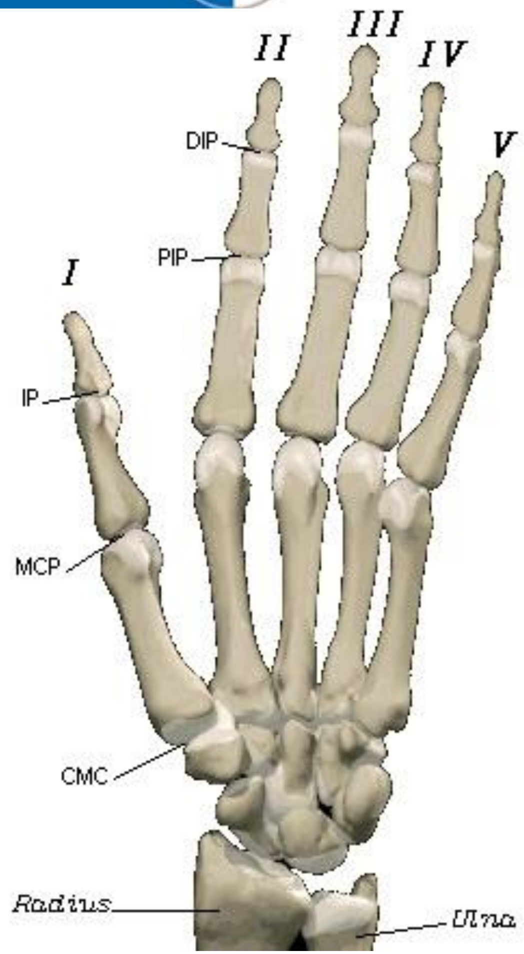
Typically, we have used a 9DOF model of the upper extremity while accounting for the glenohumeral joint and the scapula translational motion

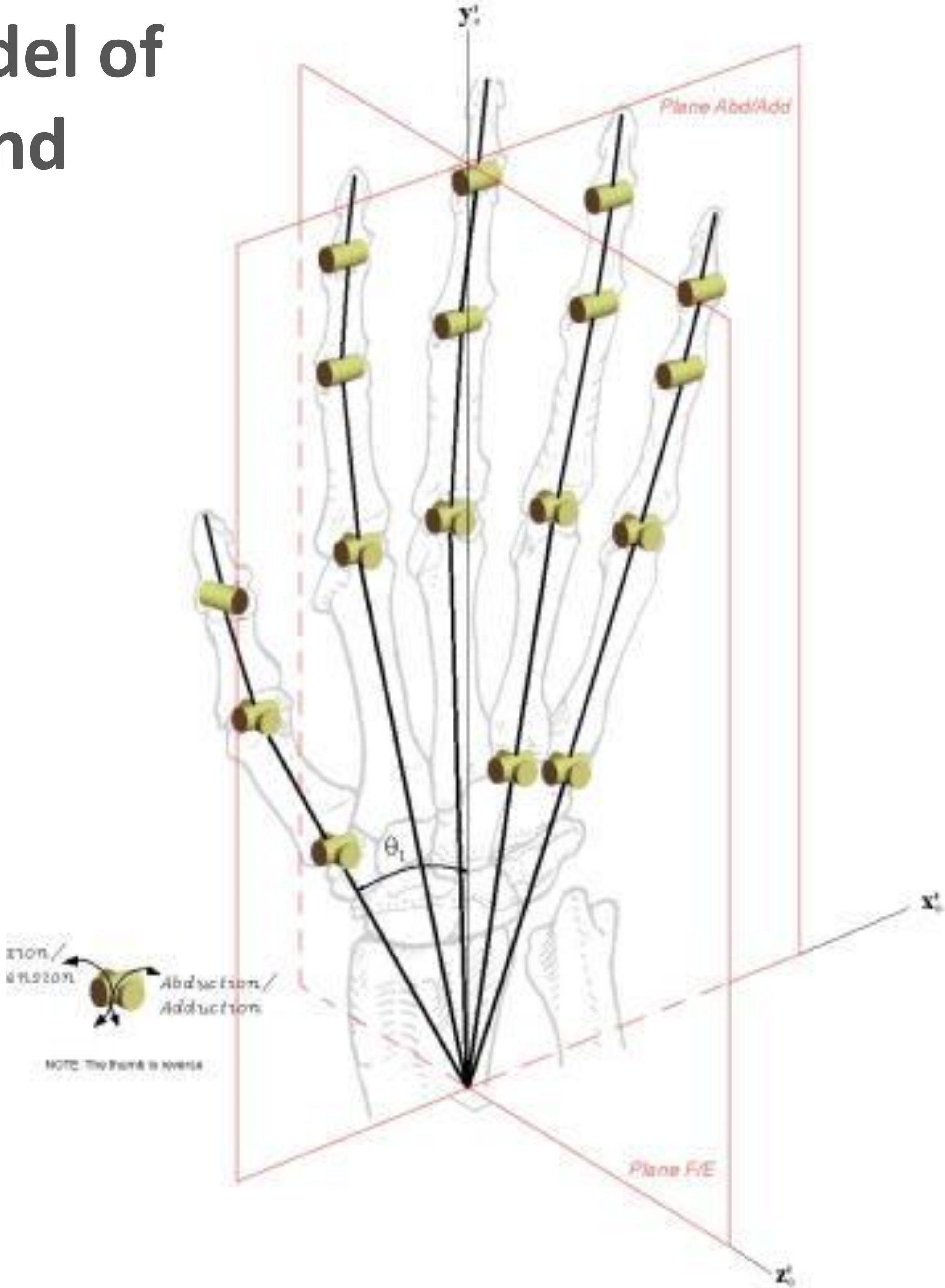
Shoulder (5DOF), Elbow (1 DOF), and Wrist (3DOF) Model





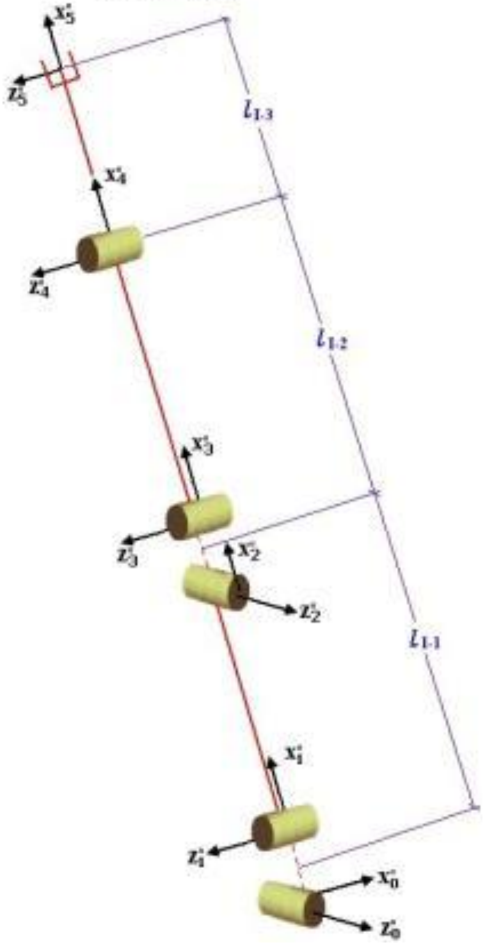
Kinematic model of the human hand





Kinematic model of the human thumb

(I) Thumb



	θ_i	d_i	a_i	α_i
1	$q_{50} + \frac{\pi}{2}$	0	0	$\frac{-\pi}{2}$
2	q_{51}	0	l_{I-1}	$\frac{\pi}{2}$
3	q_{52}	0	0	$\frac{-\pi}{2}$
4	q_{53}	0	l_{I-2}	0
5	q_{54}	0	l_{I-3}	0

	Min.	Max.
q_{50}	0	$\frac{\pi}{3}$
q_{51}	$\frac{-5}{36}\pi$	$\frac{7}{36}\pi$
q_{52}	0	$\frac{\pi}{3}$
q_{53}	$\frac{-\pi}{18}$	$\frac{11}{36}\pi$
q_{54}	$\frac{-\pi}{12}$	$\frac{4}{9}\pi$

Kinematic model of the human body

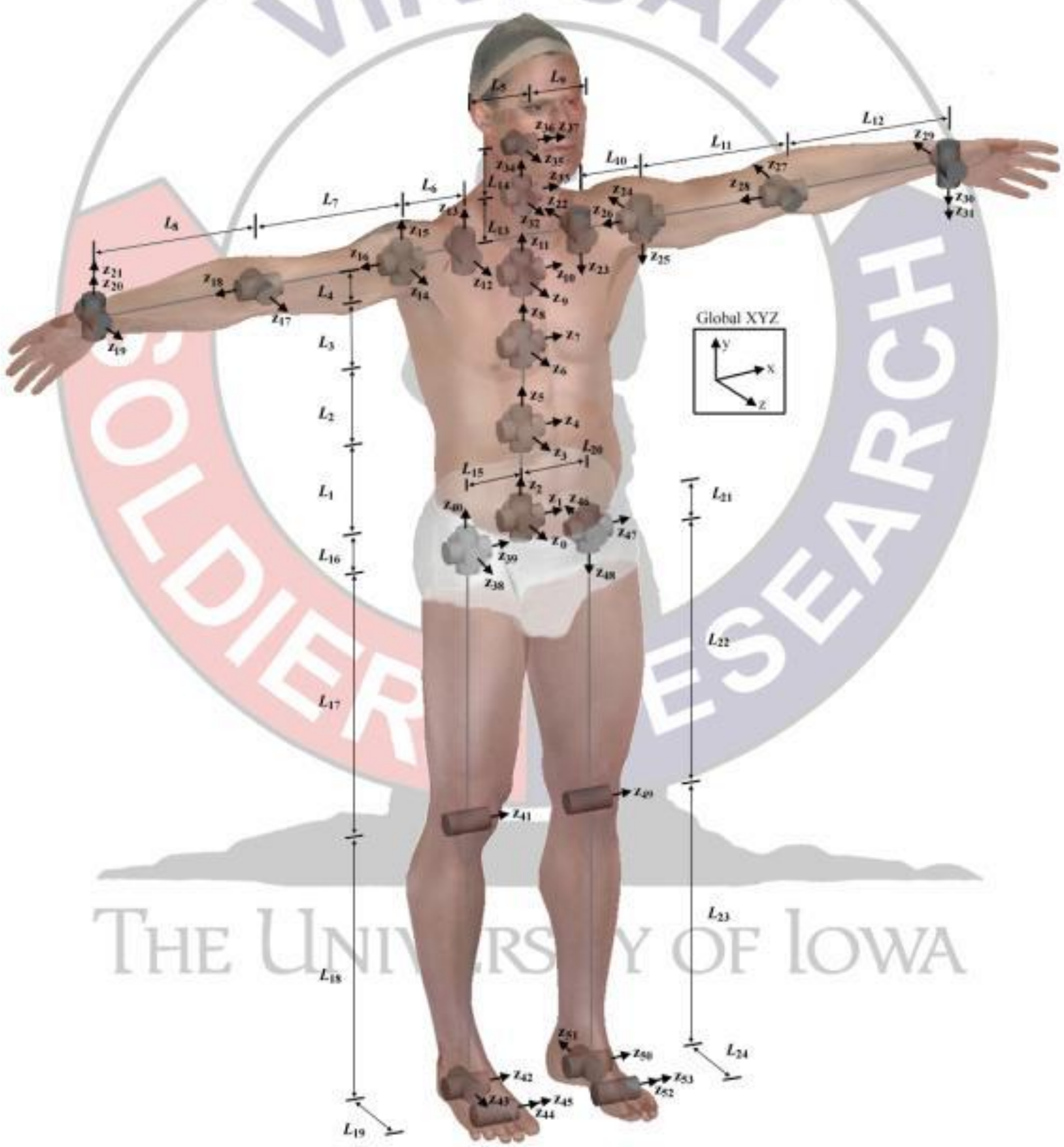


Table 1 DH table for arms, legs, and neck

#	DOF	θ_i	d_i	α_i	a_i
1	Q1	90	0	90	0
2	Q2	90	0	90	0
3	Q3	90	L1	90	0
4	Q4	90	0	90	0
5	Q5	90	0	90	0
6	Q6	90	L2	90	0
7	Q7	90	0	90	0
8	Q8	90	0	90	0
9	Q9	90	L3	90	0
10	Q10	90	0	90	0
11	Q11	90	0	90	0
12	Q12	-90	L4	-90	L5
13	Q13	0	0	90	0
14	Q14	0	0	-90	L6
15	Q15	0	0	90	0
16	Q16	90	0	90	0
17	Q17	90	L7	90	0
18	Q18	0	0	-90	0
19	Q19	0	L8	90	0
20	Q20	90	0	90	0
21	Q21	0	0	0	0