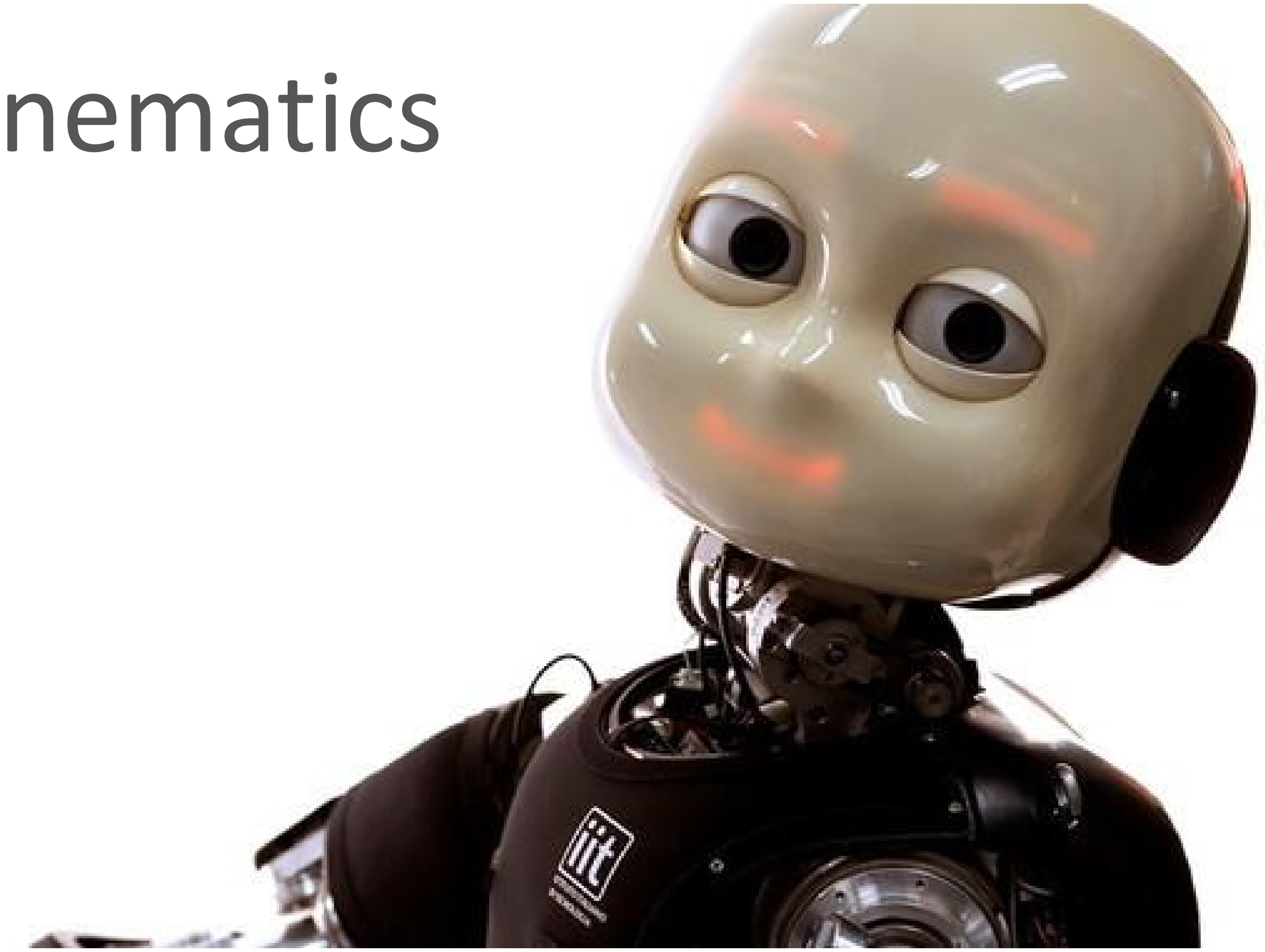
Master in Computer Science
University of Pisa
Robotics

THE BIOROBOTICS INSTITUTE

Sant'Anna
School of Advanced Studies – Pisa

Robot mechanics and kinematics



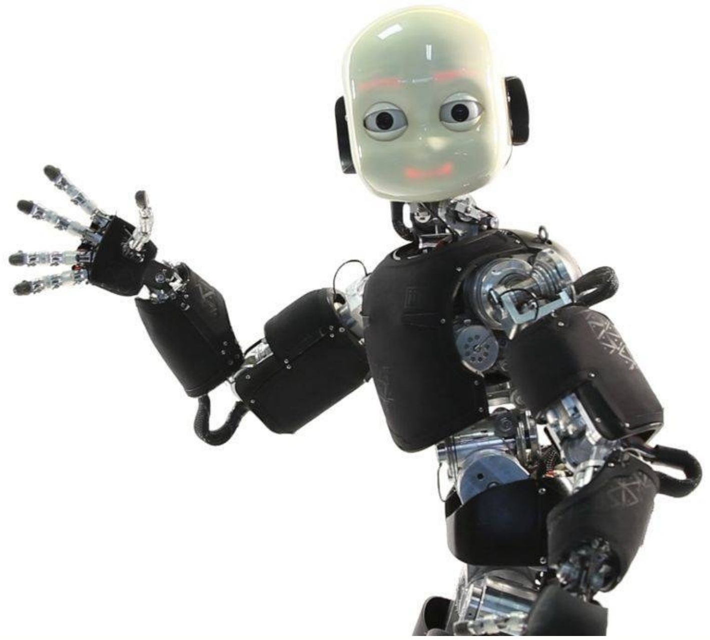
egidio.falotico@santannapisa.it

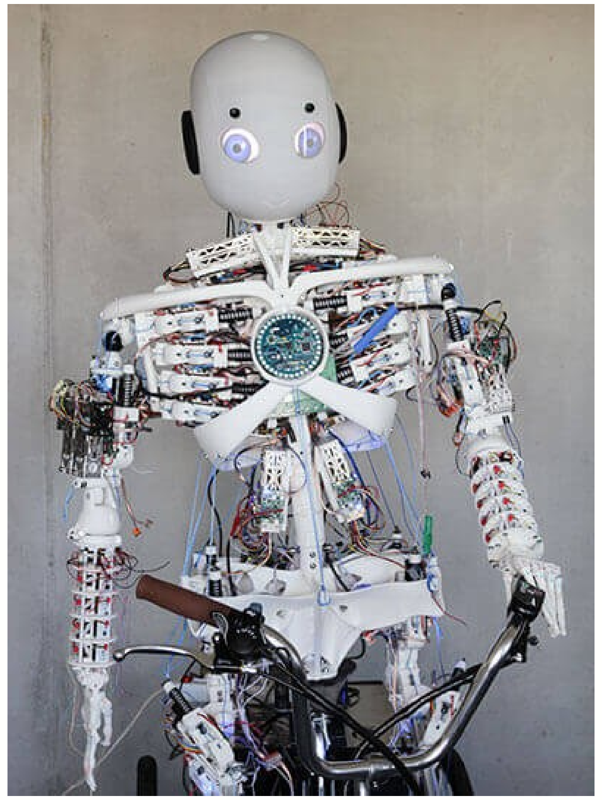
What is a robot?

What is a robot?

What robots do you know?

What robots do you know?





What is a robot?

What robots do you know?

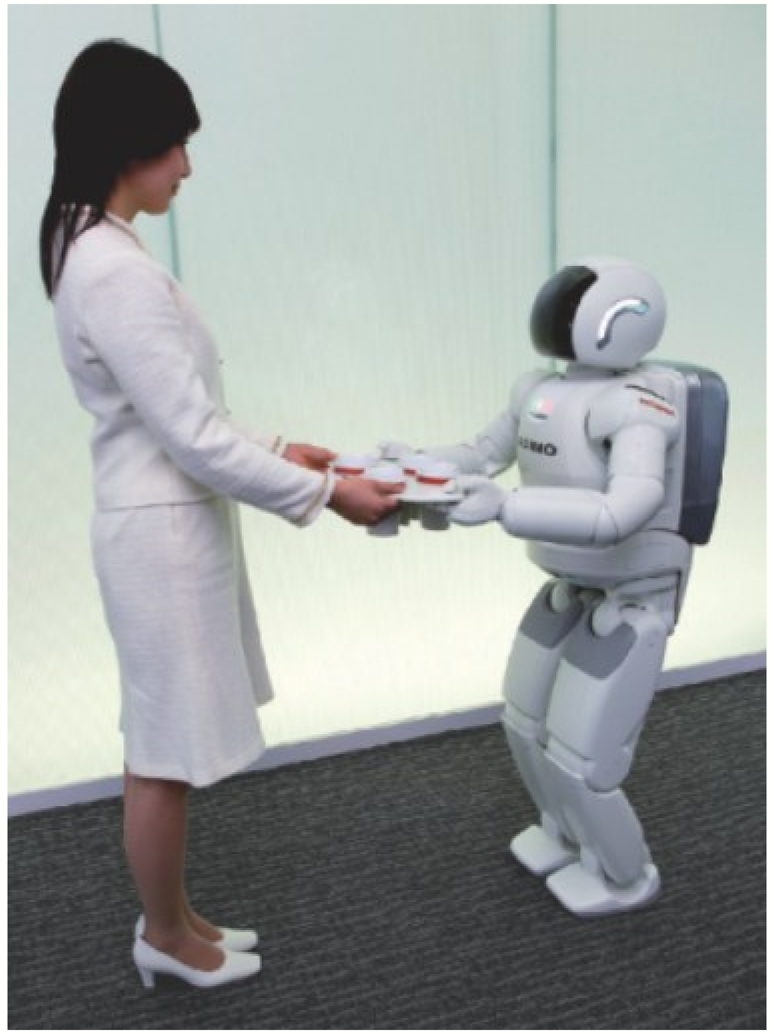
Is a robot a physical agent?

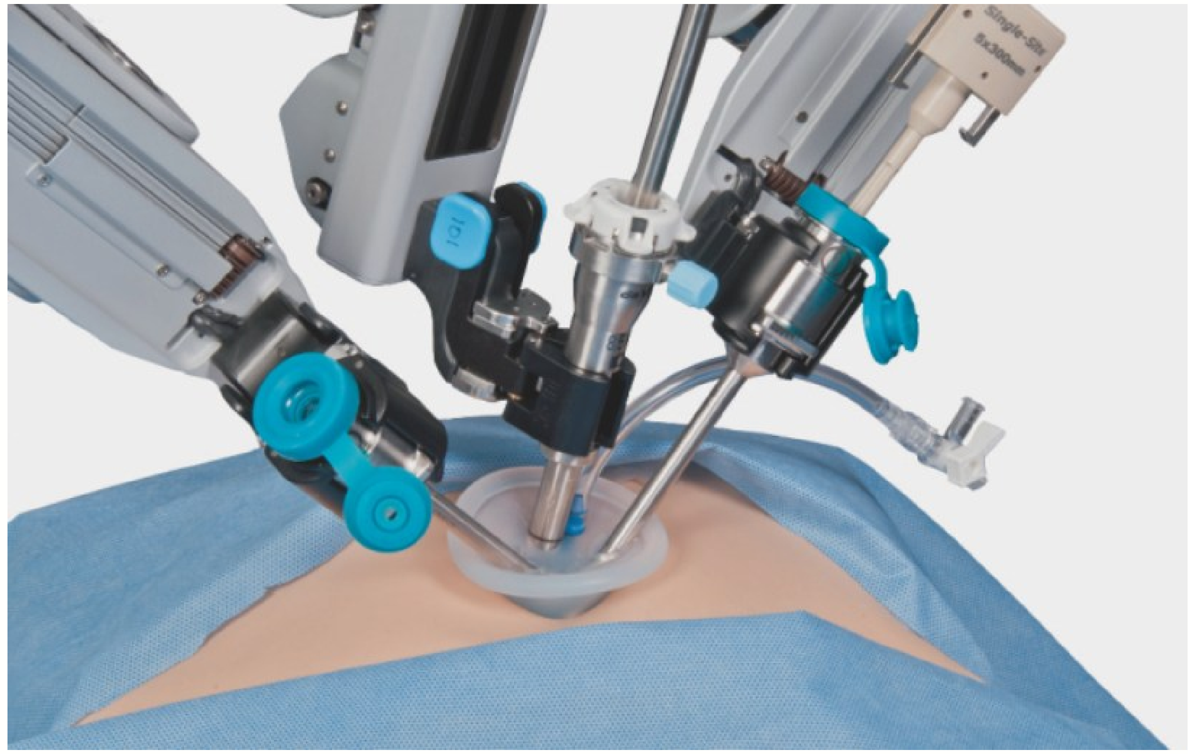
Can a robot act?

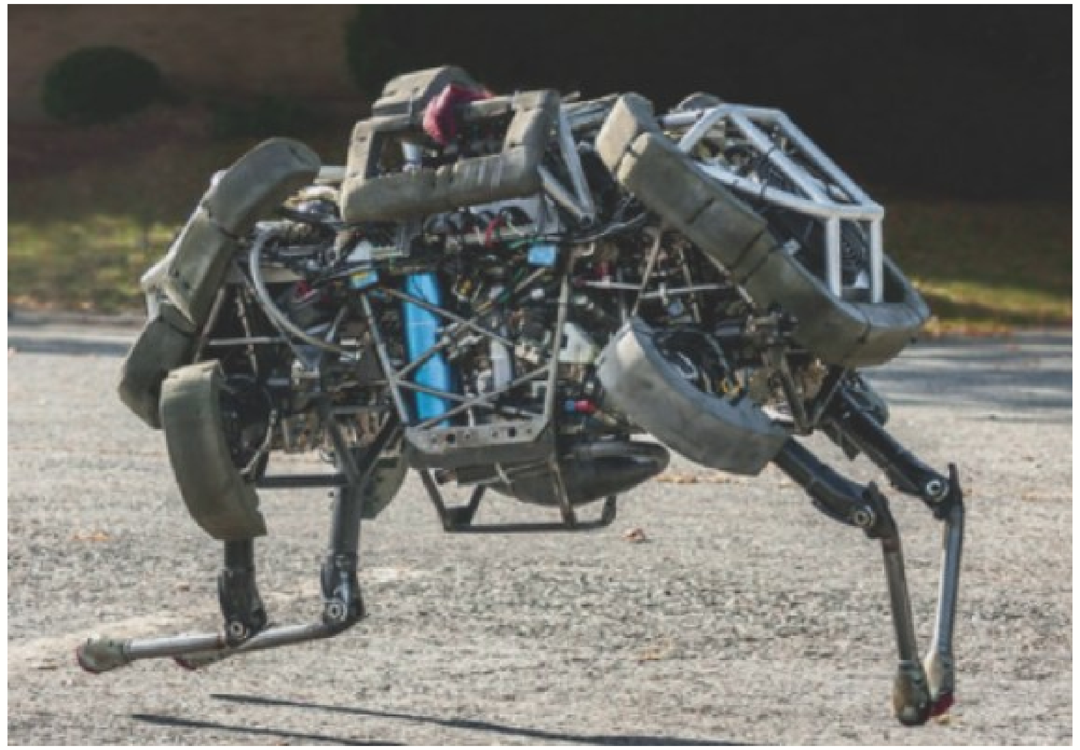
Can a robot sense?

Robot definition

A robot is an autonomous system which exists in the physical world, can sense its environment, and can act on it to achieve some goals *[Maja J Mataric, The Robotics Primer, The MIT Press, 2007]*







Is a robot autonomous?

Is a robot intelligent?

A robot is an AUTONOMOUS system

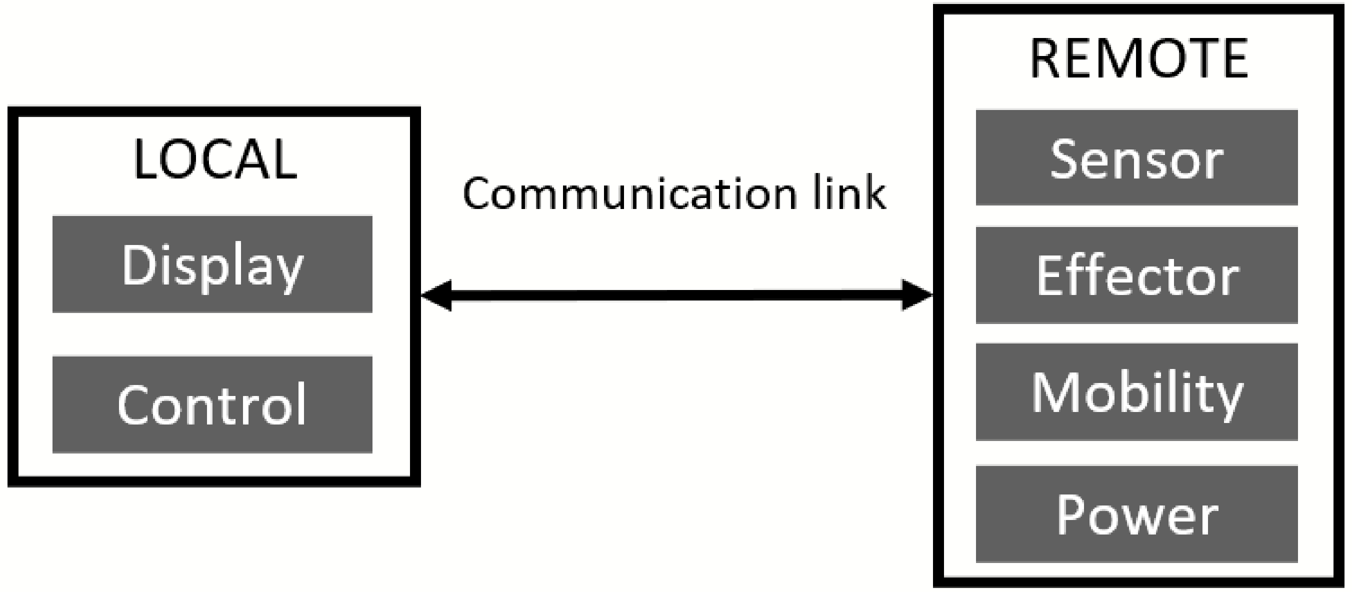
Autonomous

- An *autonomous* robot acts on the basis of its own decisions, and is not controlled by a human

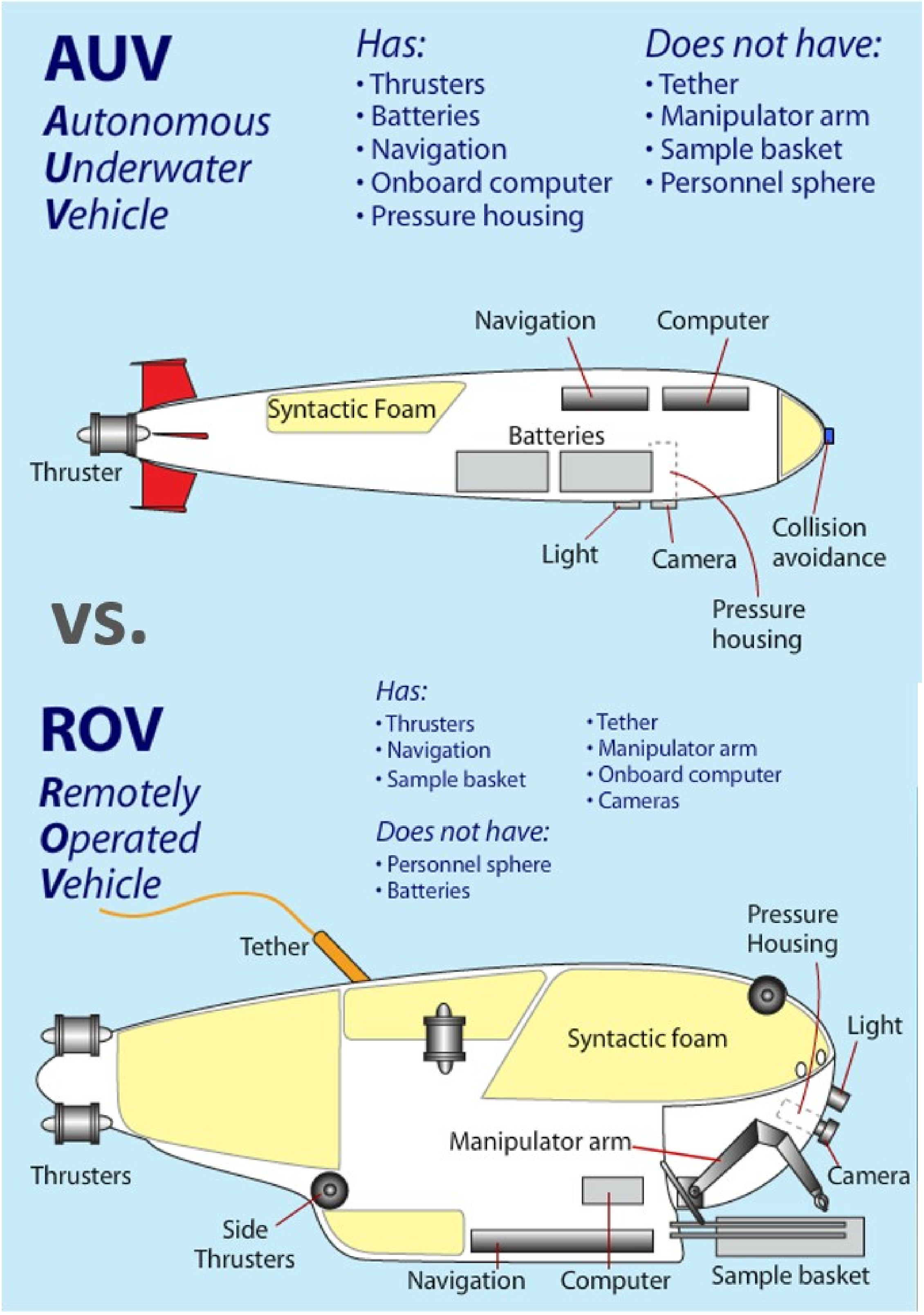
Non autonomous

- A non-autonomous robot is commanded step by step by an operator (teleoperation)
- Semi-autonomous
- Control is shared between robot and user; different levels of autonomy may exist

≠ from teleoperation



Robonaut

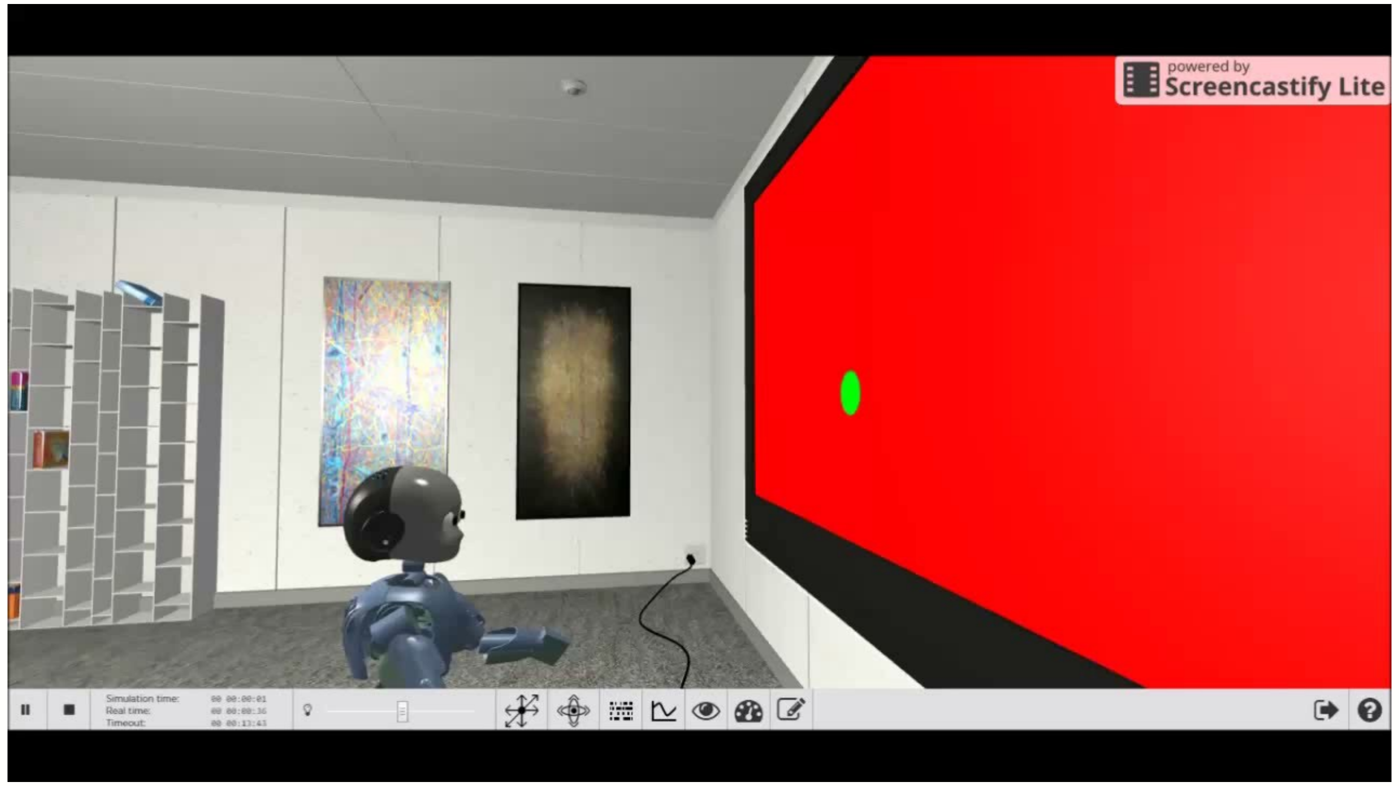


A robot is an autonomous system which exists in the PHYSICAL WORLD

Subject to the laws of physics

≠ from simulations

The physical world, the physical laws and the interactions are simulated and somehow approximated



A robot is an autonomous system which exists in the physical world, can SENSE its environment

- **the robot has *sensors*, some means of perceiving (e.g., hearing, touching, seeing, smelling, etc.) in order to get information from the world.**

A robot is an autonomous system which exists in the physical world, can sense its environment, and can ACT ON IT

- the robot has *sensors*, some means of perceiving (e.g., hearing, touching, seeing, smelling, etc.) in order to get information from the world.
- **the robot has *effectors* and *actuators*, for taking actions to respond to sensory inputs and to achieve what is desired**

A robot is an autonomous system which exists in the physical world, can sense its environment, and can act on it to ACHIEVE SOME GOALS.

- the robot has *sensors*, some means of perceiving (e.g., hearing, touching, seeing, smelling, etc.) in order to get information from the world.
- the robot has *effectors* and *actuators*, for taking actions to respond to sensory inputs and to achieve what is desired
- **Robot “intelligence”**

What’s in a robot?
Robot components

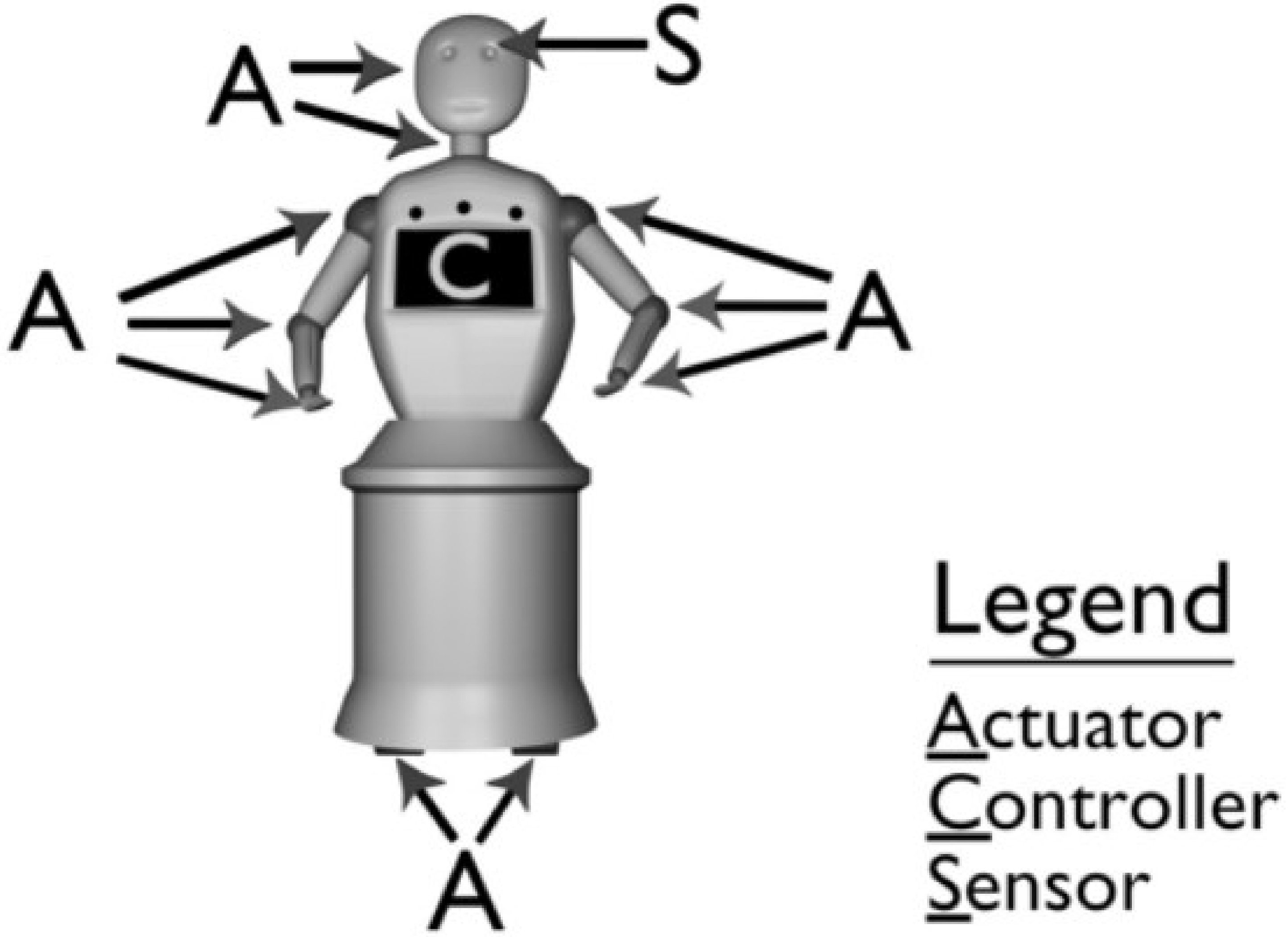
Robot’s main components are:

• A *physical body*, so it can exist and do work in the physical world

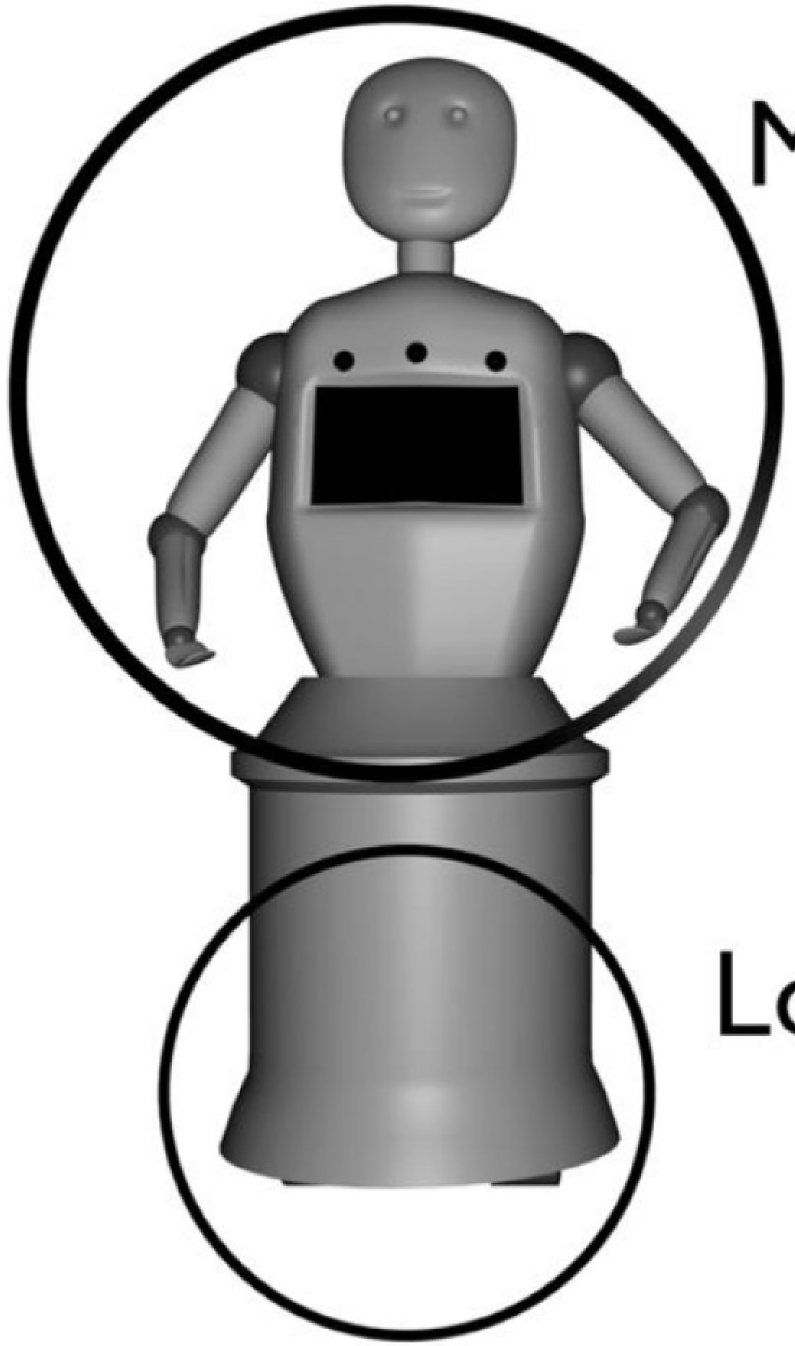
• *Sensors*, so it can sense/perceive its environment

• *Effectors* and *actuators*, so it can take actions

• A *controller*, so it can be autonomous.



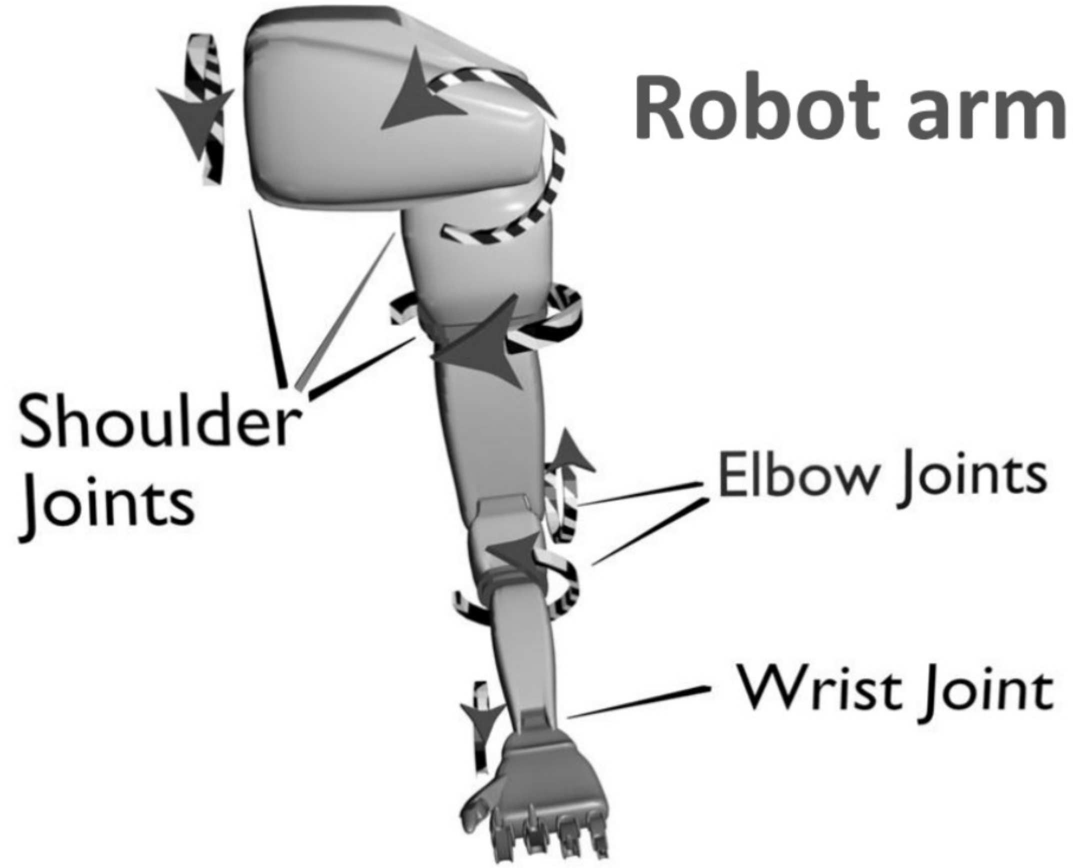
Effectors

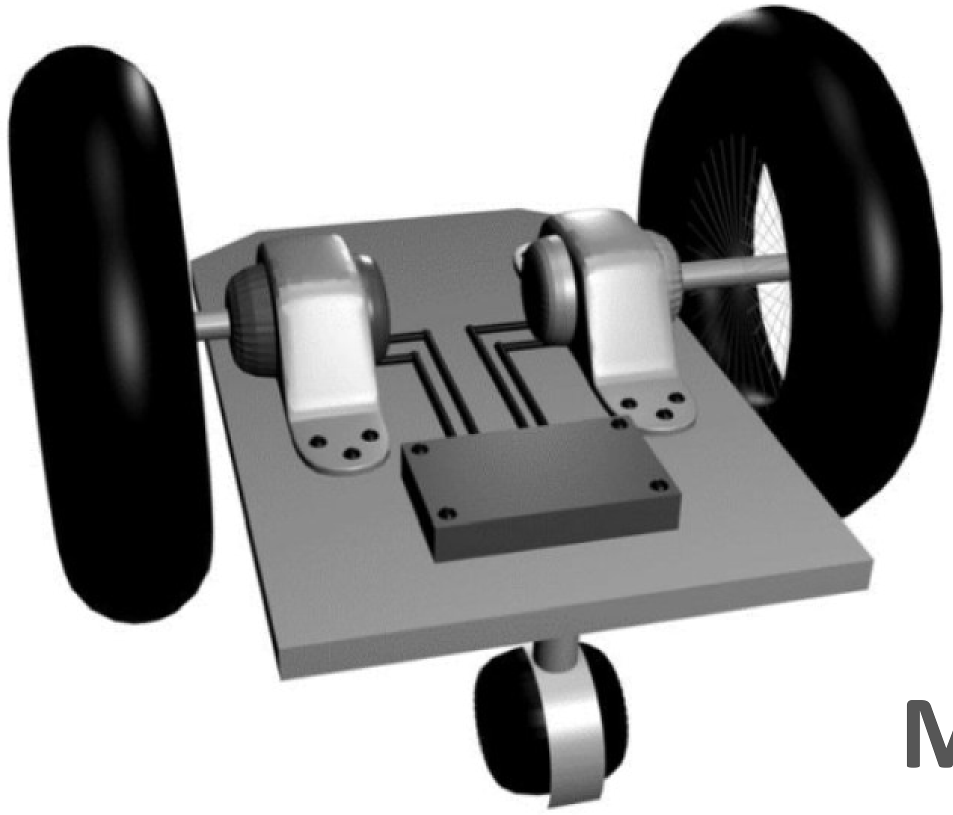


Manipulation

Locomotion

Robot arms





Mobile robot

Robot mechanics and kinematics

- **Introduction to robot mechanics**
  - Definition of degree of freedom (DOF)
  - Joint types
  - Definition of robot manipulator
  - Manipulator types

- **Definitions of joint space and Cartesian space**
  - Robot position in joint space
  - Robot position in Cartesian space
  - Definition of workspace

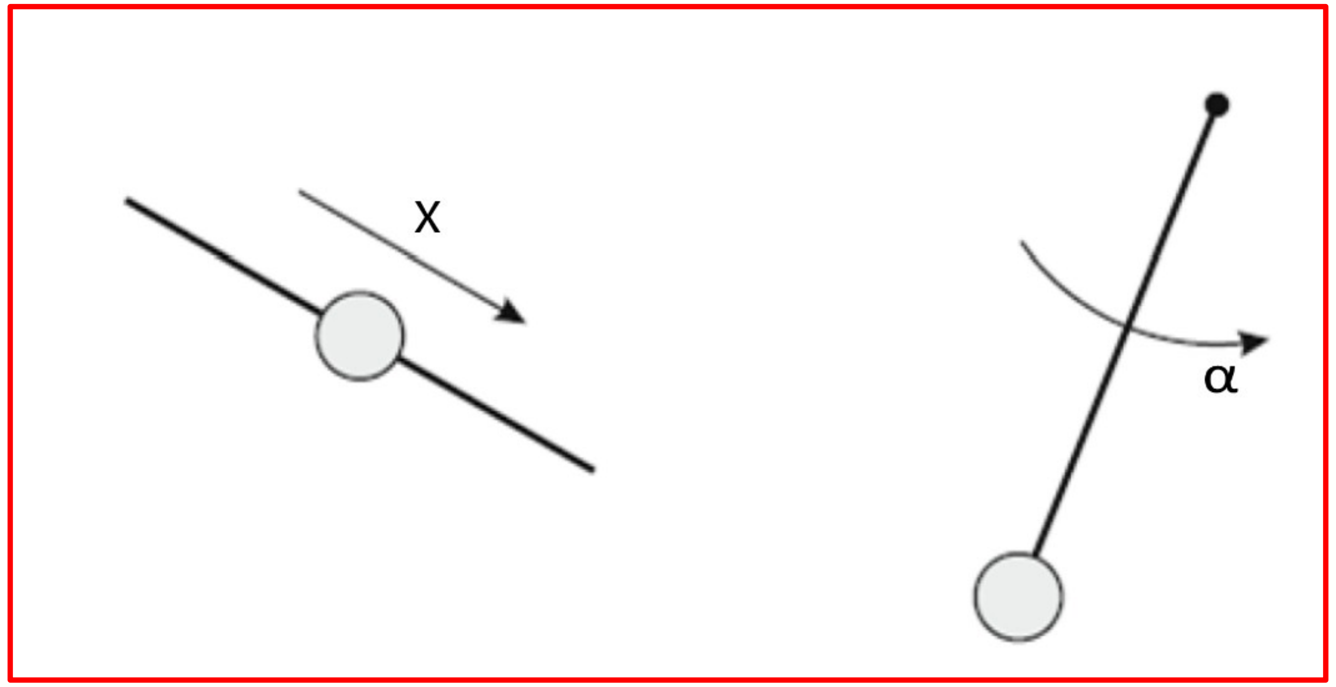
- **Direct and inverse kinematics**
  - Kinematics transformations
  - Concept of kinematic redundancy
  - Recall of transformation matrices

- **Denavit-Hartenberg representation**
  - Algorithm
  - Examples

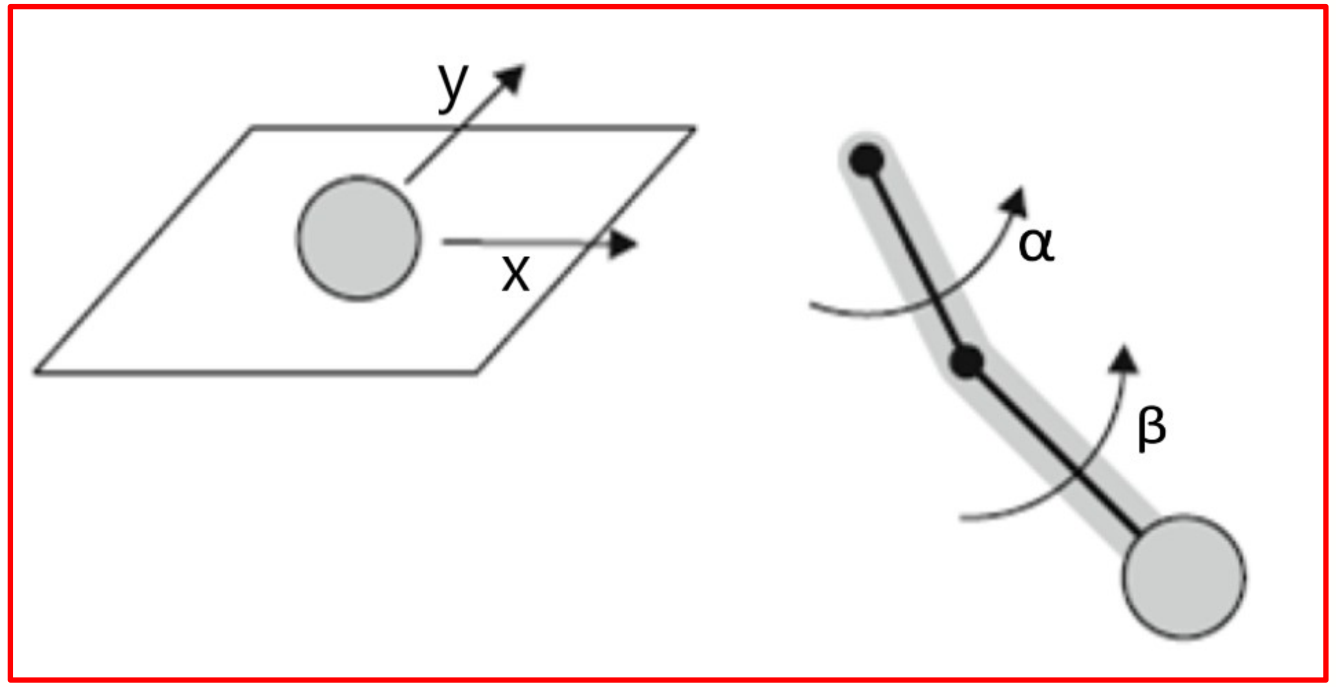
Degree of Freedom (DOF)

Definition The configuration of a robot is a complete specification of the position of every point of the robot. The minimum number n of real-valued coordinates needed to represent the configuration is the number of degrees of freedom (dof) of the robot.

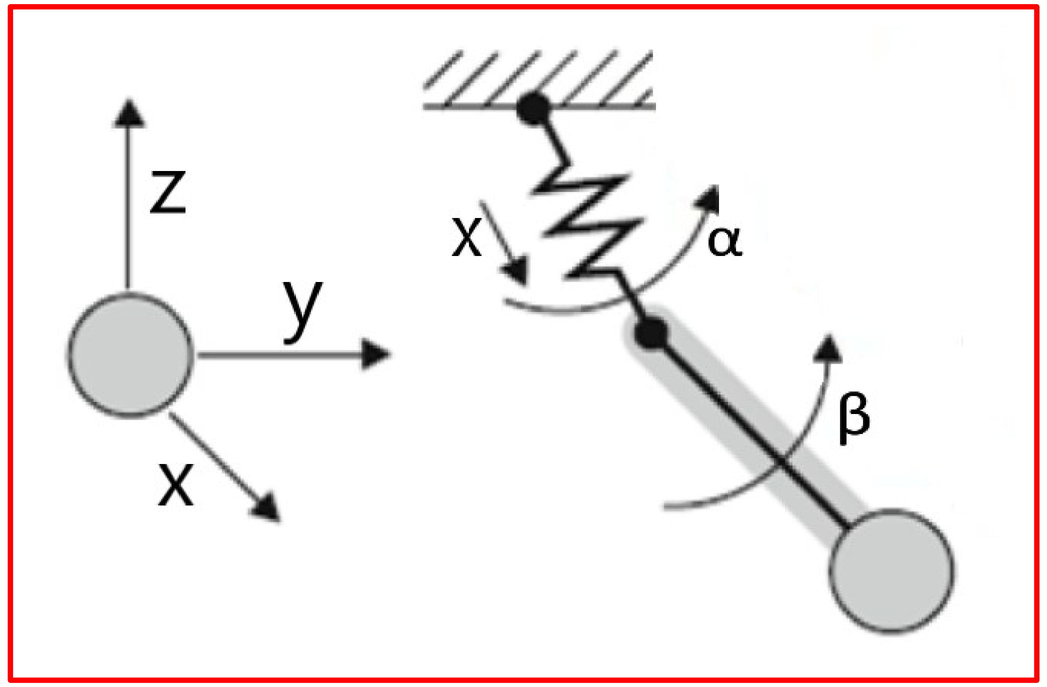
1 DOF



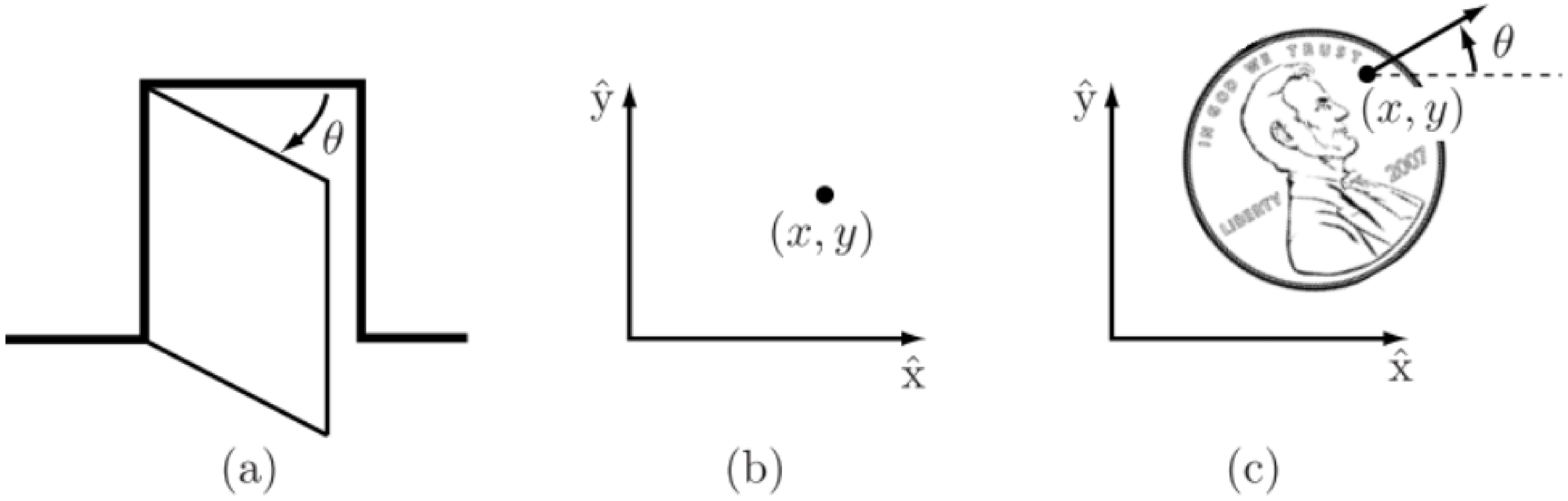
2 DOFs



3 DOFs



Degrees of Freedom (DOFs)

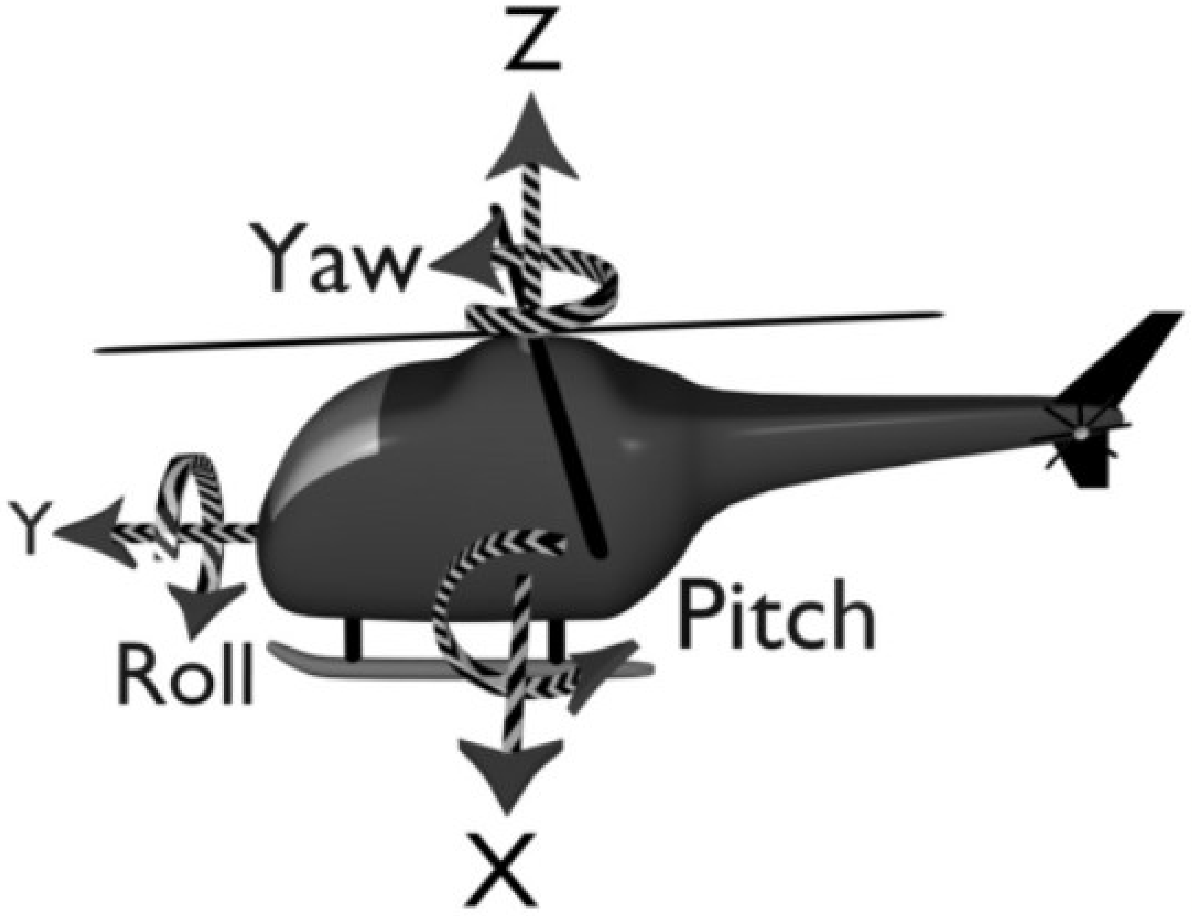


a) The configuration of a door is described by the angle ϑ.
b) The configuration of a point in a plane is described by coordinates (x, y).
c) The configuration of a coin on a table is described by (x, y, ϑ), where ϑ defines the direction in which Abraham Lincoln is looking.

Degrees of Freedom (DOFs) of a rigid body

6 DOFs of a rigid body in 3D space:

- **3 TRANSLATIONAL DOF**: x, y, z
- **3 ROTATIONAL DOF**: roll, pitch, yaw

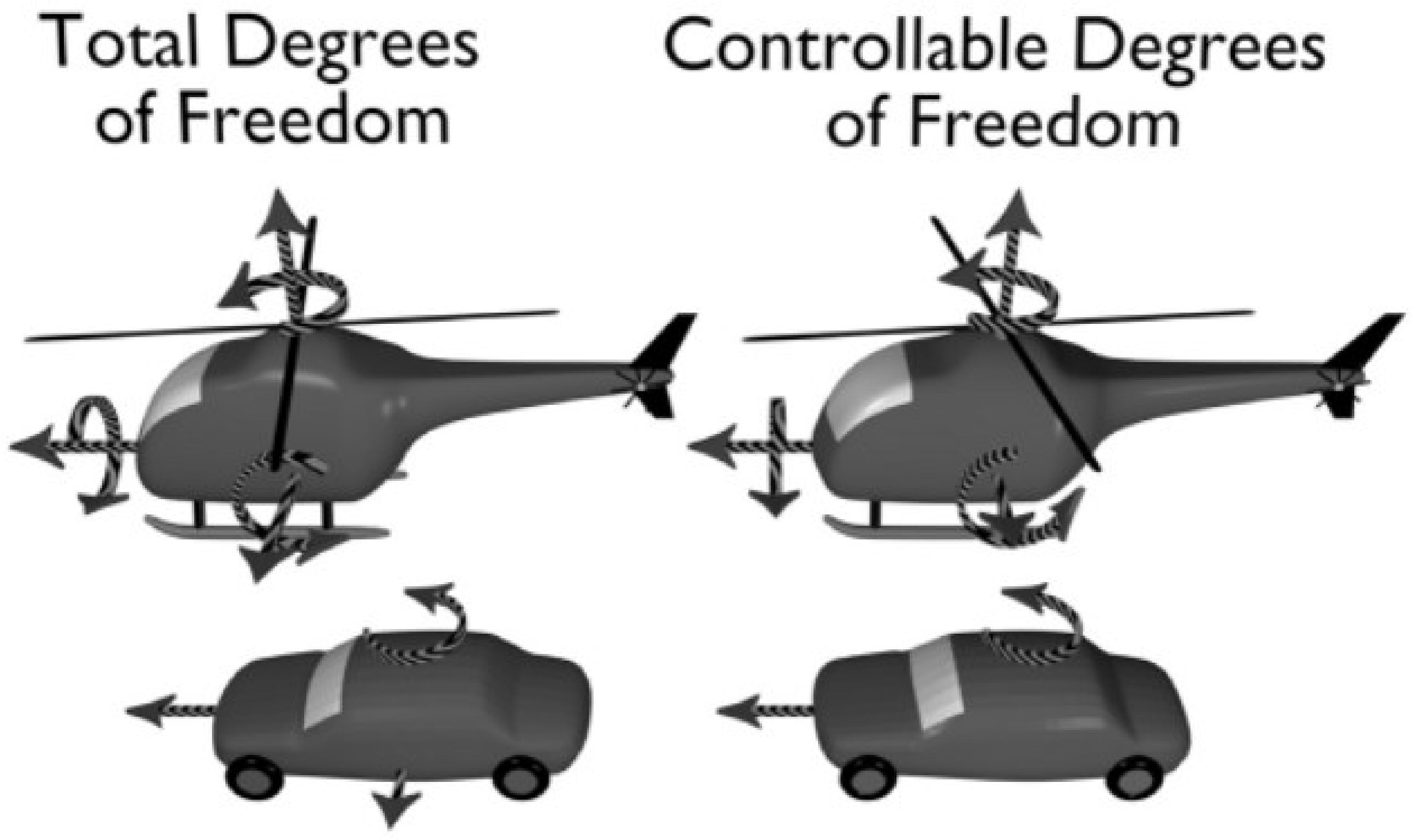


Degrees of Freedom (DOFs) of a rigid body

- **TOTAL DOFs (TDOF)**:
Helicopter ⟹[x, y, z, roll, pitch, yaw]
Car ⟹ [x, y, ϑ]

- **CONTROLLABLE DOFs (CDOF)**:
an actuator for every DOF

- **UNCONTROLLABLE DOFs**:
DOFs that are not controllable.



Degrees of Freedom (DOFs) of a rigid body

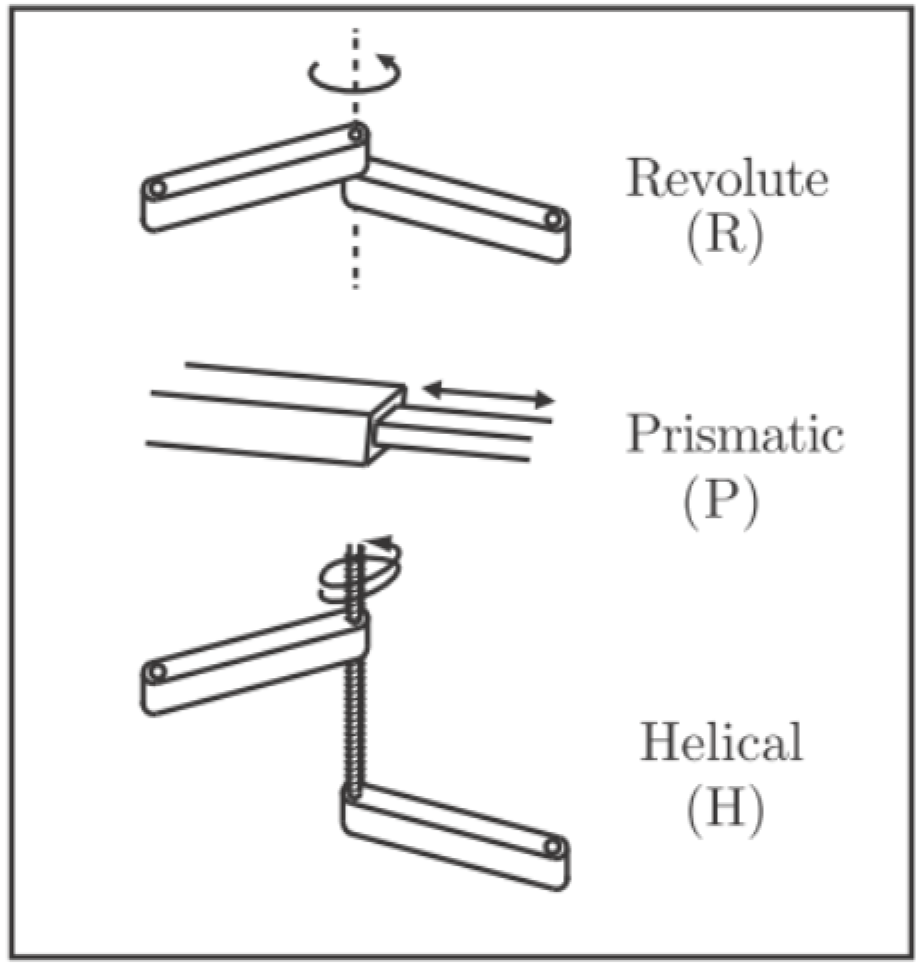
- **HOLONOMIC: CDOF = TDOF**
  When the total number of controllable DOF is equal to the total number of DOF on a robot (or actuator), the ratio is 1, and the robot is said to be holonomic. A holonomic robot or actuator can control all of its DOF.

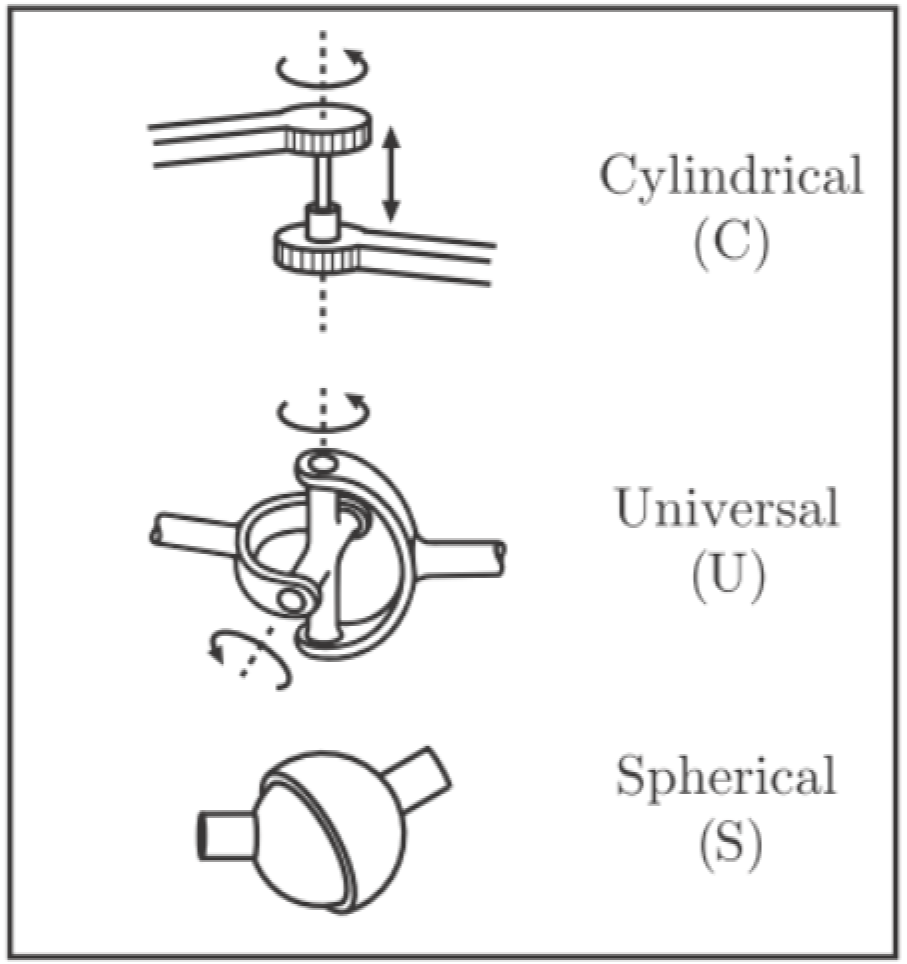
- **NONHOLONOMIC: CDOF < TDOF**
  When the number of controllable DOF is smaller than the total number of DOF, the ratio is less than 1, and the robot is said to be nonholonomic. A nonholonomic robot or actuator has more DOF than it can control.

- **REDUNDANT CDOF > TDOF**
  When the number of controllable DOF is larger than the total DOF, the ratio is larger than 1, and the robot is said to be redundant. A redundant robot or actuator has more ways of control than the DOF it has to control.

Joints and Degrees of Freedom (DOFs)

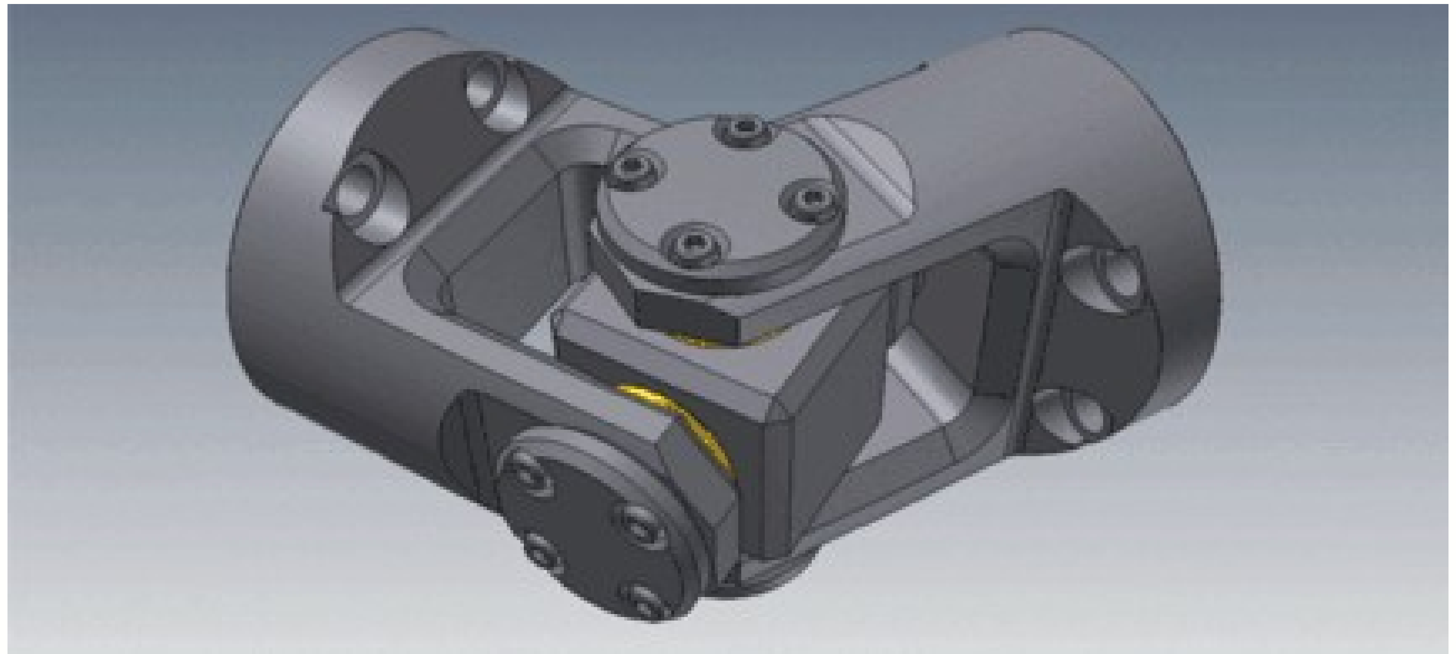
- Joint = set of two surfaces that can slide, keeping contact to one another
- Couple joint-link = robot degree of freedom (DOF)
- Link 0 = support base and origin of the reference coordinate frame for robot motion



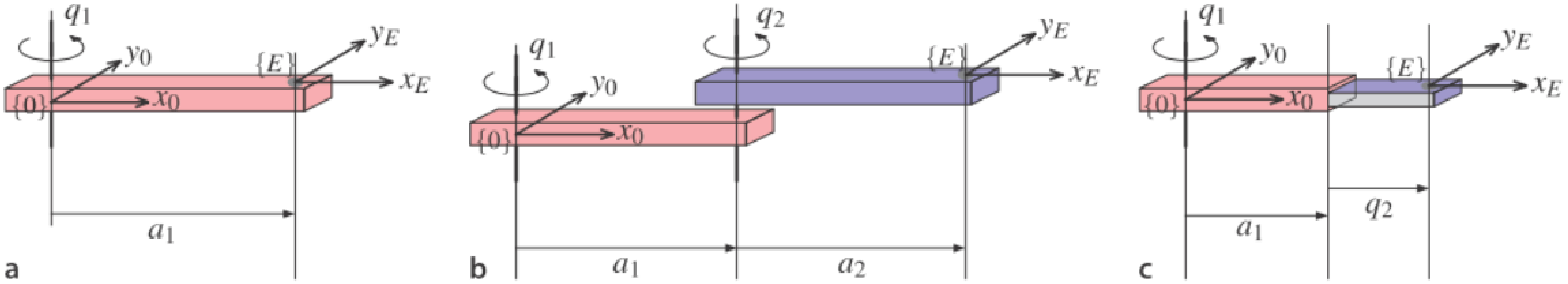


Joints Types

Robot joints are commonly revolute (rotational) but can also be prismatic (linear, sliding, telescopic, etc.).



Joints Types

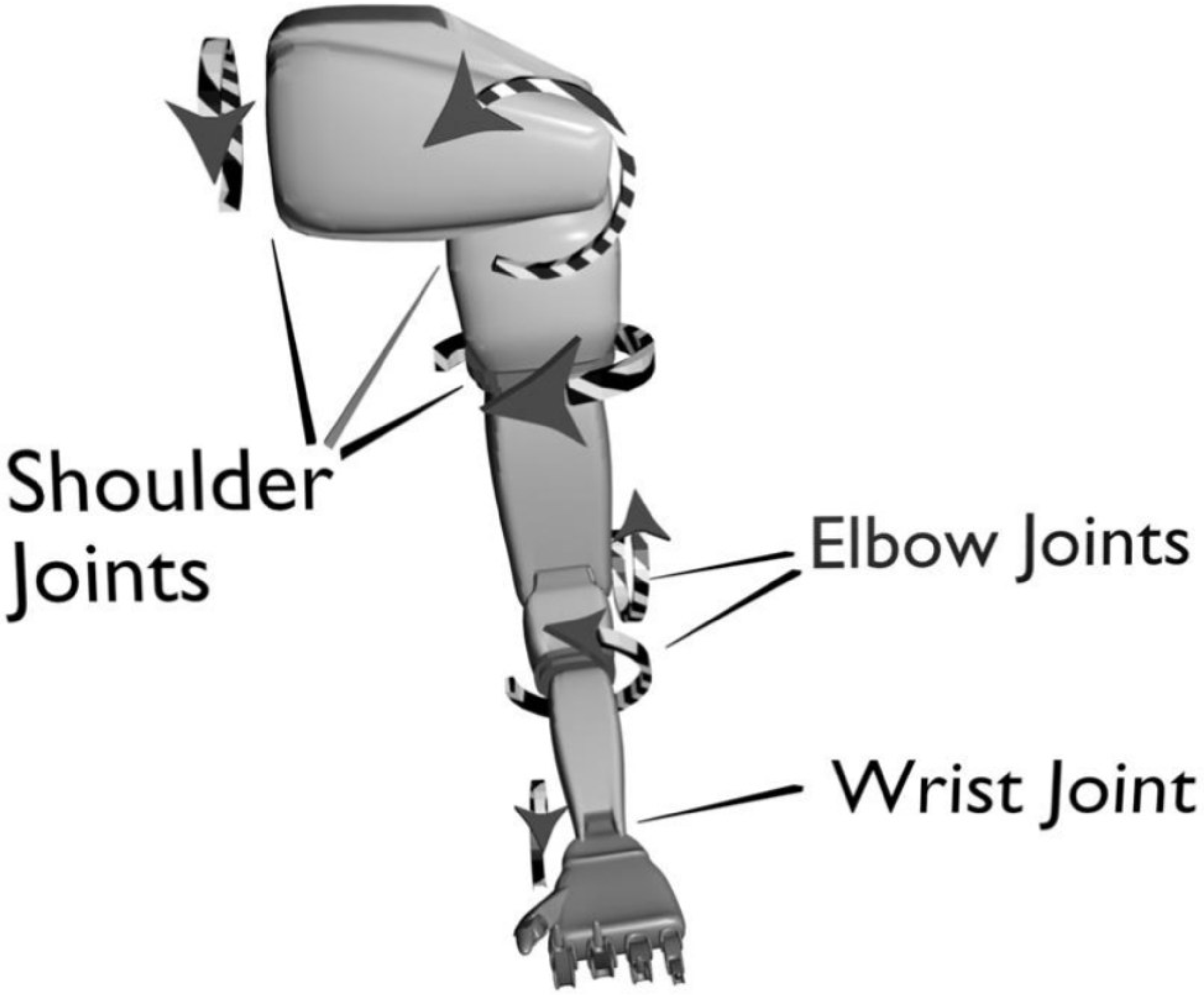


a) Planar arm with one rotational joint
b) Planar arm with two rotational joints
c) Planar arm with two joints: one rotational and one prismatic

The base {0} and end-effector {E} coordinate frames are shown.
The joint variables, angle or prismatic extension, are generalized coordinates and denoted by q

Robot manipulator

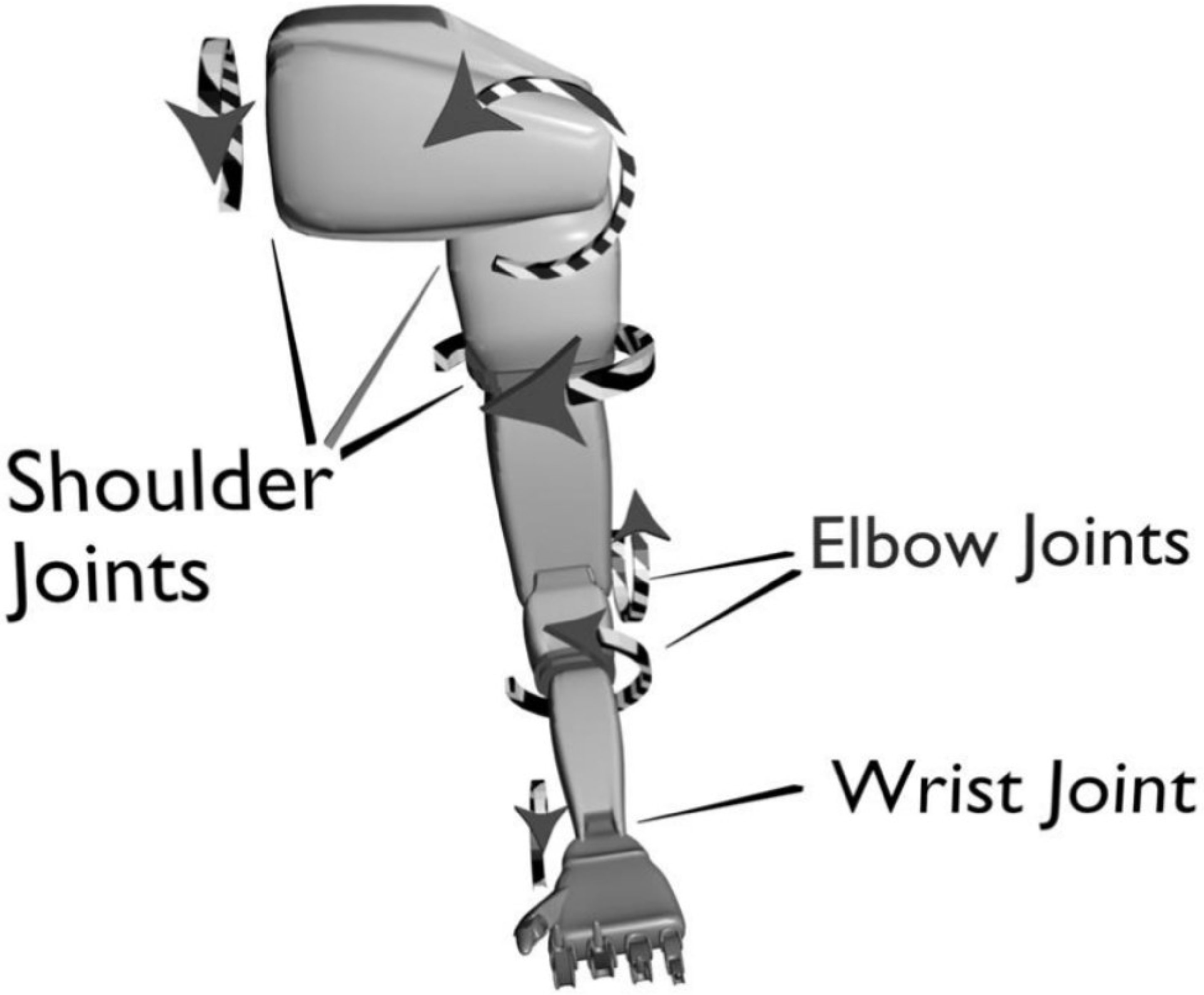
- **Definition**: open kinematic chain
- Sequence of rigid segments, or **links**, connected through revolute or translational **joints**, actuated by a motor
- One extremity is connected to a support base, the other one is free and equipped with a tool, named **end effector**



Robot manipulator

Consider again the human arm (not including the hand) with its seven DOF. Note that the arm itself has only three joints: the shoulder, the elbow, and the wrist, and those three joints control the seven DOF. This means some of the joints control more than one DOF; specifically, the shoulder joint has three DOF, as does the wrist. How does that work? Looking more closely at human anatomy, we find that the shoulder joint is a ball-and-socket joint (just like the hip) and is actuated by large muscles. It turns out that ball-and-socket joints are very difficult to create in artificial systems, not only robots but also realistically modeled animated characters in computer games.

The wrist, unlike the shoulder, is a smaller joint controlled by a collection of muscles and ligaments in the arm. It is also complex, but in an entirely different way from the shoulder.



Robot manipulator

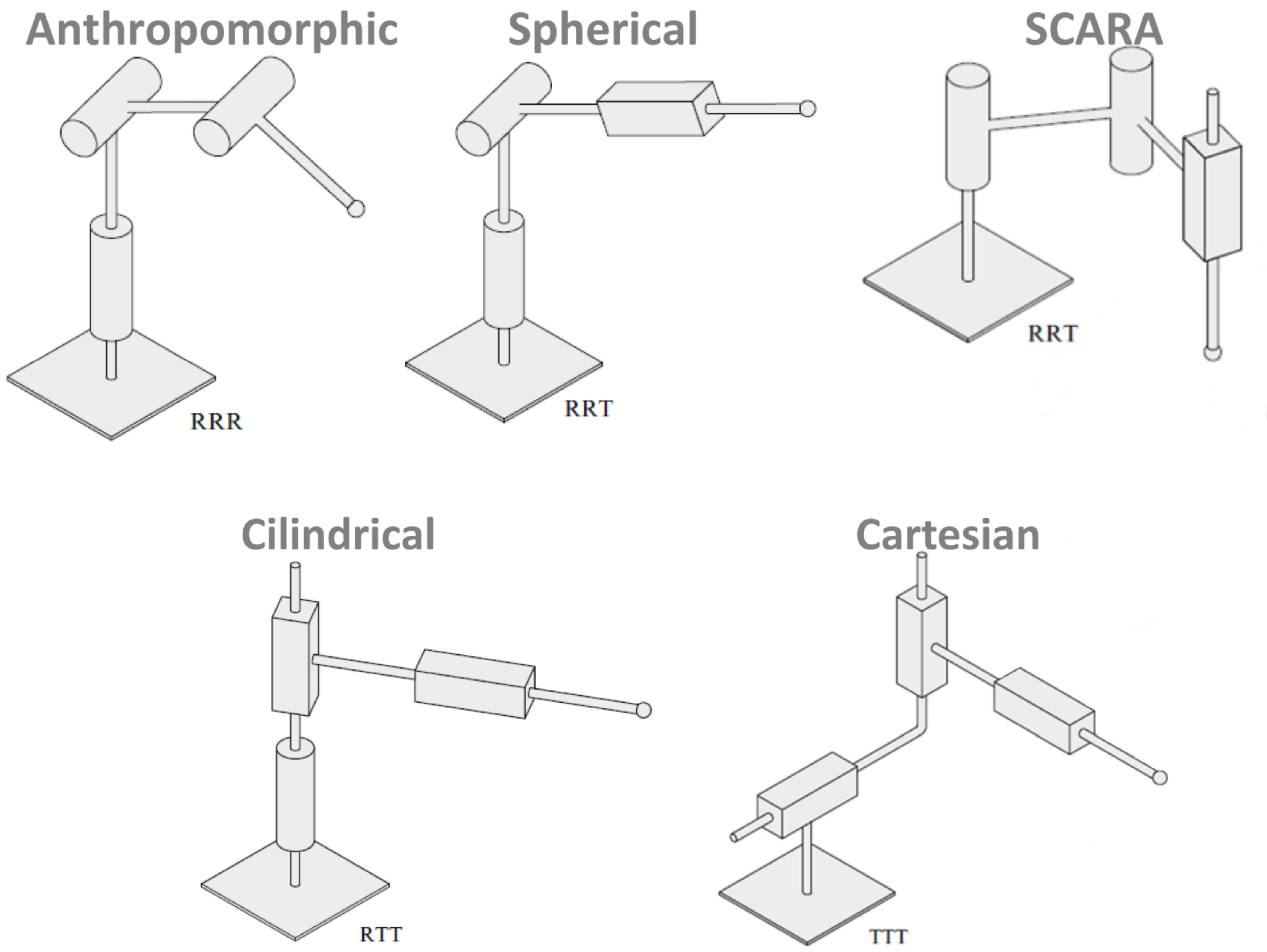
A robot manipulator consists of a robot arm, wrist, and gripper. The task of the robot manipulator is to place an object grasped by the gripper into an arbitrary pose. In this way also the industrial robot needs to have six degrees of freedom.

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.

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



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 $\mathbf{x} = (\mathbf{p}, \boldsymbol{\Phi})^T$ 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 expressed in degrees [°]
Dimension: N x 1,

- $\mathbf{x} = (\mathbf{p}, \boldsymbol{\Phi})^T$ 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).

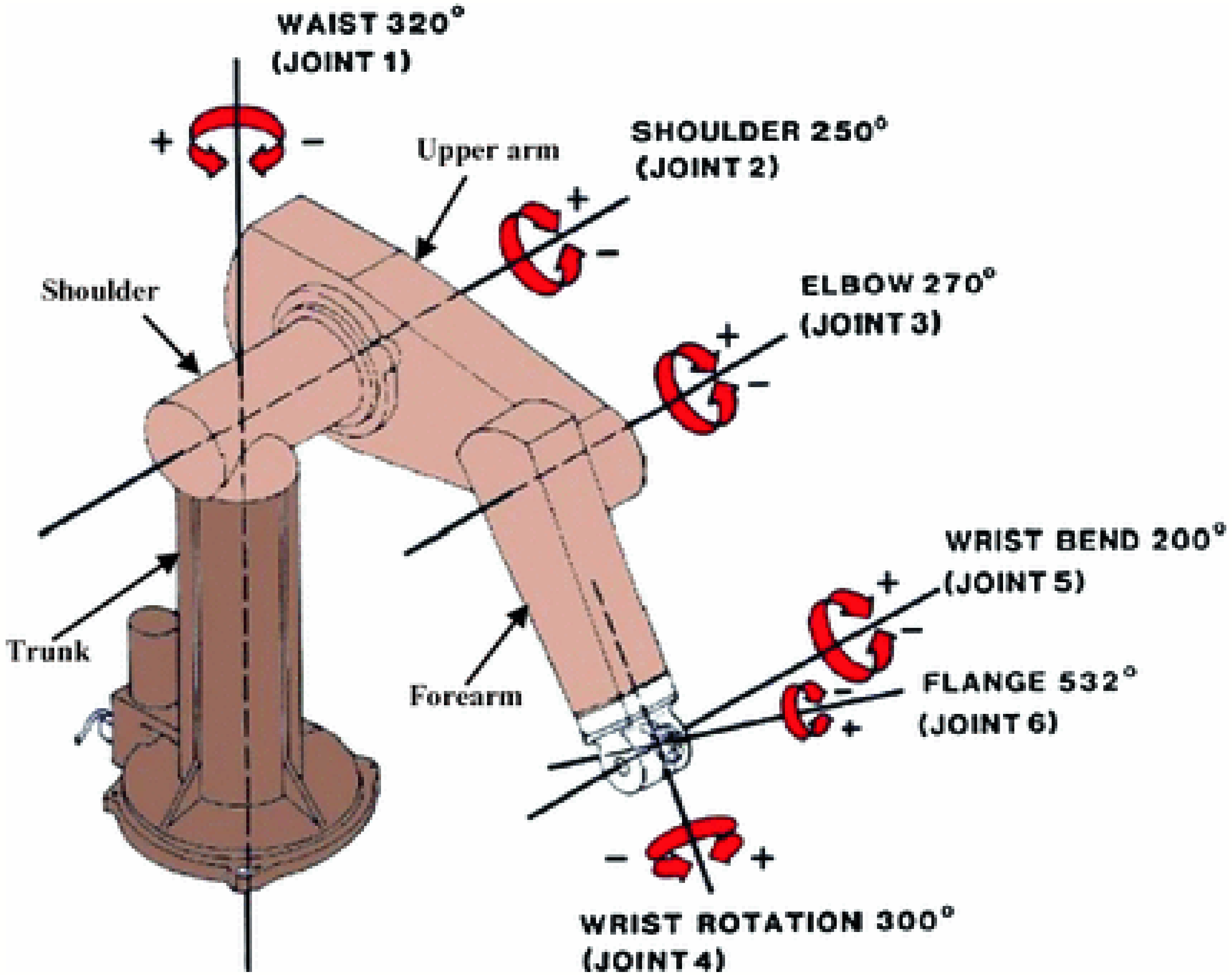
    - $\boldsymbol{\Phi}$, vector of orientation of the end effector,
    which has dimension 3x1 (roll, pitch, yaw angles).

Robot manipulator - PUMA

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



$x = (p, \Phi) = (x, y, z, roll, pitch, yaw)$

Ex. (0.7m,0.1m,0.5m,10°,-45°,5°)

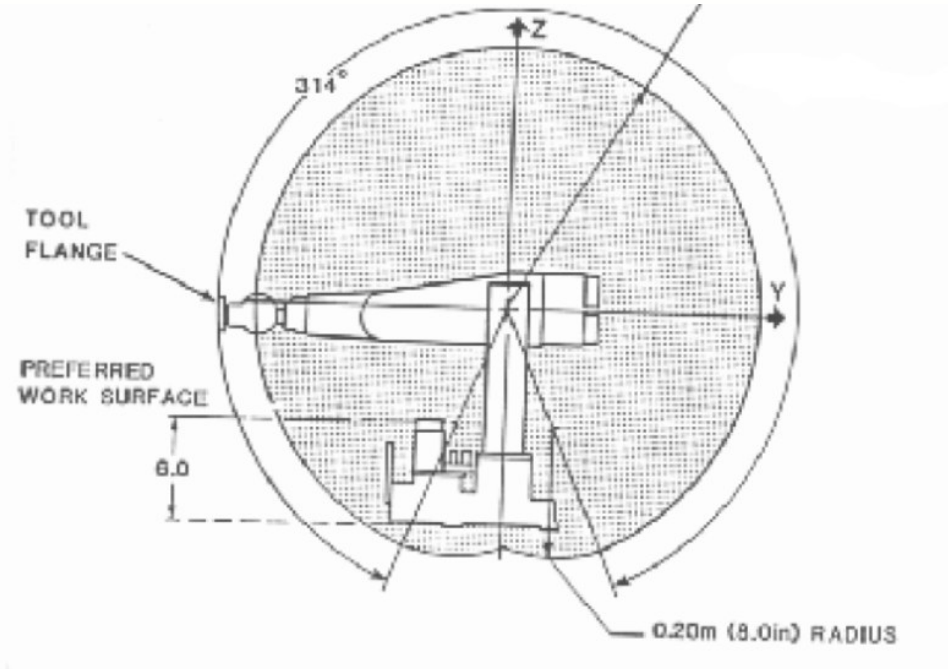
Workspace

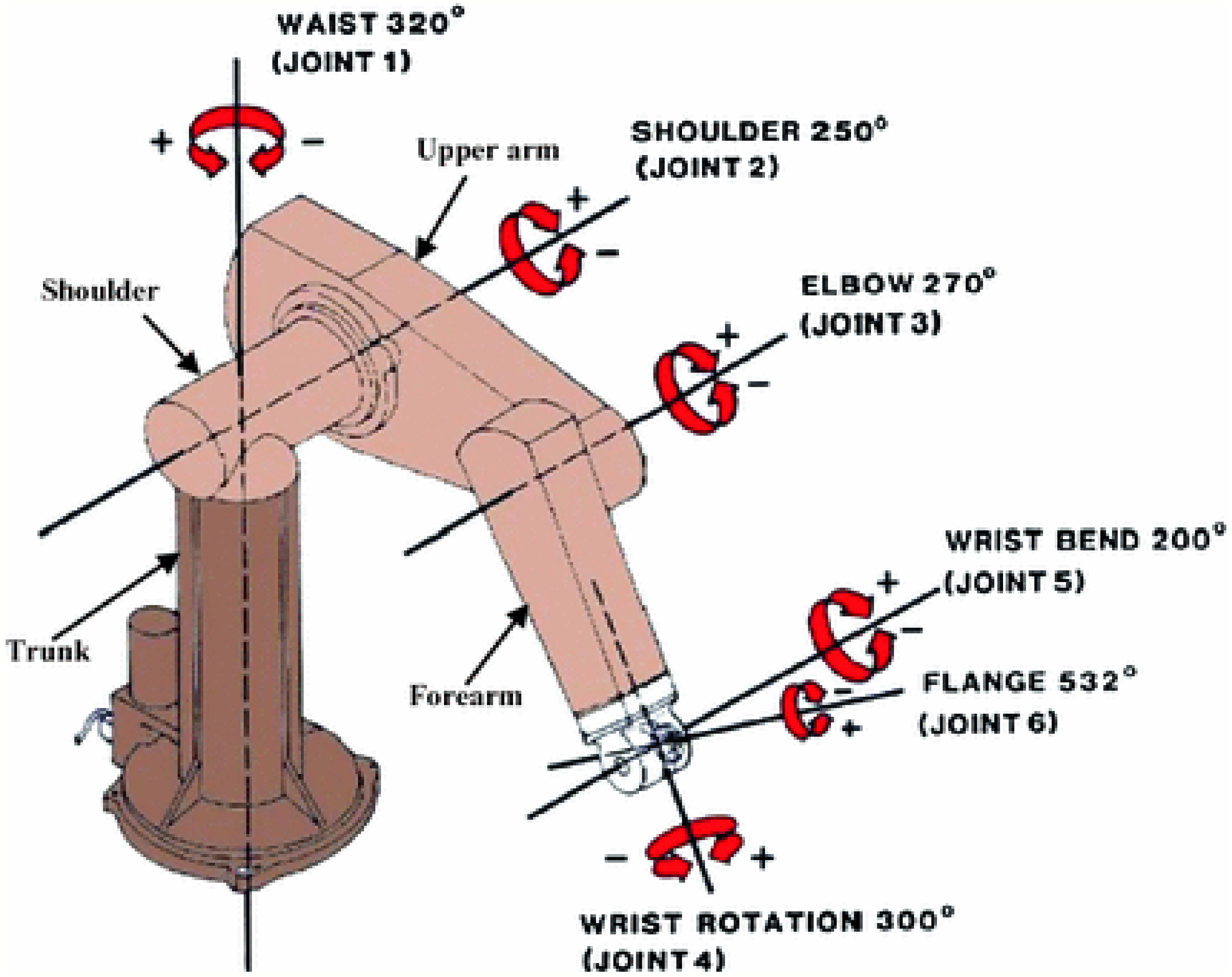
Robot workspace = region described by the origin of the end effector when the robot joints execute all possible motions

- **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.

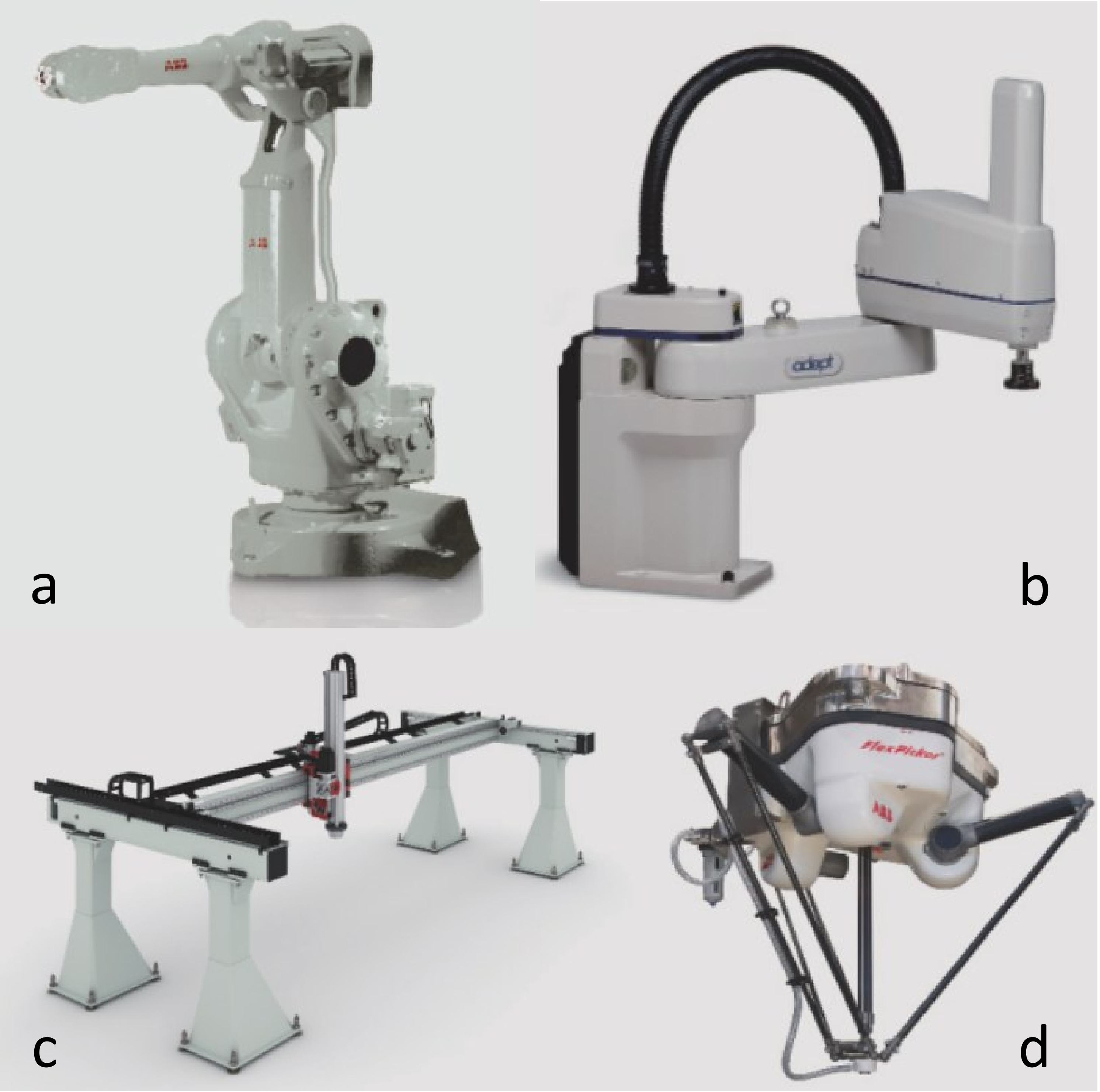
It depends on:

- Link lengths
- Joint ranges of motion





Arm-Type Robots



a) A 6DOF serial-link manipulator. General purpose industrial manipulator (source: ABB).
b) SCARA robot which has 4DOF,typically used for electronic assembly (photo of Adept Cobra s600 SCARA robot courtesy of Adept Technology, Inc.).
c) A gantry robot; the arm moves along an overhead rail (image courtesy of Güdel AG Switzerland | Mario Rothenbühler | www.gudel.com).
d) A parallel-link manipulator, the end-effector is driven by 6 parallel links (source: ABB)

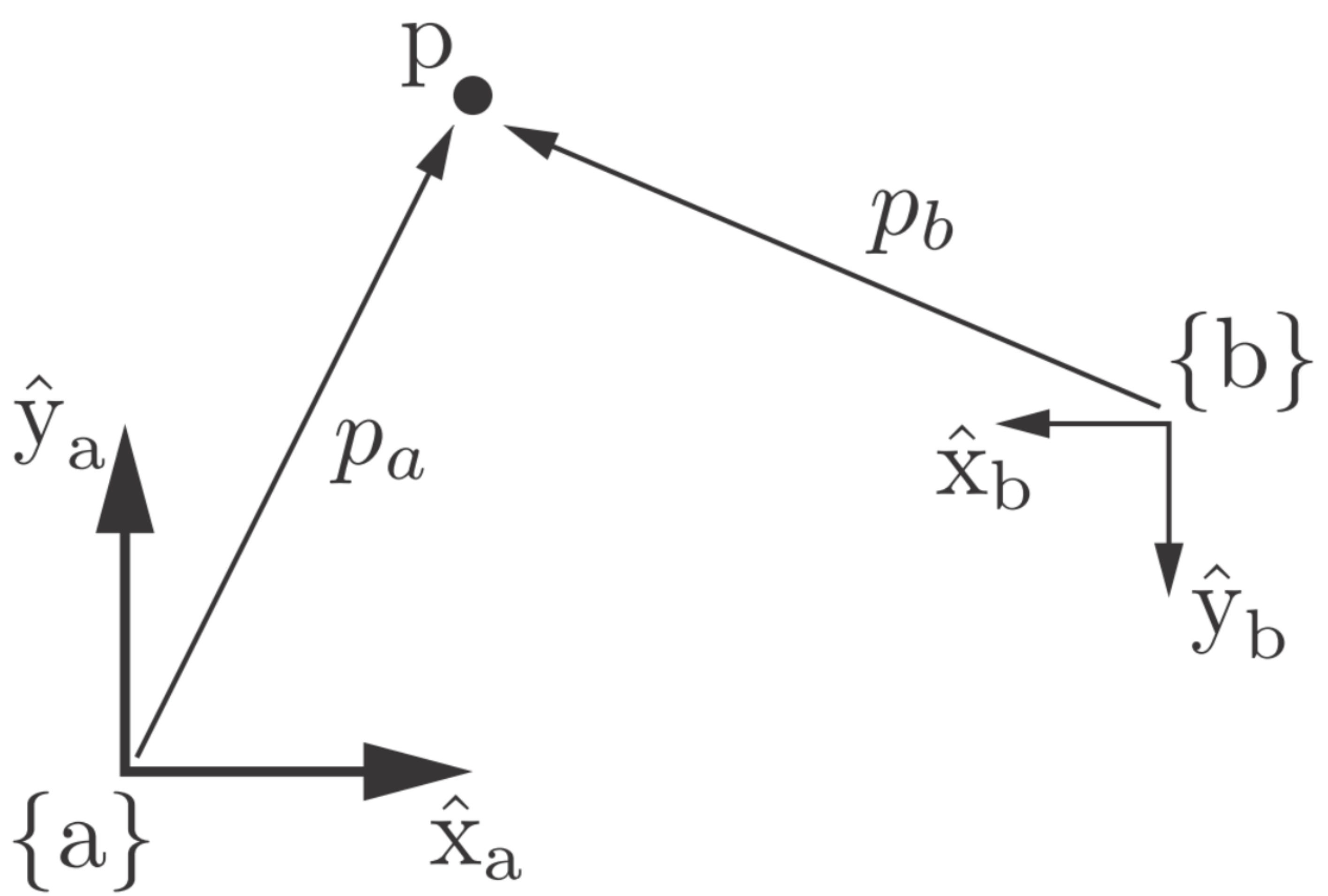
Representing Position and Orientation B

A point $\boldsymbol{p}$ in physical space can be represented as a vector from the reference frame origin to ***p.***

$$\boldsymbol{p} \in \boldsymbol{R}^{\boldsymbol{n}}$$

A different choice of reference frame and length scale for physical space leads to a different representation $\boldsymbol{p} \in \boldsymbol{R}^{\boldsymbol{n}}$ for the same point p in physical space.

If we fix a reference frame $\{\, a \,\}$, with unit coordinate axes $\hat{x}_a$ and $\hat{y}_a$, we can represent $\boldsymbol{p}$ as $p_a = (1{,}2)$. If we fix a reference frame $\{\, b \,\}$ at a different location, a different orientation, and a different length scale, we can represent p as $p_b = (4, -2)$.



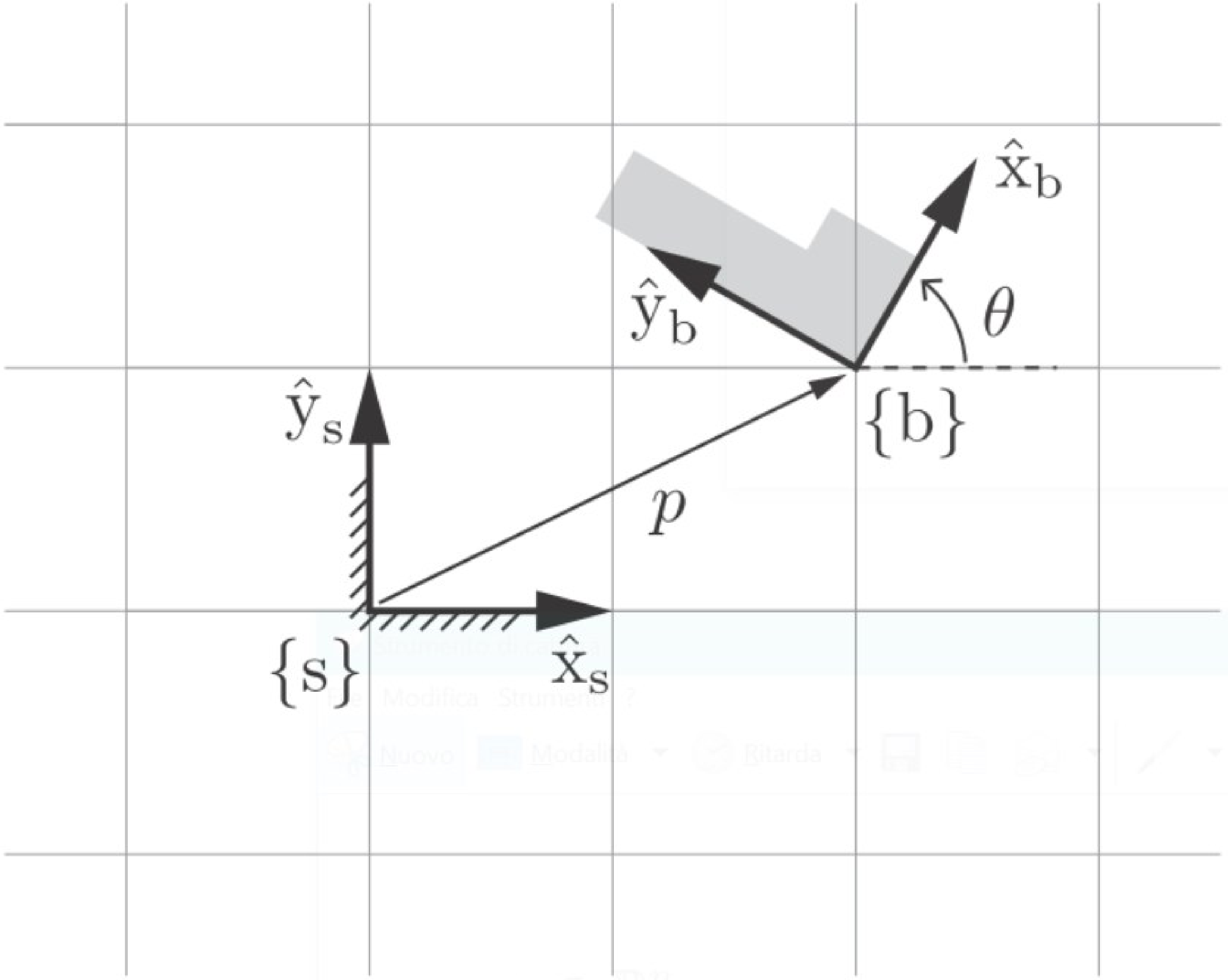
"Modern Robotics: Mechanics, Planning, and Control," Kevin M. Lynch and Frank C. Park, Cambridge University Press, 2017, ISBN 9781107156302
Chapter 3 Rigid-Body Motions

Representing Position and Orientation B

A reference frame can be placed anywhere in space, and any reference frame leads to an equally valid representation of the underlying space and the objects in it.

We always assume that exactly one stationary **fixed frame**, or **space frame**, denoted $\{s\}$, has been defined.

We often assume that at least one frame has been attached to some moving rigid body.
This **body frame**, denoted $\{b\}$, is the stationary frame that is coincident with the body-attached frame at any instant and it is common to attach the origin of the $\{b\}$ frame to some important point on the body, such as its center of mass, this is not necessary.

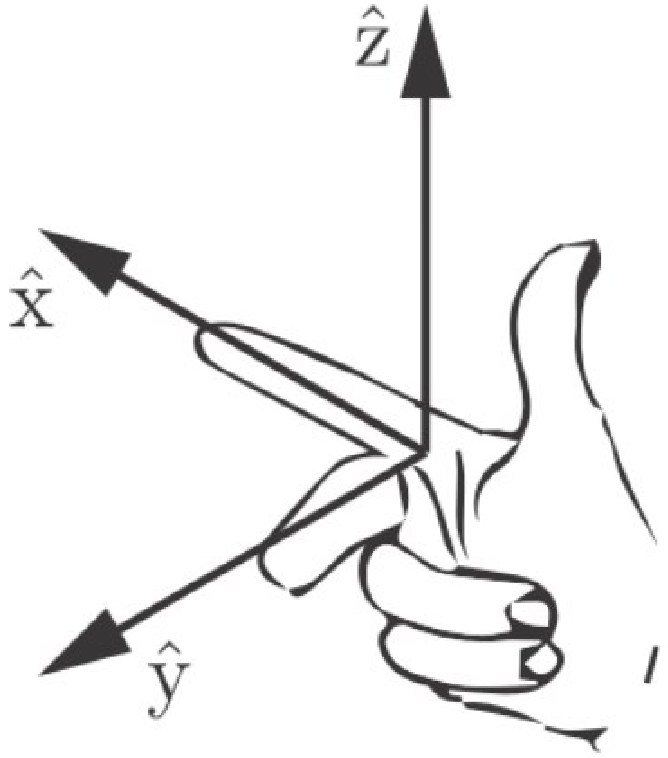


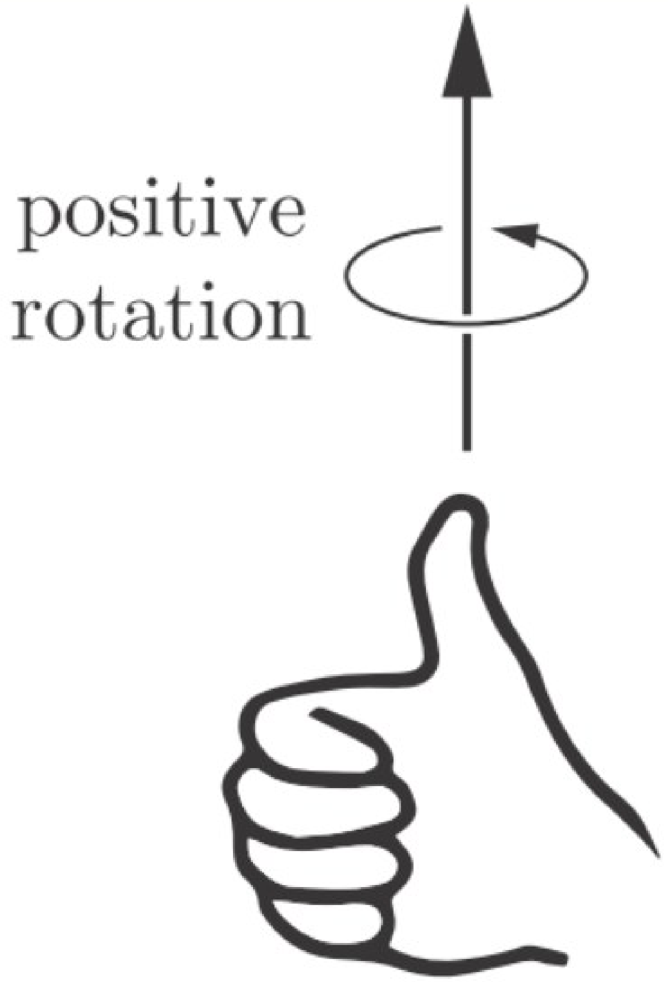
"Modern Robotics: Mechanics, Planning, and Control," Kevin M. Lynch and Frank C. Park, Cambridge University Press, 2017, ISBN 9781107156302
Chapter 3 Rigid-Body Motions

Representing Position and Orientation B

The $\hat{x}$, $\hat{y}$, and $\hat{z}$ axes of a right-handed reference frame are aligned with the index finger, middle finger, and thumb of the right hand, respectively.

A positive rotation about an axis is in the direction in which the fingers of the right hand curl when the thumb is pointed along the axis.





"Modern Robotics: Mechanics, Planning, and Control," Kevin M. Lynch and Frank C. Park, Cambridge University Press, 2017, ISBN 9781107156302
Chapter 3 Rigid-Body Motions

Rigid-Body Motions in the Plane

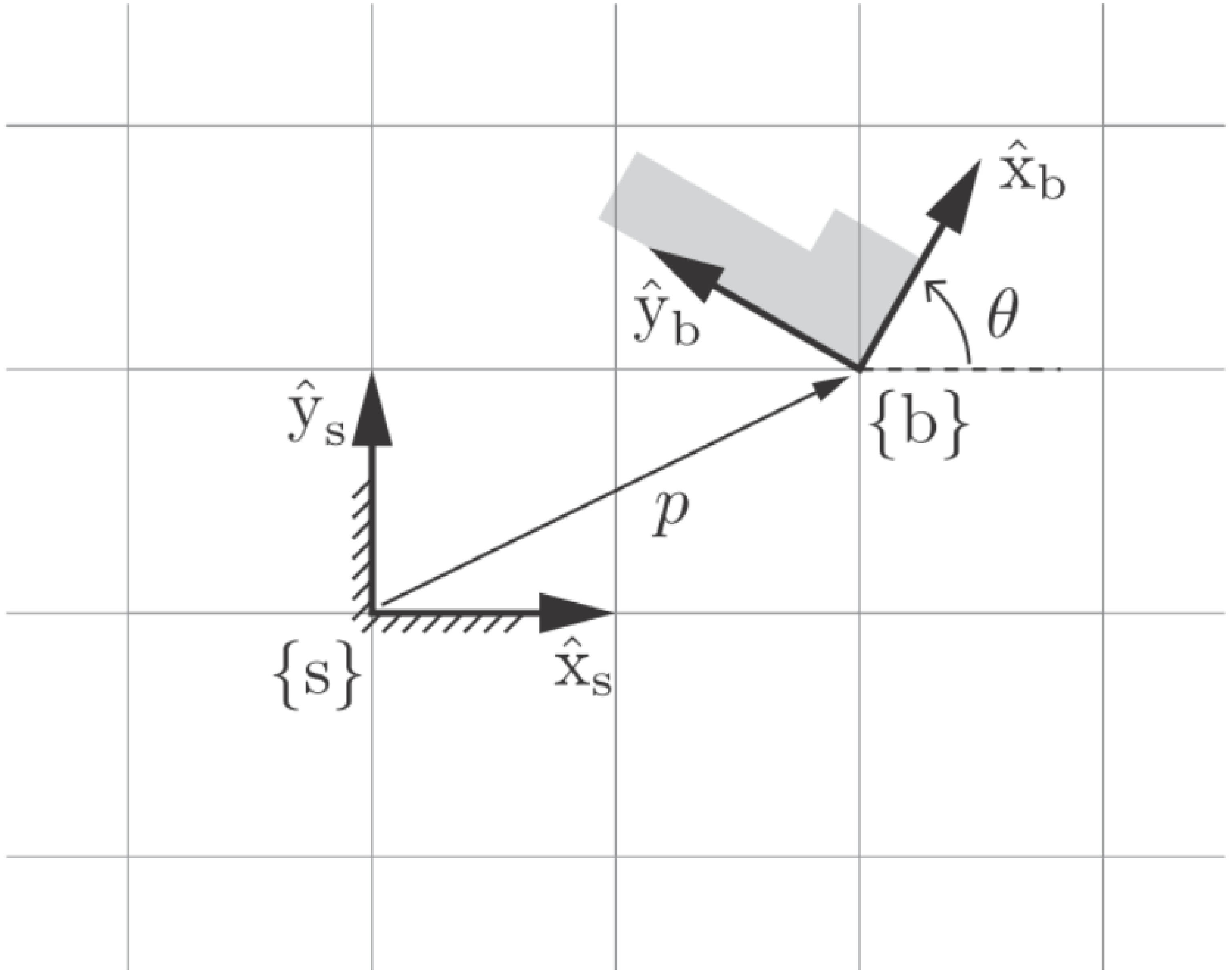
A fixed reference frame $\{ s \}$ have been chosen with unit axes $\hat{x}_s$ and $\hat{y}_s$ (the hat notation indicates a unit vector).

We attach a reference frame $\{ b \}$ with unit axes $\hat{x}_b$ and $\hat{y}_b$ to the planar body.

To describe the **configuration** of the planar body, only the position and orientation of the body frame with respect to the fixed frame need to be specified.

The body-frame origin p can be expressed:

$$p = p_x \hat{x}_s + p_y \hat{y}_s$$



The body frame $\{b\}$ is expressed in the fixed-frame coordinates $\{s\}$ by the vector p and the directions of the unit axes $\hat{x}_b$ and $\hat{y}_b$. In this example, $p = (2, 1)$ and $\theta = 60°$, so $\hat{x}_b = (\cos\theta, \sin\theta) = (0.5, 1/\sqrt{2})$ and $\hat{y}_b = (-\sin\theta, \cos\theta) = (-1/\sqrt{2}, 0.5)$.

"Modern Robotics: Mechanics, Planning, and Control," Kevin M. Lynch and Frank C. Park, Cambridge University Press, 2017, ISBN 9781107156302
Chapter 3 Rigid-Body Motions

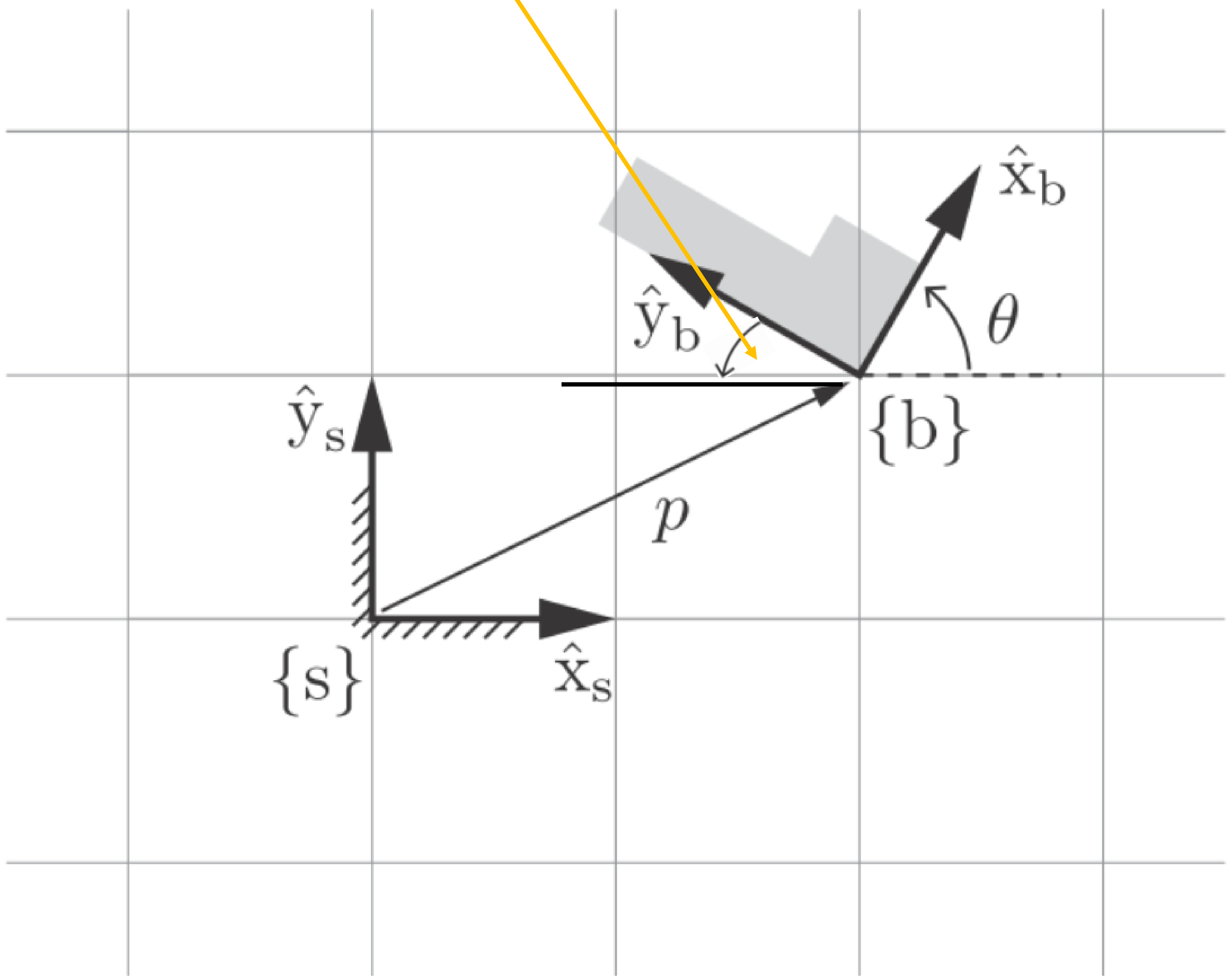
Rigid-Body Motions in the Plane

$$\pi - \frac{\pi}{2} - \vartheta = \frac{\pi}{2} - \vartheta$$

$$p = p_x \hat{x}_s + p_y \hat{y}_s$$

The simplest way to describe the orientation of the body frame $\{\, b \,\}$ relative to the fixed frame $\{\, s \,\}$ is to specify the directions of $\hat{x}_b$ and $\hat{y}_b$ relative to { s } , in the form:

$$\hat{x}_b = \cos\theta \hat{x}_s + \sin\theta \hat{y}_s$$
$$\hat{y}_b = -\sin\theta \hat{x}_s + \cos\theta \hat{y}_s$$



The body frame {b} is expressed in the fixed-frame coordinates {s} by the vector p and the directions of the unit axes $\hat{x}_b$ and $\hat{y}_b$. In this example, $p = (2, 1)$ and $\theta = 60°$, so $\hat{x}_b = (\cos\theta, \sin\theta) = (0.5, 1/\sqrt{2})$ and $\hat{y}_b = (-\sin\theta, \cos\theta) = (-1/\sqrt{2}, 0.5)$.

"Modern Robotics: Mechanics, Planning, and Control," Kevin M. Lynch and Frank C. Park, Cambridge University Press, 2017, ISBN 9781107156302
Chapter 3 Rigid-Body Motions

Rigid-Body Motions in the Plane

$\hat{x}_s$,$\hat{y}_s$, $\hat{x}_b$ and $\hat{y}_b$ (the hat notation indicates a unit vector)

$$\hat{x}_b = \cos\theta\hat{x}_s + \sin\theta\hat{y}_s$$

$$\hat{y}_b = -\cos\alpha\hat{x}_s + \sin\alpha\,\hat{y}_s$$

Note that :

- $\alpha = (\pi - \frac{\pi}{2} - \vartheta) = \frac{\pi}{2} - \vartheta$
- $\cos\left(\frac{\pi}{2} - \vartheta\right) = \sin(\vartheta)\ and\ \sin\left(\frac{\pi}{2} - \vartheta\right) = \cos(\vartheta)$

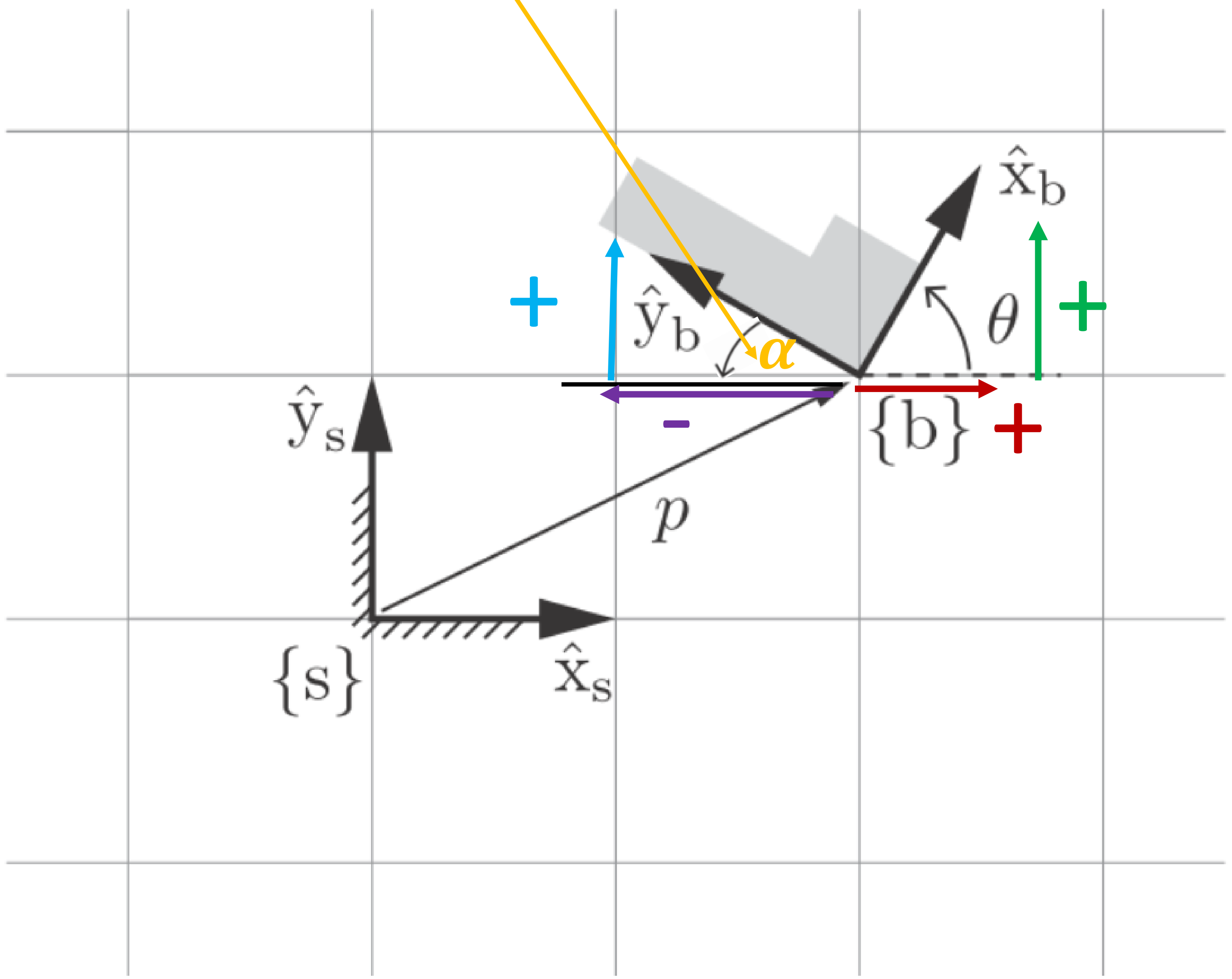
Therefore:

$$\hat{y}_b = -\sin\theta\hat{x}_s + \cos\theta\hat{y}_s$$

If $\mathbf{\theta = 0}$ then:

$$\hat{x}_b = \cos(0)\hat{x}_s + \sin(0)\hat{y}_s = 1 * \hat{x}_s + 0 * \hat{y}_s = \hat{x}_s$$
$$\hat{y}_b = -\sin(0)\hat{x}_s + \cos(0)\hat{y}_s = 0 * \hat{x}_s + 1 * \hat{y}_s = \hat{y}_s$$

$$\alpha = \pi - \frac{\pi}{2} - \vartheta = \frac{\pi}{2} - \vartheta$$



The body frame {b} is expressed in the fixed-frame coordinates {s} by the vector p and the directions of the unit axes $\hat{x}_b$ and $\hat{y}_b$. In this example, $p = (2, 1)$ and $\theta = 60°$, so $\hat{x}_b = (\cos\theta, \sin\theta) = (0.5, 1/\sqrt{2})$ and $\hat{y}_b = (-\sin\theta, \cos\theta) = (-1/\sqrt{2}, 0.5)$.

"Modern Robotics: Mechanics, Planning, and Control," Kevin M. Lynch and Frank C. Park, Cambridge University Press, 2017, ISBN 9781107156302
Chapter 3 Rigid-Body Motions

Rigid-Body Motions in the Plane

$$p = p_x\hat{x}_s + p_y\hat{y}_s$$

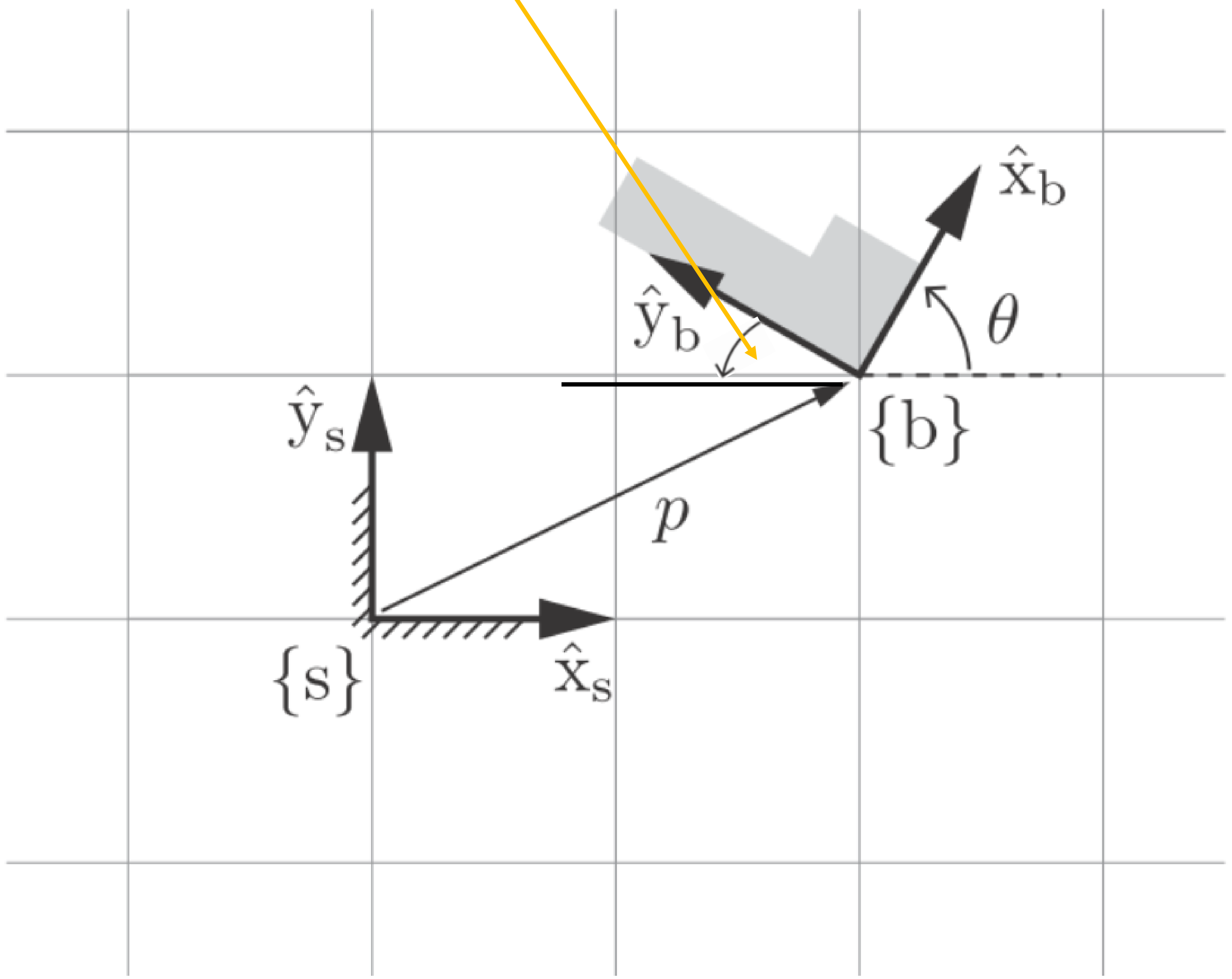
The simplest way to describe the orientation of the body frame $\{\, b \,\}$ relative to the fixed frame $\{\, s \,\}$ is to specify the directions of $\hat{x}_b$ and $\hat{y}_b$ relative to { s }, in the form:

$$\hat{x}_b = \cos\theta\hat{x}_s + \sin\theta\hat{y}_s$$
$$\hat{y}_b = -\sin\theta\hat{x}_s + \cos\theta\hat{y}_s$$

Expressed everything in terms of $\{\, s \,\}$ then, just as the point ***p*** can be represented as a column vector $p \in R^2$ of the form

$$p = \begin{bmatrix} p_x \\ p_y \end{bmatrix}$$

$$\pi - \frac{\pi}{2} - \vartheta = \frac{\pi}{2} - \vartheta$$



The body frame {b} is expressed in the fixed-frame coordinates {s} by the vector p and the directions of the unit axes $\hat{x}_b$ and $\hat{y}_b$. In this example, $p = (2, 1)$ and $\theta = 60°$, so $\hat{x}_b = (\cos\theta, \sin\theta) = (0.5, 1/\sqrt{2})$ and $\hat{y}_b = (-\sin\theta, \cos\theta) = (-1/\sqrt{2}, 0.5)$.

"Modern Robotics: Mechanics, Planning, and Control," Kevin M. Lynch and Frank C. Park, Cambridge University Press, 2017, ISBN 9781107156302
Chapter 3 Rigid-Body Motions

Rigid-Body Motions in the Plane

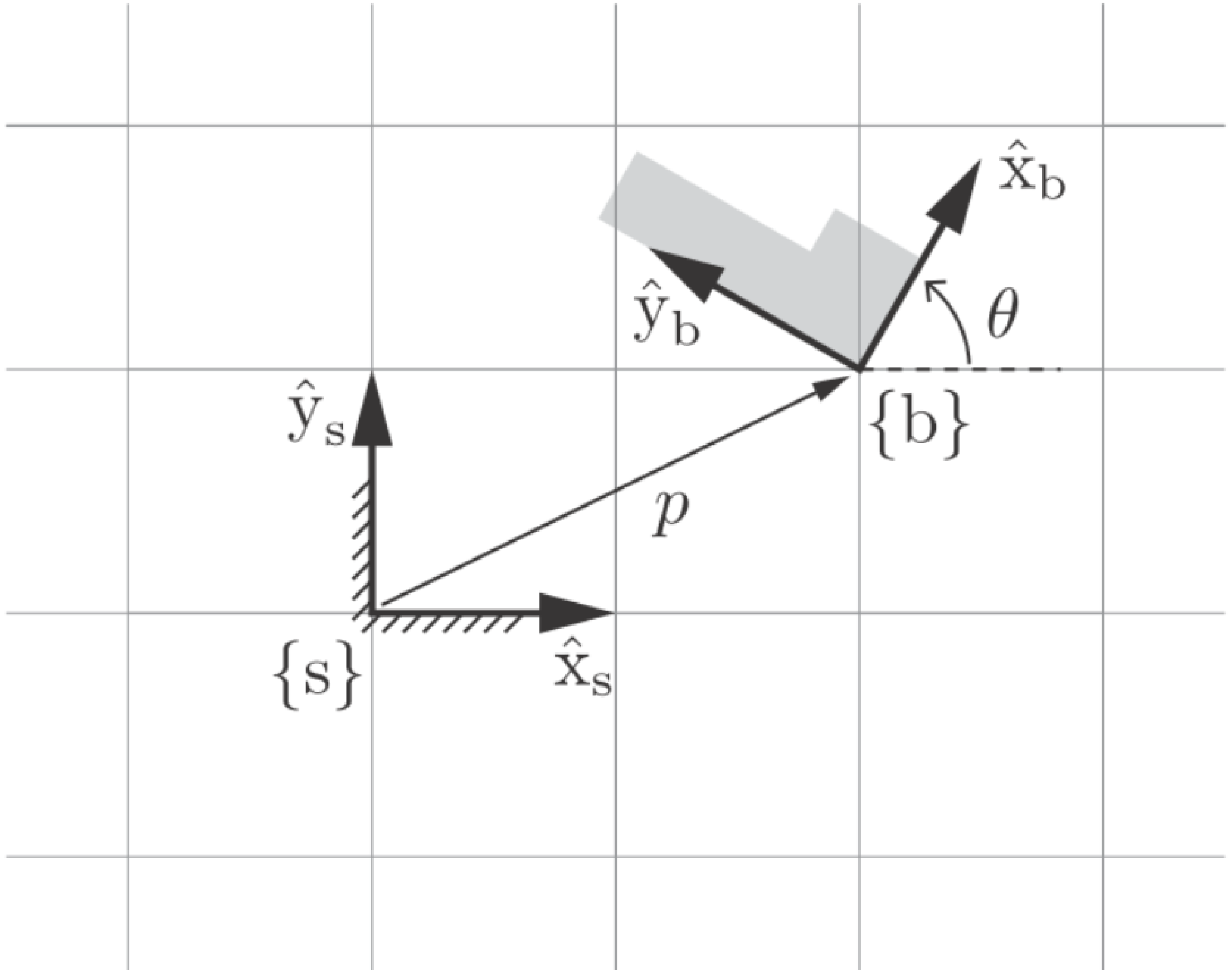
The two vectors $\hat{x}_b$ and $\hat{y}_b$ can also be written as column vectors and packaged into the following 2×2 matrix **P**:

$$P = [\hat{\mathrm{x}}_\mathrm{b} \;\; \hat{\mathrm{y}}_\mathrm{b}] = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

The matrix P is an example of a **rotation matrix**. Although P consists of four numbers, they are subject to three constraints:

- Each column of **P** must be a unit vector
- The two columns must be orthogonal to each other
- The one remaining degree of freedom is parametrized by $\boldsymbol{\vartheta}$.

Together, the pair $(\boldsymbol{P}, \boldsymbol{p})$ provides a description of the orientation and position of $\{\, b \,\}$ relative to $\{\, s \,\}$.



The body frame $\{\mathrm{b}\}$ is expressed in the fixed-frame coordinates $\{\mathrm{s}\}$ by the vector p and the directions of the unit axes $\hat{\mathrm{x}}_\mathrm{b}$ and $\hat{\mathrm{y}}_\mathrm{b}$. In this example, $p = (2, 1)$ and $\theta = 60^\circ$, so $\hat{\mathrm{x}}_\mathrm{b} = (\cos\theta, \sin\theta) = (0.5, 1/\sqrt{2})$ and $\hat{\mathrm{y}}_\mathrm{b} = (-\sin\theta, \cos\theta) = (-1/\sqrt{2}, 0.5)$.

"Modern Robotics: Mechanics, Planning, and Control," Kevin M. Lynch and Frank C. Park, Cambridge University Press, 2017, ISBN 9781107156302
Chapter 3 Rigid-Body Motions

Rigid-Body Motions in the Plane

Repeating the approach and expressing $\{c\}$ in $\{s\}$ as the pair $(\boldsymbol{R}, \boldsymbol{r})$ we can write:

$$r = \begin{bmatrix} r_x \\ r_y \end{bmatrix}, \qquad R = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix}$$

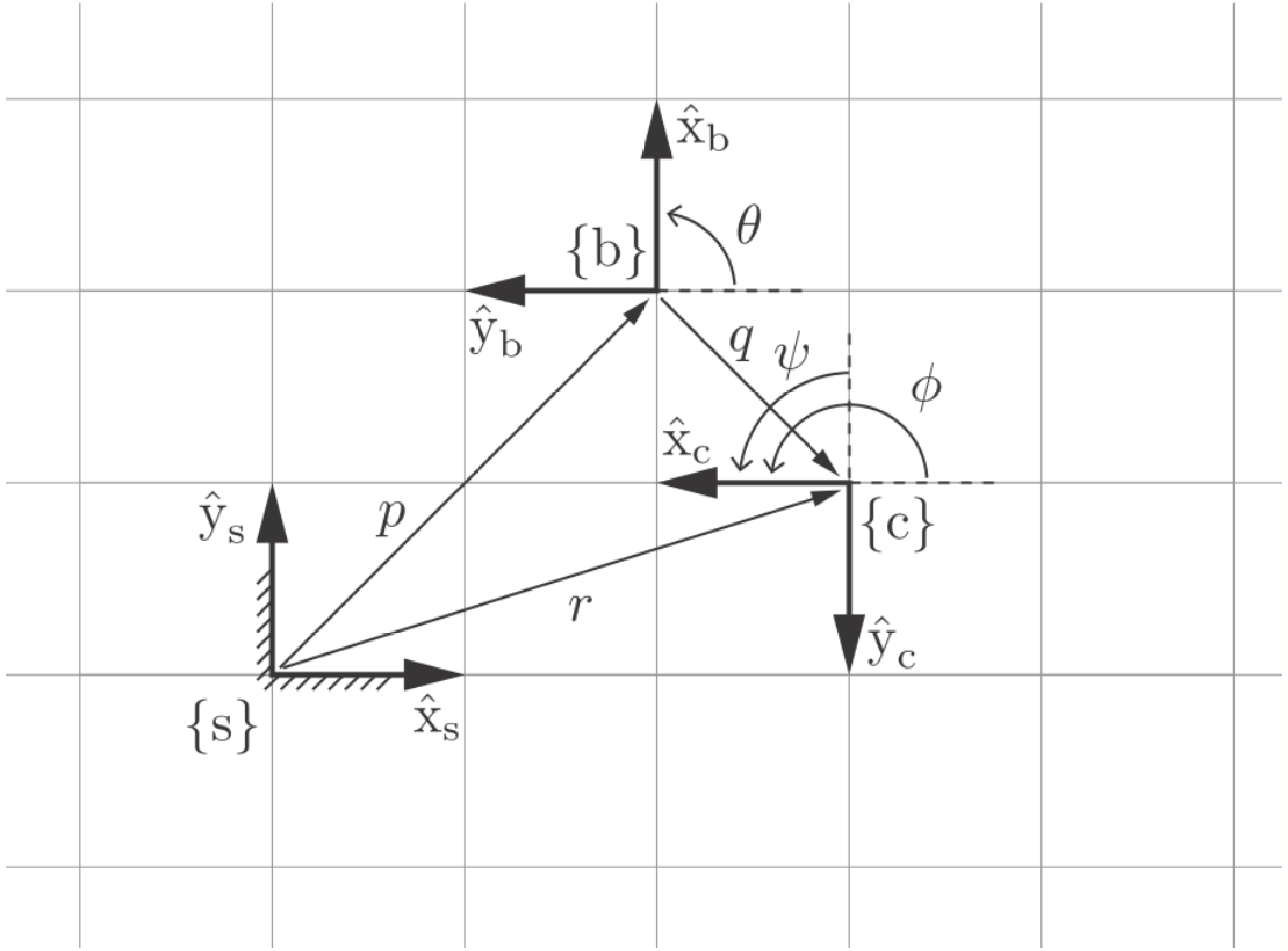
And also the frame $\{c\}$ relative to $\{b\}$:

$$q = \begin{bmatrix} q_x \\ q_y \end{bmatrix}, \qquad Q = \begin{bmatrix} \cos\psi & -\sin\psi \\ \sin\psi & \cos\psi \end{bmatrix}$$

If we know $(\boldsymbol{Q}, \boldsymbol{q}) : \{\boldsymbol{c} \rightarrow \mathbf{b}\}$ and $(\boldsymbol{P}, \boldsymbol{p}) : \{\boldsymbol{b} \rightarrow \boldsymbol{s}\}$, we can compute the configuration $\{\boldsymbol{c}\}$ relative to $\{\boldsymbol{s}\}$ as:

$$R = PQ \quad (\text{convert } Q \text{ to the } \{\text{s}\} \text{ frame})$$
$$r = Pq + p \quad (\text{convert } q \text{ to the } \{\text{s}\} \text{ frame and vector-sum with } p)$$

Thus $(\boldsymbol{P}, \boldsymbol{p})$ not only represents a configuration of $\{b\}$ in $\{s\}$; it can also be used to convert the representation of a point or frame from { b } coordinates to $\{s\}$coordinates.



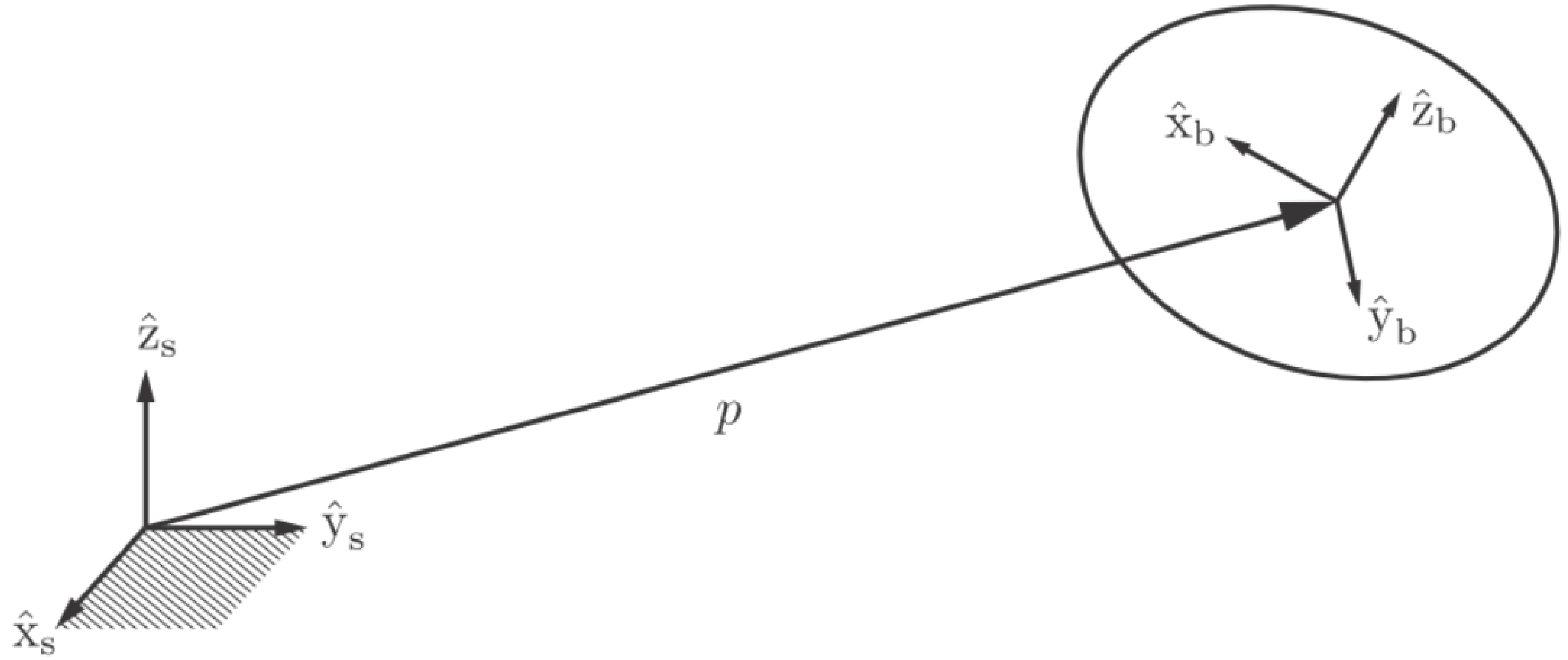
The frame {b} in {s} is given by (P, p), and the frame {c} in {b} is given by (Q, q). From these we can derive the frame {c} in {s}, described by (R, r). The numerical values of the vectors p, q, and r and the coordinate-axis directions of the three frames are evident from the grid of unit squares.

"Modern Robotics: Mechanics, Planning, and Control," Kevin M. Lynch and Frank C. Park, Cambridge University Press, 2017, ISBN 9781107156302
Chapter 3 Rigid-Body Motions

3D - Rigid-Body Motions

- All reference frames are right-handed
- The unit axes $\{\hat{x}, \hat{y}, \hat{z}\}$ alway satisfy $\hat{x} \times \hat{y} = \hat{z}$
- Unit axes of the fixed frame $\{\hat{x}_s, \hat{y}_s, \hat{z}_s\}$
- Unit axes of the body frame $\{\hat{x}_b, \hat{y}_b, \hat{z}_b\}$

$$
\begin{aligned}
\hat{\mathrm{x}}_\mathrm{b} &= r_{11}\hat{\mathrm{x}}_\mathrm{s} + r_{21}\hat{\mathrm{y}}_\mathrm{s} + r_{31}\hat{\mathrm{z}}_\mathrm{s} \\
\hat{\mathrm{y}}_\mathrm{b} &= r_{12}\hat{\mathrm{x}}_\mathrm{s} + r_{22}\hat{\mathrm{y}}_\mathrm{s} + r_{32}\hat{\mathrm{z}}_\mathrm{s} \\
\hat{\mathrm{z}}_\mathrm{b} &= r_{13}\hat{\mathrm{x}}_\mathrm{s} + r_{23}\hat{\mathrm{y}}_\mathrm{s} + r_{33}\hat{\mathrm{z}}_\mathrm{s}
\end{aligned}
$$



- ***p*** the vector from the fixed-frame origin to the body-frame origin. In terms of the fixed-frame coordinates:

$$
p = p_1\hat{\mathrm{x}}_\mathrm{s} + p_2\hat{\mathrm{y}}_\mathrm{s} + p_3\hat{\mathrm{z}}_\mathrm{s}
$$

- A description of the position and orientation of the rigid body is given by 12 parameters defining $p \in \boldsymbol{R}^3$ and $R \in \boldsymbol{R}^{3\times3}$ as:

$$
p = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix}, \qquad R = [\hat{\mathrm{x}}_\mathrm{b} \;\; \hat{\mathrm{y}}_\mathrm{b} \;\; \hat{\mathrm{z}}_\mathrm{b}] = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}
$$

"Modern Robotics: Mechanics, Planning, and Control," Kevin M. Lynch and Frank C. Park, Cambridge University Press, 2017, ISBN 9781107156302
Chapter 3 Rigid-Body Motions

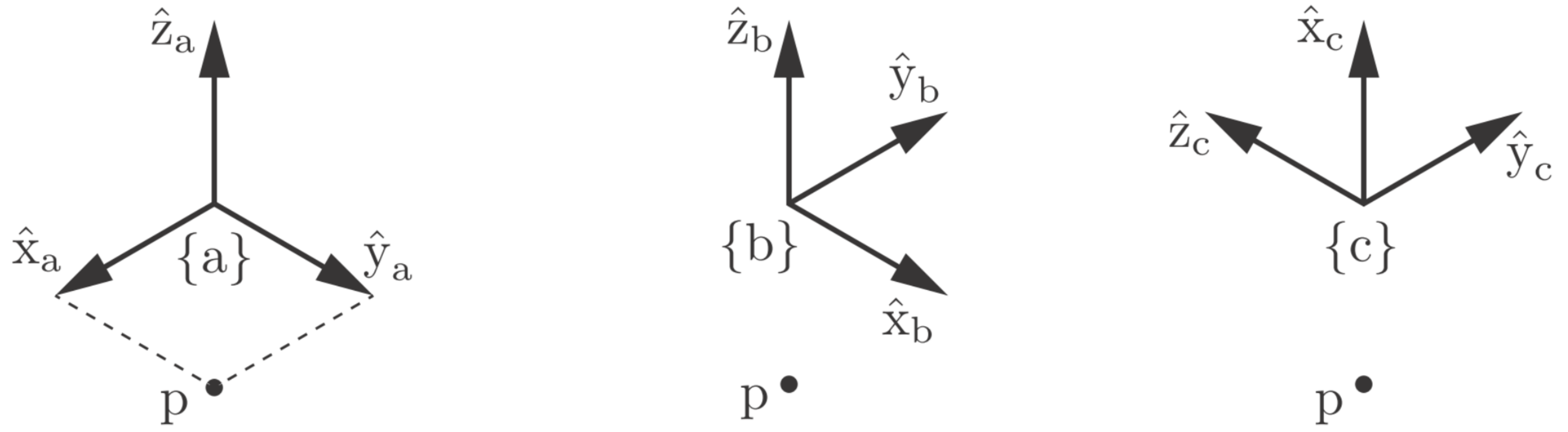
Rotation Matrices

Uses for rotation matrix R:

- To represent an orientation
- To change the reference frame in which a vector or a frame is represented
- To rotate a vector or a frame

A fixed space frame $\{ s \}$ is aligned with $\{ a \}$.
The orientations of the three frames relative to $\{ s \}$ can be written:

$$R_a = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad R_b = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad R_c = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix}$$



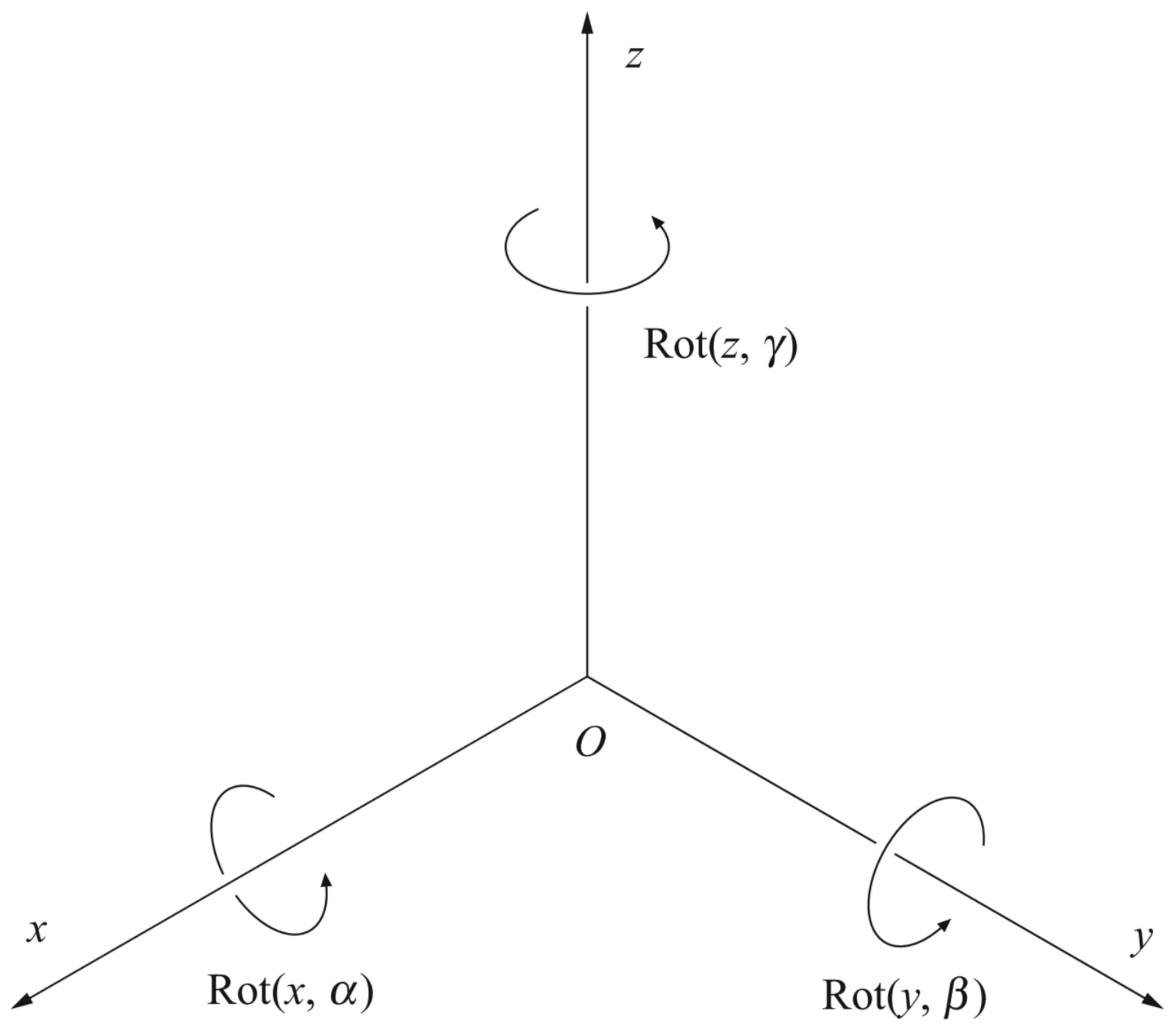
and the location of the point p in these frames can be written:

$$p_a = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \quad p_b = \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}, \quad p_c = \begin{bmatrix} 0 \\ -1 \\ -1 \end{bmatrix}$$

"Modern Robotics: Mechanics, Planning, and Control," Kevin M. Lynch and Frank C. Park, Cambridge University Press, 2017, ISBN 9781107156302
Chapter 3 Rigid-Body Motions

Rotation Matrices

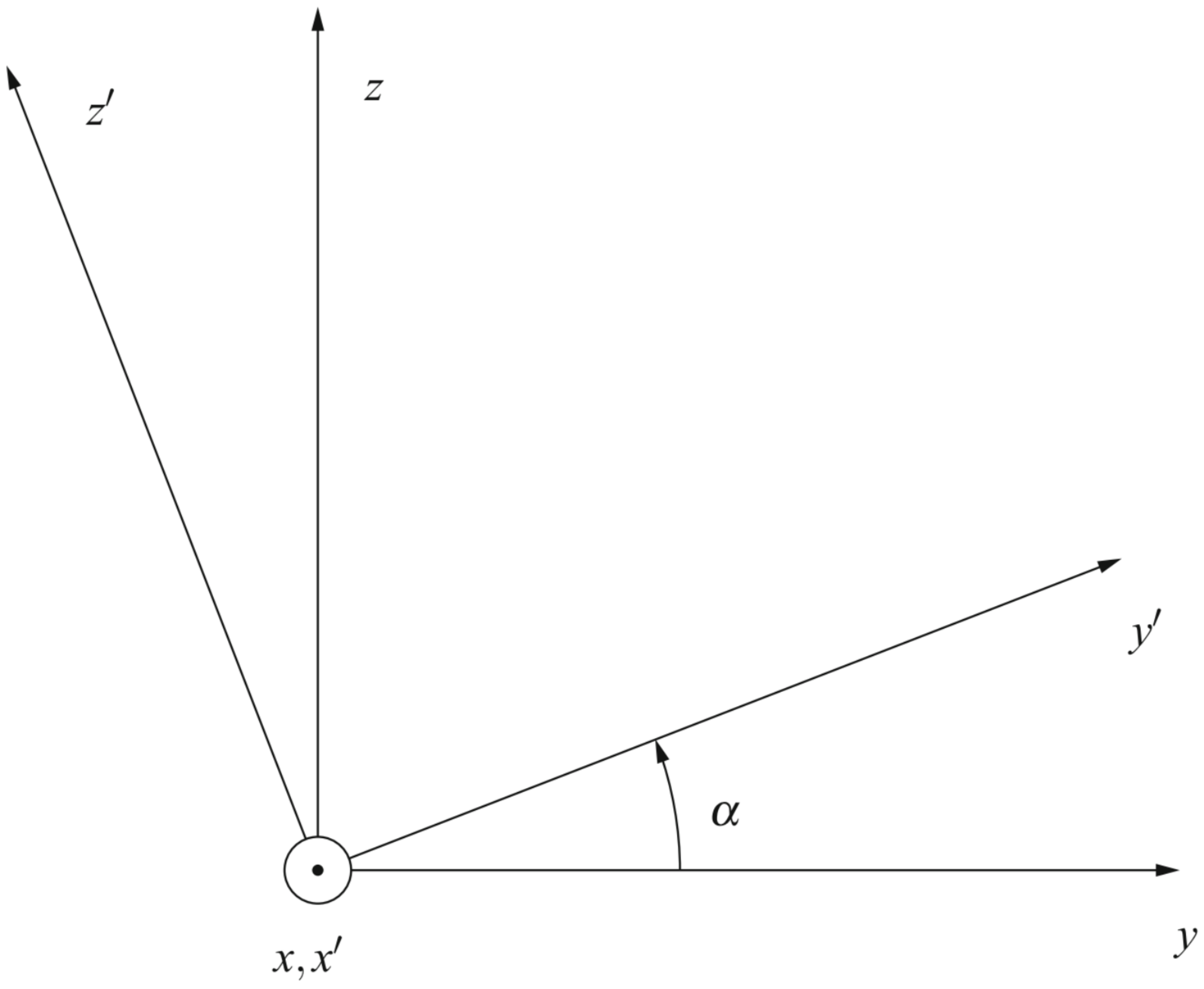
- Rotational displacements will be described in a right-handed rectangular coordinate frame, where the rotations around the three axes, as shown in Fig, are considered as positive.
- Positive rotations around the selected axis are counter-clockwise when looking from the positive end of the axis towards the origin O of the frame x–y–z.
- The positive rotation can be described also by the so called right hand rule, where the thumb is directed along the axis towards its positive end, while the fingers show the positive direction of the rotational displacement.
- The direction of running of athletes in a stadium is also an example of a positive rotation.



Right-hand rectangular frame with positive rotations

Rotation Matrices

- Let us first take a closer look at the rotation around the x axis. The coordinate frame $x' - y' - z'$ shown in Fig. was obtained by rotating the reference frame x–y–z in the positive direction around the x axis for the angle α.
- The axes x and x ′ are collinear.



Rotation around x axis

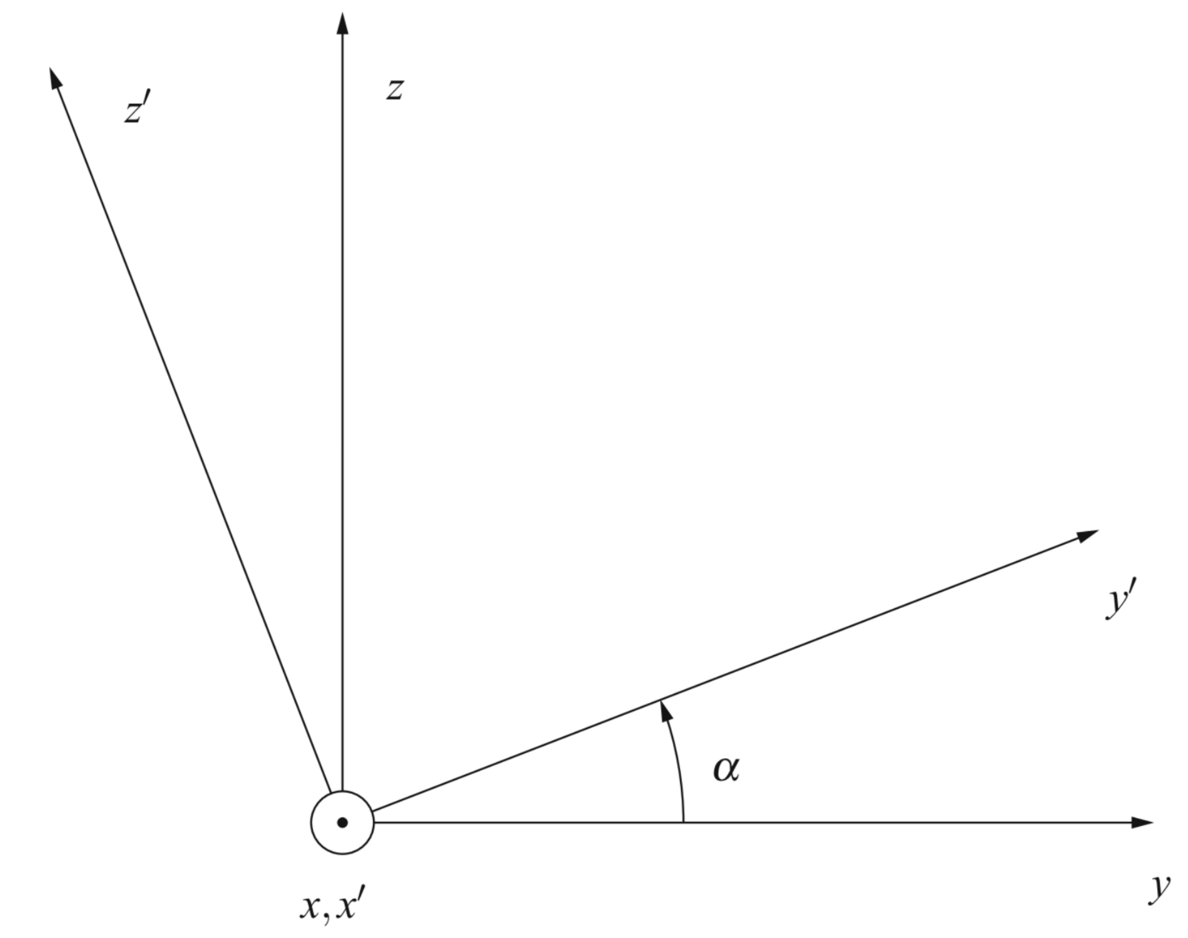
Rotation Matrices

- The rotational displacement is also described by a homogenous transformation matrix.
- The first three rows of the transformation matrix correspond to the x, y, and z axes of the reference frame, while the first three columns refer to the x'– y'– z' axes of the rotated frame.
- The upper left nine elements of the matrix H represent the 3 $\times$ 3 rotation matrix.
- The elements of the rotation matrix are cosines of the angles between the axes given by the corresponding column and row

$$
Rot(x,\alpha) = \begin{array}{c c}
\begin{array}{cccc} \quad x' \quad & \quad y' \quad & \quad z' \quad & \end{array} & \\
\begin{bmatrix}
\cos 0^\circ & \cos 90^\circ & \cos 90^\circ & 0 \\
\cos 90^\circ & \cos\alpha & \cos(90^\circ + \alpha) & 0 \\
\cos 90^\circ & \cos(90^\circ - \alpha) & \cos\alpha & 0 \\
0 & 0 & 0 & 1
\end{bmatrix} & \begin{array}{c} x \\ y \\ z \\ \end{array}
\end{array}
$$

$$
= \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & \cos\alpha & -\sin\alpha & 0 \\
0 & \sin\alpha & \cos\alpha & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}.
$$

Rotation Matrices



Rotation around x axis

$$Rot(x,\alpha) = \begin{array}{c} \\ \\ \\ \\ \\ \end{array}\begin{array}{cccc} x' & y' & z' & \\ \end{array}$$

$$Rot(x,\alpha) = \begin{array}{cccc|c} x' & y' & z' & & \\ \cos 0^\circ & \cos 90^\circ & \cos 90^\circ & 0 & x \\ \cos 90^\circ & \cos\alpha & \cos(90^\circ+\alpha) & 0 & y \\ \cos 90^\circ & \cos(90^\circ-\alpha) & \cos\alpha & 0 & z \\ 0 & 0 & 0 & 1 & \end{array}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

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

- To become more familiar with rotation matrices, we shall derive the matrix describing a rotation around the *y* axis.
- The collinear axes are *y* and *y’*

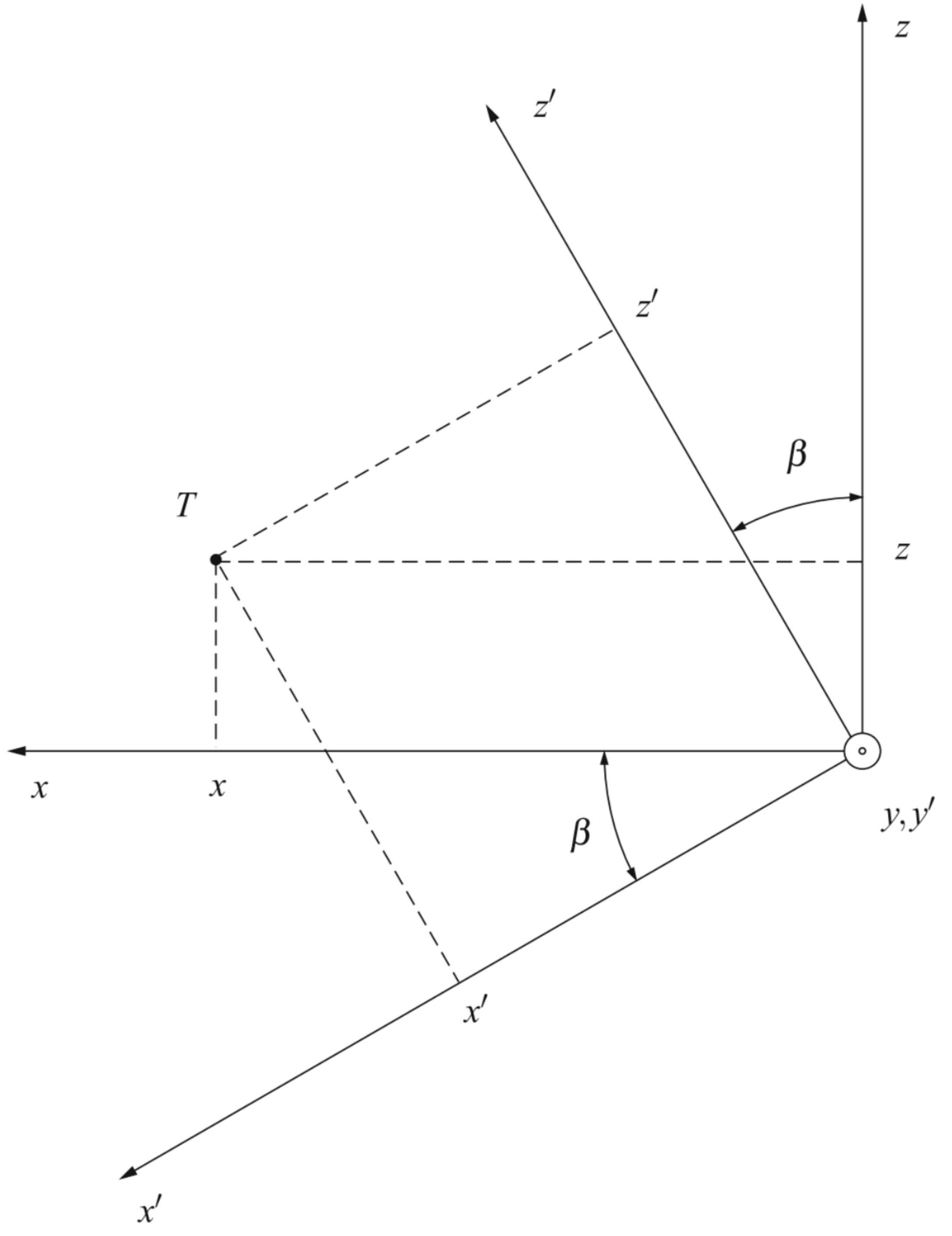
$$y = y'$$

- By considering the similarity of triangles in Fig., it is not difficult to derive the following two equations

$$x = x' \cos\beta + z' \sin\beta$$
$$z = -x' \sin\beta + z' \cos\beta.$$

- In the matrix form

$$Rot(y,\beta) = \begin{array}{c} \begin{array}{ccc} x' & y' & z' \end{array} \\ \left[\begin{array}{cccc} \cos\beta & 0 & \sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{array}\right] \begin{array}{c} x \\ y \\ z \\ \end{array} \end{array} \quad Rot(z,\gamma) = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 & 0 \\ \sin\gamma & \cos\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$



Rotation around *y* axis

Rotation Matrices – Numerical Example

- We wish to determine the vector w, which is obtained by rotating the vector

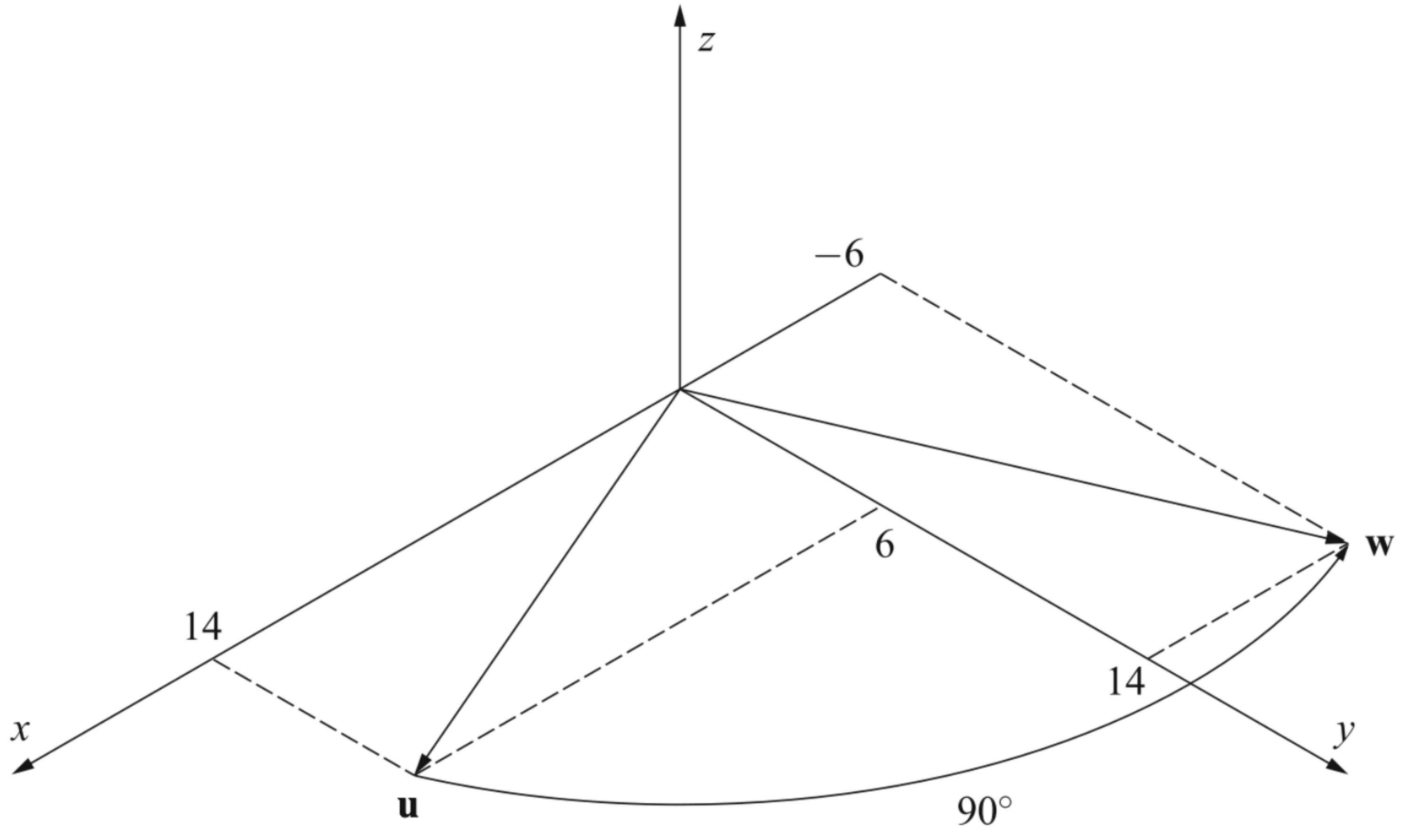
$$\boldsymbol{u} = 14\boldsymbol{i} + 6\boldsymbol{j} + 0\boldsymbol{k}$$

for 90° in the counter clockwise (i.e., positive) direction around the *z* axis.

- As $\cos(90°) = 0$ and $\sin(90°) = 1$, it is not difficult to determine the matrix describing $Rot(z, 90°)$ and multiplying it by the vector $\boldsymbol{u}$

$$\boxed{\mathbf{w} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 14 \\ 6 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -6 \\ 14 \\ 0 \\ 1 \end{bmatrix}}$$

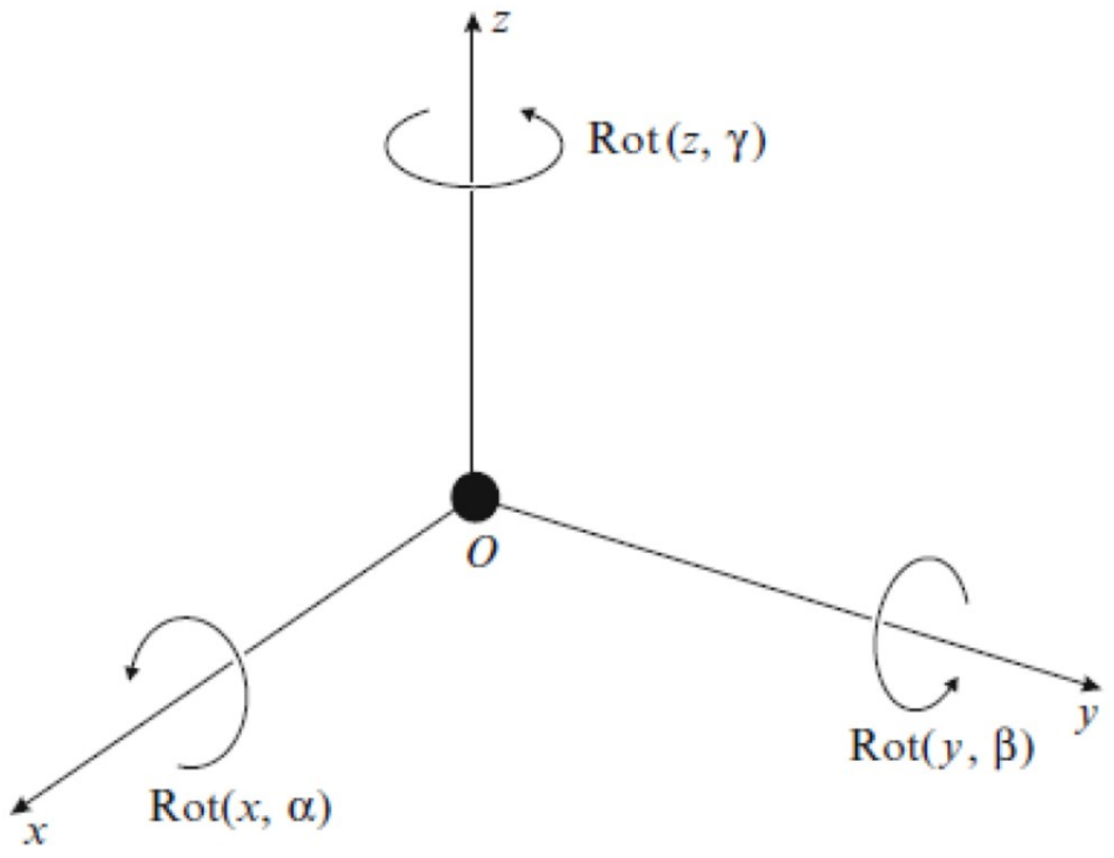
- The graphical presentation of rotating the vector $\boldsymbol{u}$ around the z axis is shown in Fig.



Example of rotational transformation

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

Fundamental rotation matrices



$$R_{x,\alpha} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\alpha) & -sin(\alpha) \\ 0 & sin(\alpha) & cos(\alpha) \end{bmatrix}$$ Rotation around the X axis

$$R_{y,\phi} = \begin{bmatrix} cos(\phi) & 0 & sin(\phi) \\ 0 & 1 & 0 \\ -sin(\phi) & 0 & cos(\phi) \end{bmatrix}$$ Rotation around the Y axis

$$R_{z,\vartheta} = \begin{bmatrix} cos(\vartheta) & -sin(\vartheta) & 0 \\ sin(\vartheta) & cos(\vartheta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$ Rotation around the Z axis

The fundamental rotation matrices can be multiplied to represent a sequence of rotations around the main axes of the reference frame: $R = R_{x,\alpha}\, R_{y,\phi}\, R_{z,\vartheta}$

Robotics – 2019 Springer Second Edition
Authors: Mihelj, M., Bajd, T., Ude, A., Lenarčič, J., Stanovnik, A., Munih, M., Rejc, J., Šlajpah, S.
Chapter 2 Homogenous Transformation Matrices

Homogeneous Transformation Matrices

- We now consider representations for the combined orientation and position of a rigid body. A natural choice would be to use a rotation matrix $\boldsymbol{R}$ to represent the orientation of the body frame $\{b\}$ in the fixed frame $\{s\}$ and a vector $\boldsymbol{p}$ to represent the origin of $\{b\}$ in $\{s\}$. Rather than identifying $\boldsymbol{R}$ and $\boldsymbol{p}$ separately, we package them into a single Transformation matrix.

$$T = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_1 \\ r_{21} & r_{22} & r_{23} & p_2 \\ r_{31} & r_{32} & r_{33} & p_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- The inverse of a transformation matrix T is also a transformation matrix:

$$T^{-1} = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} R^{\mathrm{T}} & -R^{\mathrm{T}}p \\ 0 & 1 \end{bmatrix}$$

- The product of two transformation matrices is also a transformation matrix. The multiplication is *associative* but *not (generally) commutative:*

$$(T_1T_2)T_3 = T_1(T_2T_3) \qquad\qquad T_1T_2 \neq T_2T_1$$

"Modern Robotics: Mechanics, Planning, and Control," Kevin M. Lynch and Frank C. Park, Cambridge University Press, 2017, ISBN 9781107156302
Chapter 3 Rigid-Body Motions

Homogeneous Rotation Matrices

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

Rotation around the X 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 Y axis

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

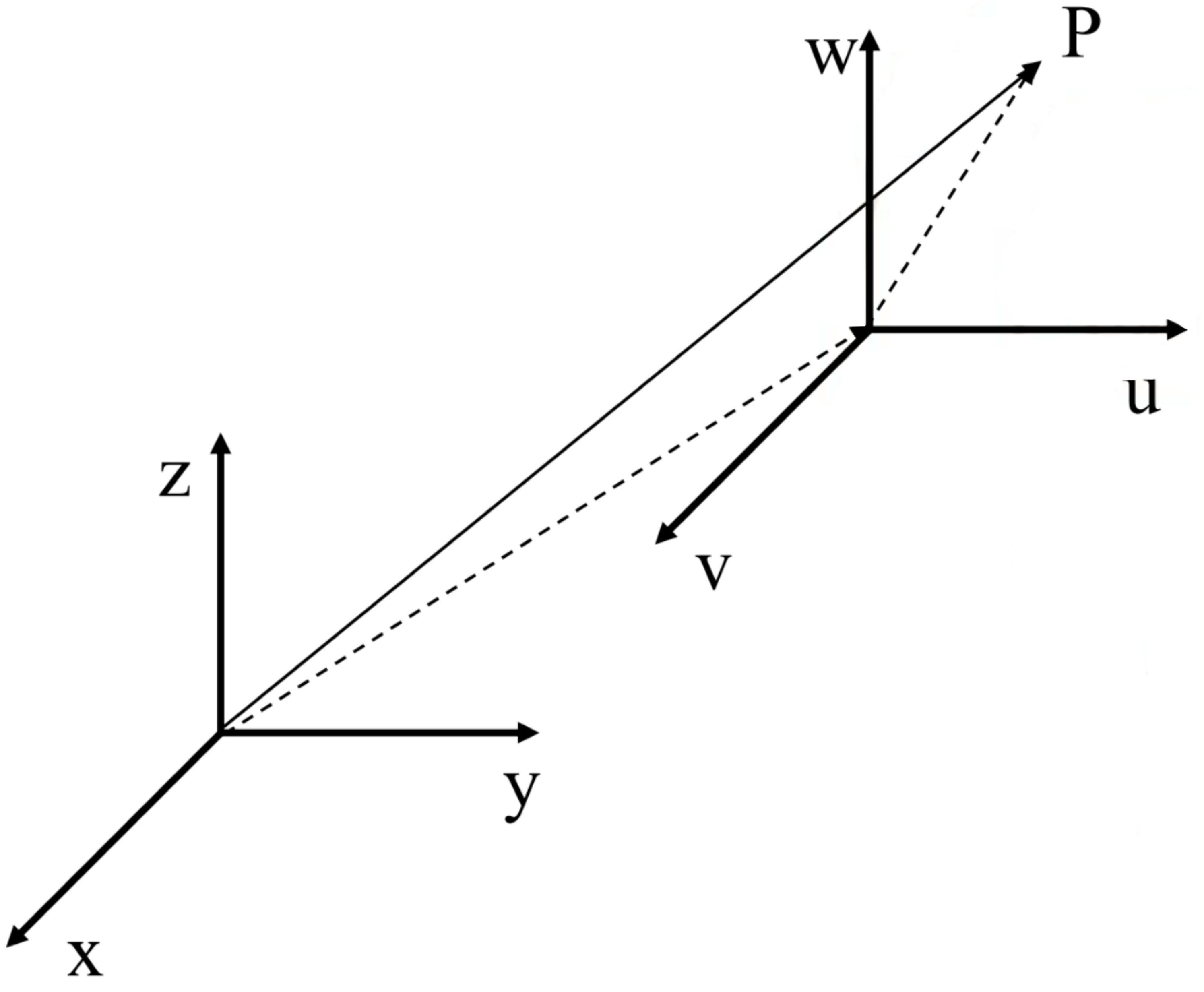
Rotation around the Z axis

"Modern Robotics: Mechanics, Planning, and Control," Kevin M. Lynch and Frank C. Park, Cambridge University Press, 2017, ISBN 9781107156302
Chapter 3 Rigid-Body Motions

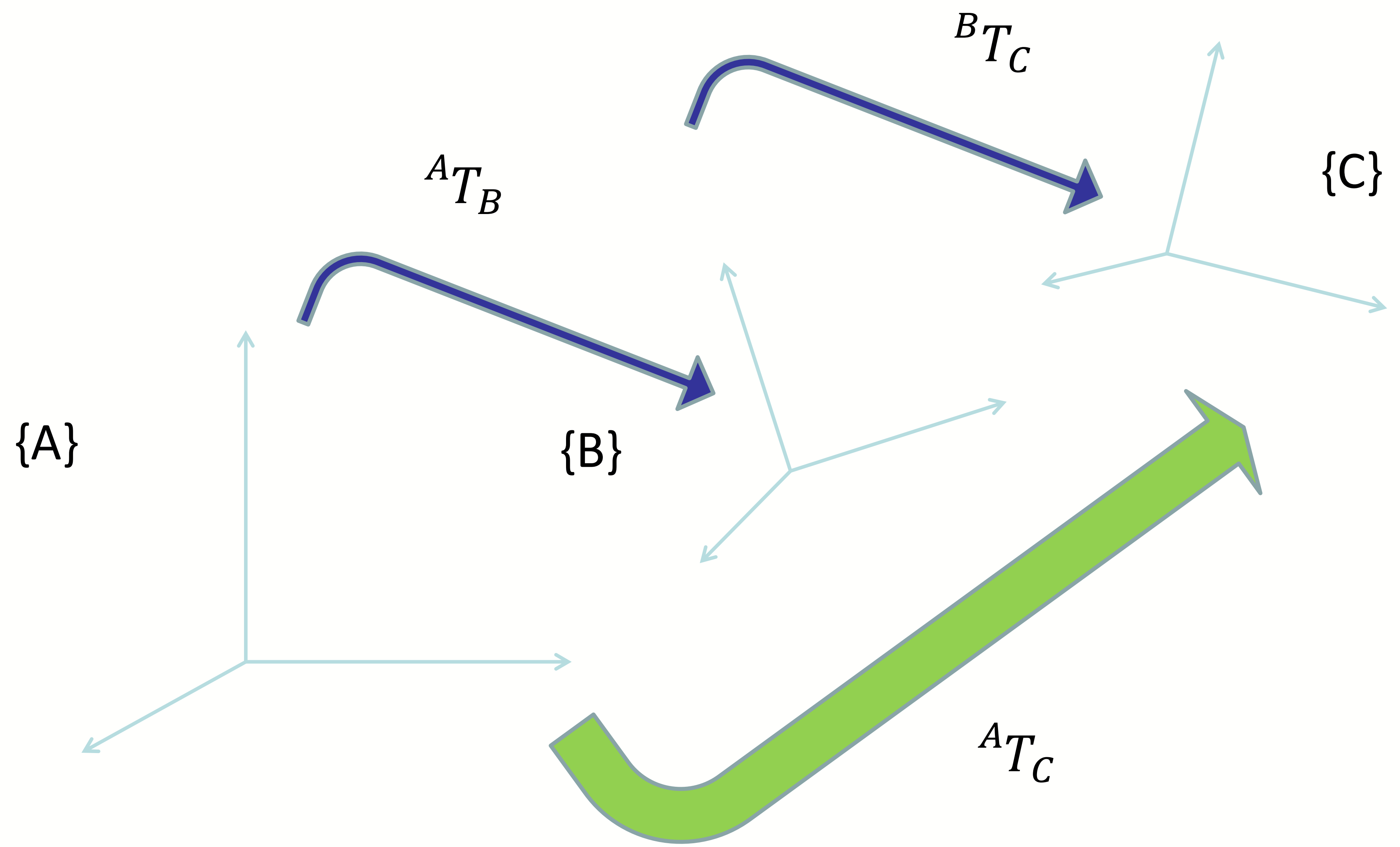
Fundamental homogeneous translation matrix

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

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



Homogeneous Transformations Composition…



$$^{A}T_{C} = {}^{A}T_{B}\,{}^{B}T_{C}$$

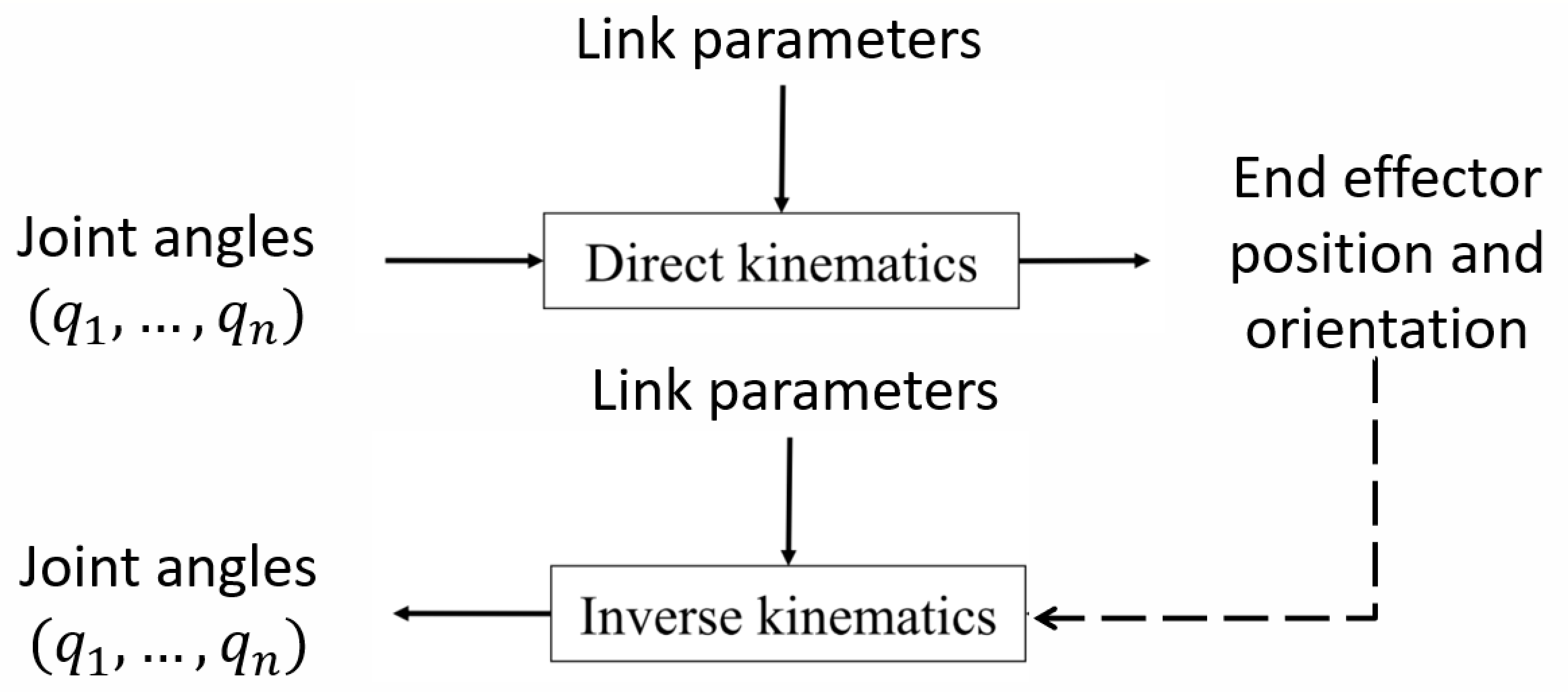
Robot arm kinematics

Direct kinematics:

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

Inverse kinematics:

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



Direct/Inverse kinematics problem

- **Direct**
  - 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

$$x = K(q) \quad x\ unknown, q\ known$$

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

Direct/Inverse kinematics problem

- **Inverse**
  - 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

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

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

Kinematics redundancy

Number of DOFs higher than the number of variables needed for characterizing a task ↔The operational space size is smaller than the joint space size.

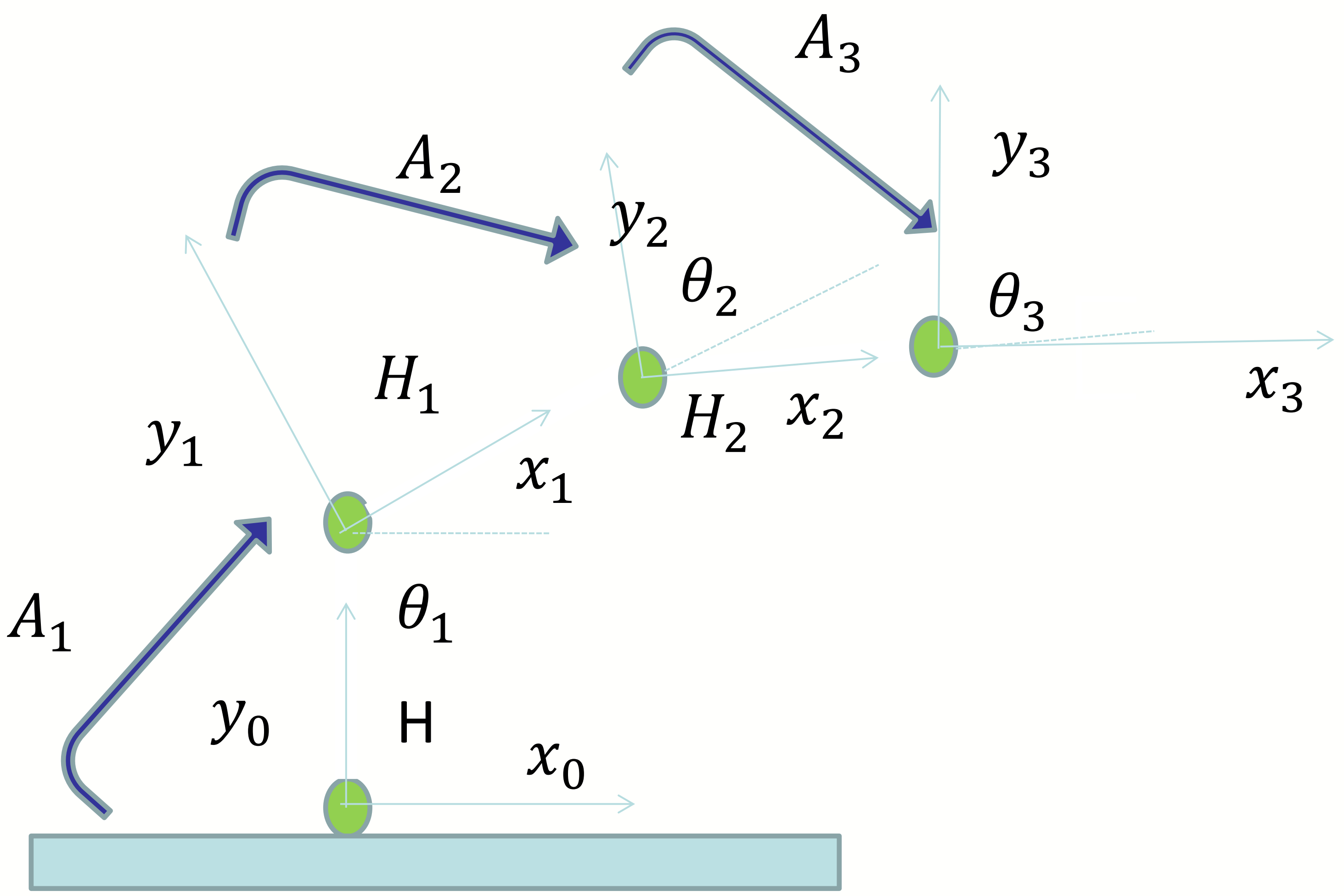
The number of redundancy degrees is $R = N - M$

Advantages: *multiple solutions*
Disadvantages: *computing and control complexity*

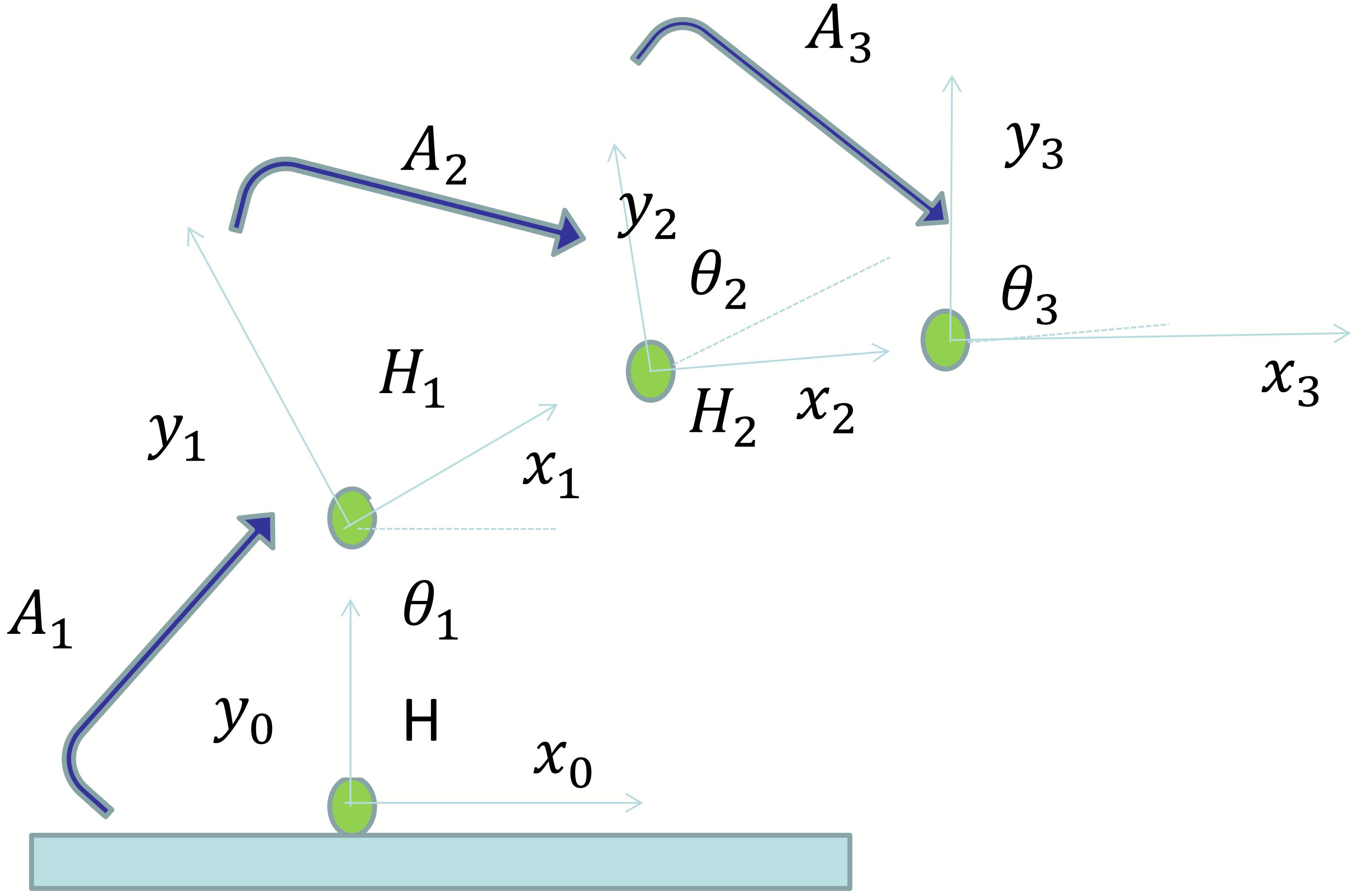
Inverse kinematics problem

- The 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

Exercise 2D – Direct kinematics



Exercise 2D – Direct kinematics



$$A_1 = \begin{pmatrix} \cos\theta_1 & -sin\theta_1 & 0 \\ sin\theta_1 & \cos\theta_1 & H \\ 0 & 0 & 1 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} \cos\theta_2 & -sin\theta_2 & H_1 \\ sin\theta_2 & \cos\theta_2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$A_3 = \begin{pmatrix} \cos\theta_3 & -sin\theta_3 & H_2 \\ sin\theta_3 & \cos\theta_3 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$T_3 = A_1\,A_2\,A_3$$

$$T_3 = \begin{pmatrix} \cos(\theta_1 + \theta_2 + \theta_3) & -sin(\theta_1 + \theta_2 + \theta_3) & \cos(\theta_1 + \theta_2)H_2 + \cos(\theta_1)H_1 \\ sin(\theta_1 + \theta_2 + \theta_3) & \cos(\theta_1 + \theta_2 + \theta_3) & \mathrm{H} + \sin(\theta_1 + \theta_2)H_2 + \sin(\theta_1)H_1 \\ 0 & 0 & 1 \end{pmatrix}$$

Example of transformation of a reference frame

- In this section we shall deal with the pose and the displacement of rectangular frames.
- Here we see that a homogenous transformation matrix describes either the pose of a frame with respect to a reference frame, or it represents the displacement of a frame into a new pose.
- In the first case the upper left 3×3 matrix represents the orientation of the object, while the right-hand 3×1 column describesits position (e.g., the position of its center of mass).
- The last row of the homogenous transformation matrix will be always represented by [0 0 0 1].
- In the case of object displacement, the upper left matrix corresponds to rotation and the right-hand column corresponds to translation of the object.
- We shall examine both cases through simple examples.

$$\mathbf{H} = Trans(8, -6, 14)Rot(y, 90^\circ)Rot(z, 90^\circ)$$

$$= \begin{bmatrix} 1 & 0 & 0 & 8 \\ 0 & 1 & 0 & -6 \\ 0 & 0 & 1 & 14 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 8 \\ 1 & 0 & 0 & -6 \\ 0 & 1 & 0 & 14 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Example of transformation of a reference frame

- Let us first clear up the meaning of the homogenous transformation matrix describing the pose of an arbitrary frame with respect to the reference frame.
- Let us consider the following product of homogenous matrices which gives a new homogenous transformation matrix **H**
- When defining the homogenous matrix representing rotation, we learned that the first three columns describe the rotation of the frame $x' - y' - z'$ with respect to the reference frame $x - y - z$
- The fourth column represents the position of the origin of the frame $x' - y' - z'$ with respect to the reference frame $x - y - z$.

$$
\mathbf{H} = Trans(8, -6, 14)Rot(y, 90^\circ)Rot(z, 90^\circ)
$$

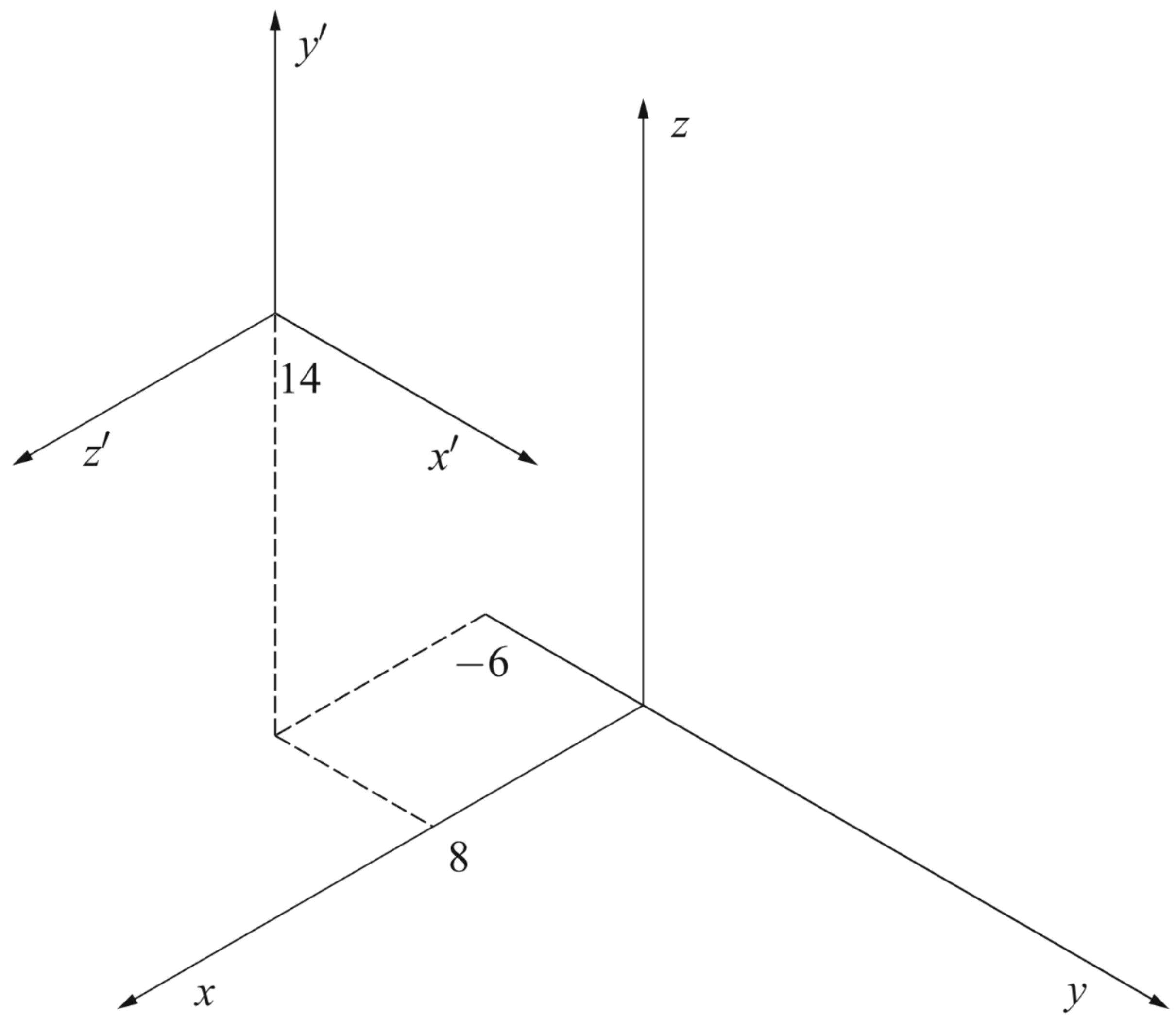
$$
= \begin{bmatrix} 1 & 0 & 0 & 8 \\ 0 & 1 & 0 & -6 \\ 0 & 0 & 1 & 14 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
= \begin{array}{cc} \begin{array}{cccc} x' & y' & z' & \end{array} & \\ \begin{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} & \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} & \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} & \begin{matrix} 8 \\ -6 \\ 14 \end{matrix} \\ 0 & 0 & 0 & 1 \end{bmatrix} & \begin{matrix} x \\ y \\ z \\ \end{matrix} \end{array}
$$

Example of transformation of a reference frame

- With this knowledge we can represent graphically the frame $x'-y'-z'$ described by the homogenous transformation matrix, relative to the reference frame $x-y-z$.
- The x' axis points in the direction of y axis of the reference frame, the y' axis is in the direction of the z axis, and the z' axis is in the x direction.

$$\mathbf{H} = Trans(8, -6, 14)Rot(y, 90^\circ)Rot(z, 90^\circ)$$

$$= \begin{bmatrix} 1 & 0 & 0 & 8 \\ 0 & 1 & 0 & -6 \\ 0 & 0 & 1 & 14 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{array}{cc} \begin{array}{cccc} x' & y' & z' & \end{array} & \\ \begin{bmatrix} 0 & 0 & 1 & 8 \\ 1 & 0 & 0 & -6 \\ 0 & 1 & 0 & 14 \\ 0 & 0 & 0 & 1 \end{bmatrix} & \begin{array}{c} x \\ y \\ z \\ \ \end{array} \end{array}$$



The pose of an arbitrary frame x'–y'–z' with respect to the reference frame x–y–z

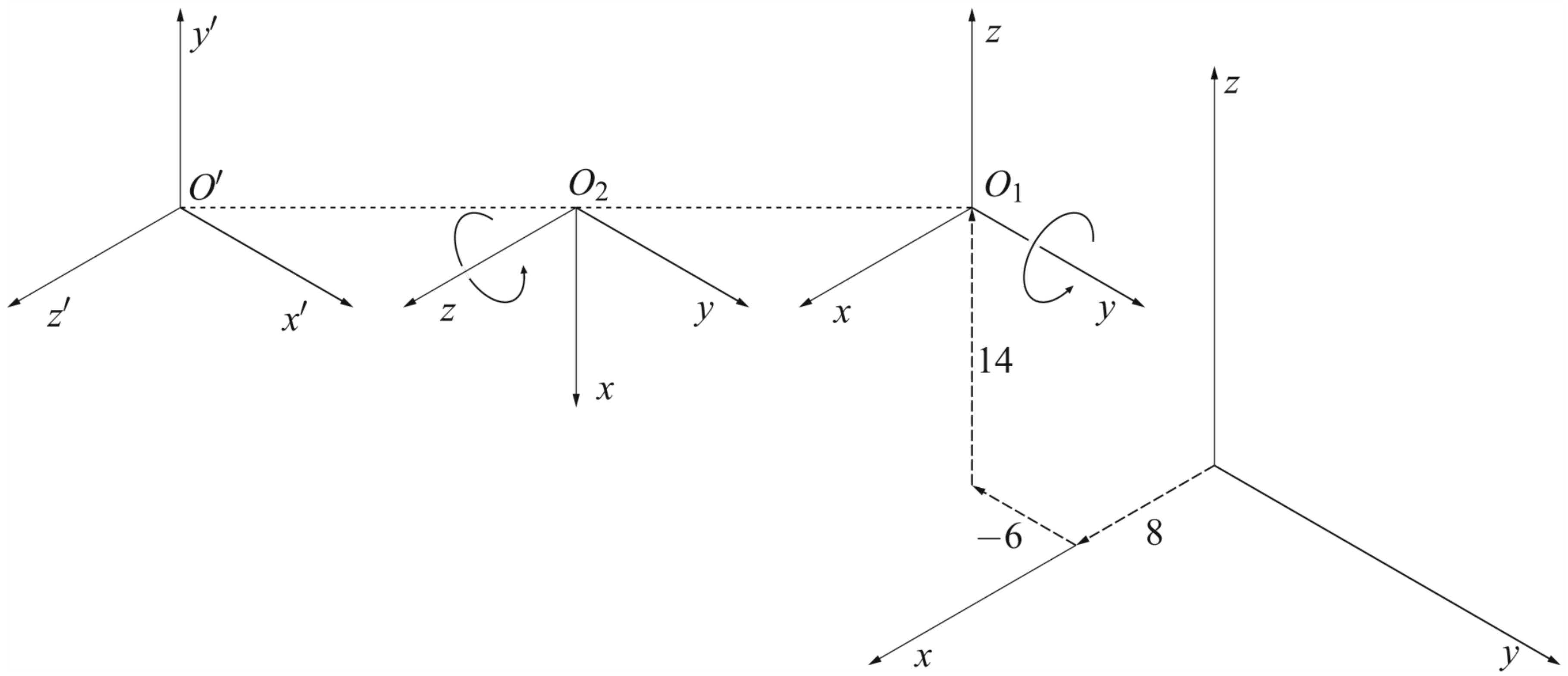
Robotics – 2019 Springer Second Edition
Authors: Mihelj, M., Bajd, T., Ude, A., Lenarčič, J., Stanovnik, A., Munih, M., Rejc, J., Šlajpah, S.
Chapter 2 Homogenous Transformation Matrices

Example of transformation of a reference frame

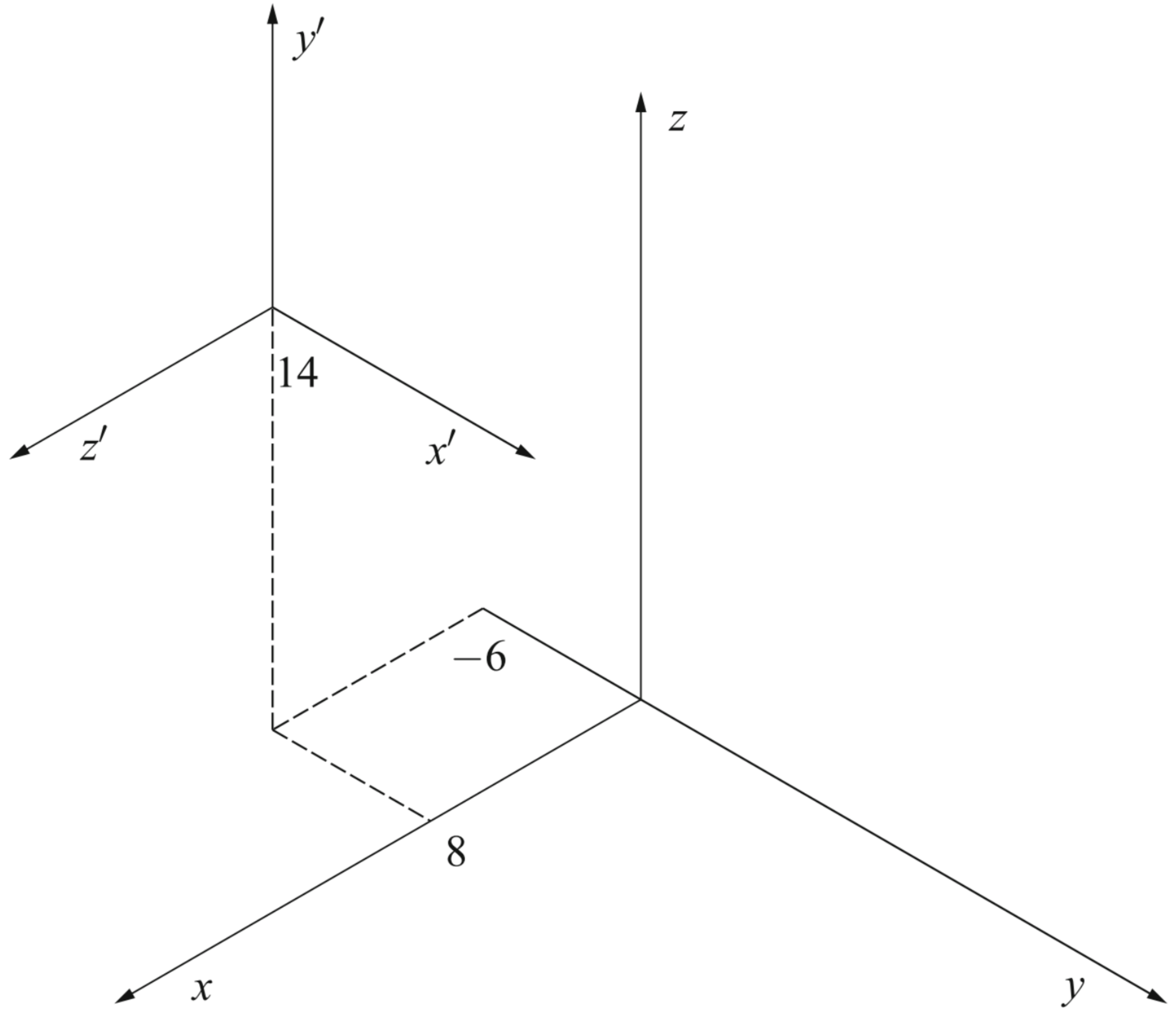
- The reference frame is first translated into the point (8,−6, 14), afterwards it is rotated for 90° around the new y axis and finally it is rotated for 90° around the newest z axis

$$\mathbf{H} = Trans(8, -6, 14)Rot(y, 90^\circ)Rot(z, 90^\circ)$$

$$= \begin{bmatrix} 1 & 0 & 0 & 8 \\ 0 & 1 & 0 & -6 \\ 0 & 0 & 1 & 14 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{array}{c c} \begin{array}{cccc} x' & y' & z' & \end{array} & \\ \begin{bmatrix} 0 & 0 & 1 & 8 \\ 1 & 0 & 0 & -6 \\ 0 & 1 & 0 & 14 \\ 0 & 0 & 0 & 1 \end{bmatrix} & \begin{array}{c} x \\ y \\ z \\ \\ \end{array} \end{array}$$



Displacement of the reference frame into a new pose (from right to left). The origins O_1, O_2 and O' are in the same point



The pose of an arbitrary frame x'–y'–z' with respect to the reference frame x–y–z

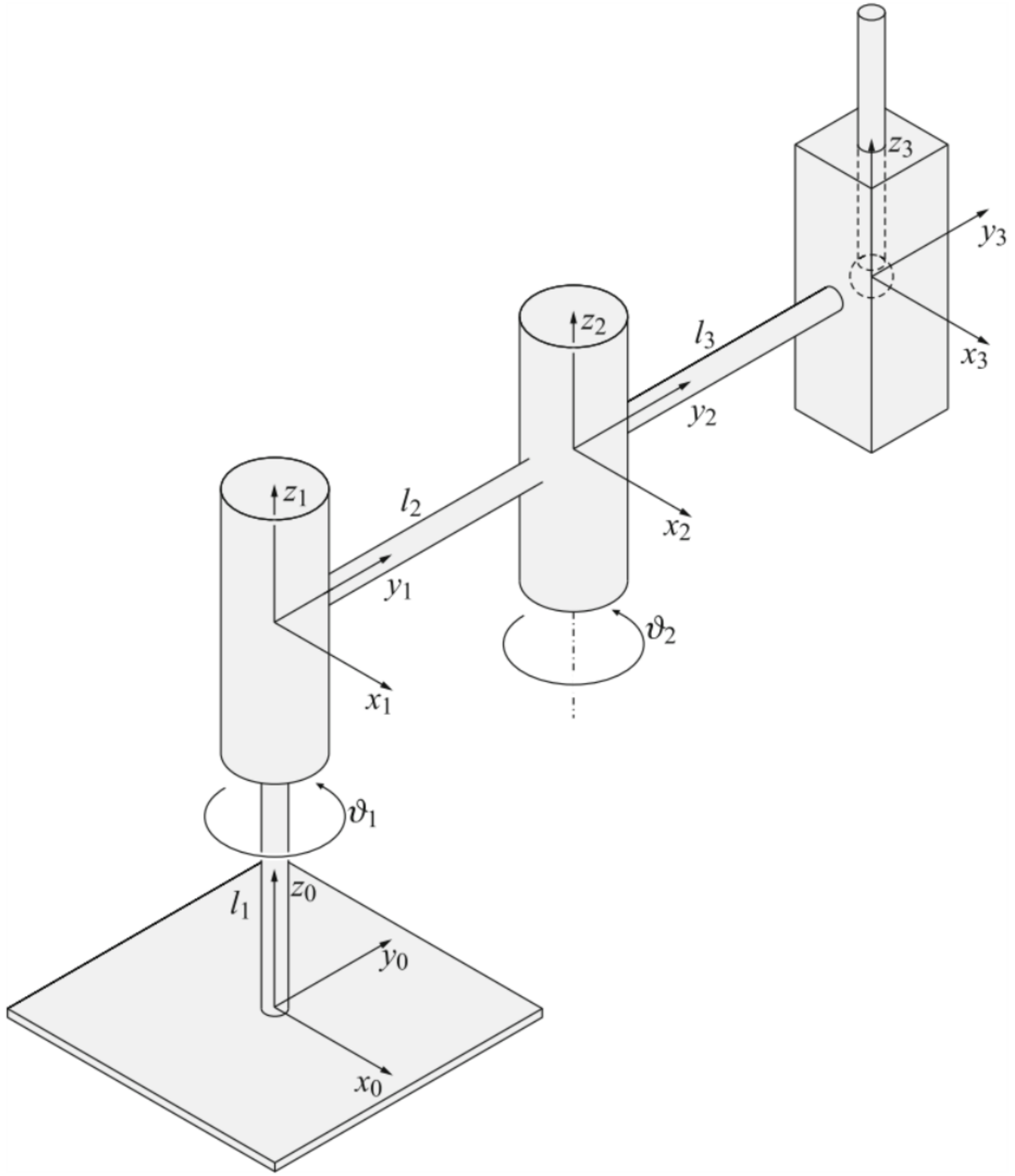
Geometric manipulator model

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

The matrices $^{0}\boldsymbol{H_1}$, $^{1}\boldsymbol{H_2}$, $^{2}\boldsymbol{H_3}$ 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 it is evident that the 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 $cos(\vartheta_1) = c1$



The SCARA robot manipulator in the initial pose

Geometric manipulator model

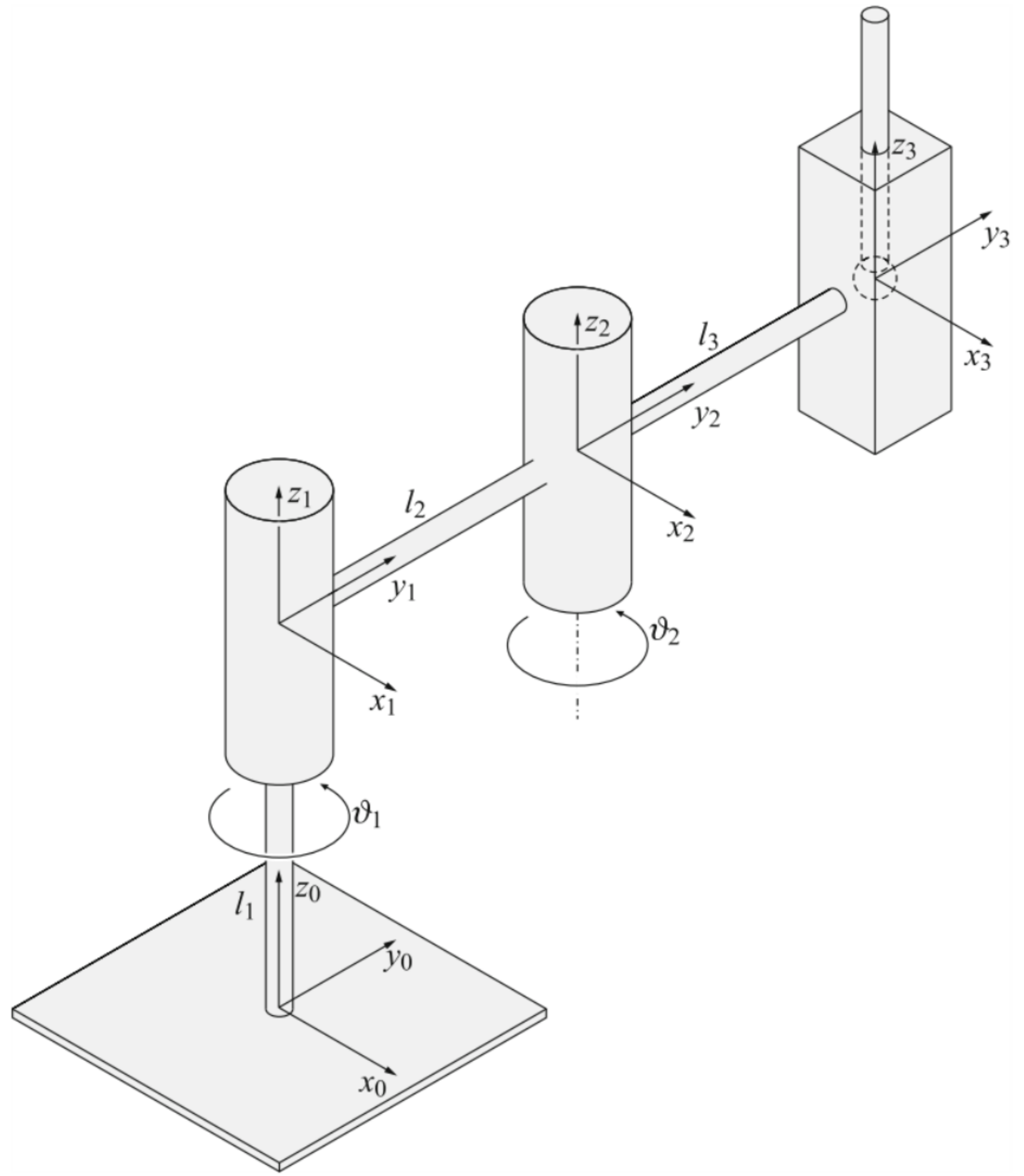
$$^{0}\mathbf{H}_3 = (^{0}\mathbf{H}_1\mathbf{D}_1)\cdot(^{1}\mathbf{H}_2\mathbf{D}_2)\cdot(^{2}\mathbf{H}_3\mathbf{D}_3)$$

In 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 SCARA robot manipulator in the initial pose

Geometric manipulator model

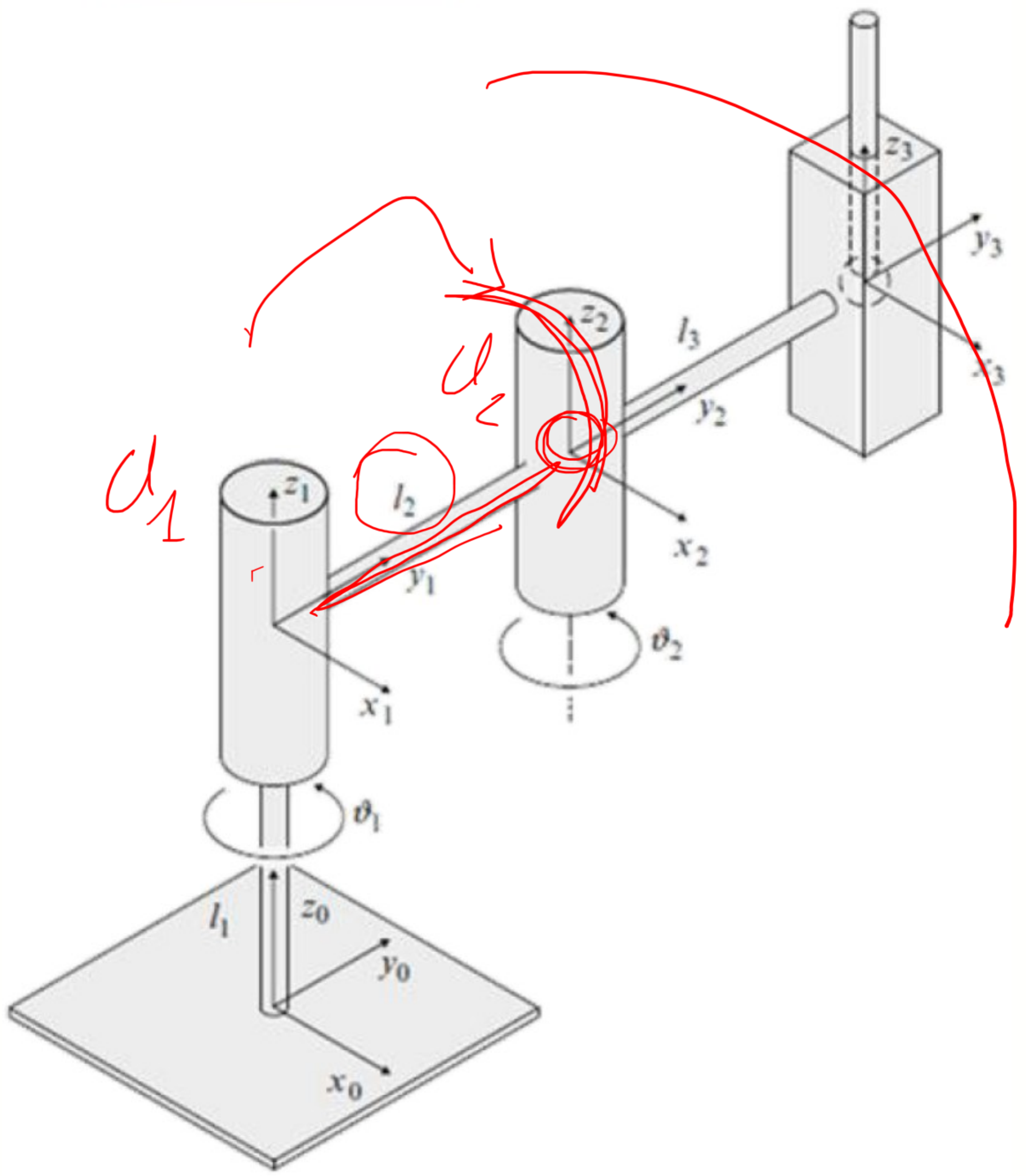
$$^0\mathbf{H}_3 = (^0\mathbf{H}_1\mathbf{D}_1) \cdot (^1\mathbf{H}_2\mathbf{D}_2) \cdot (^2\mathbf{H}_3\mathbf{D}_3)$$

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

$$^0\mathbf{H}_3 = \begin{bmatrix} c12 & -s12 & 0 & -l_3 s12 - l_2 s1 \\ s12 & c12 & 0 & l_3 c12 + l_2 c1 \\ 0 & 0 & 1 & l_1 - d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

When multiplying the three matrices the following abbreviation was introduced:

$$c12 = \cos(\vartheta_1 + \vartheta_2) = c1c2 - s1s2$$
$$s12 = sin(\vartheta_1 + \vartheta_2) = s1c2 + c1s2$$



The Universal Robot Description Format (URDF)

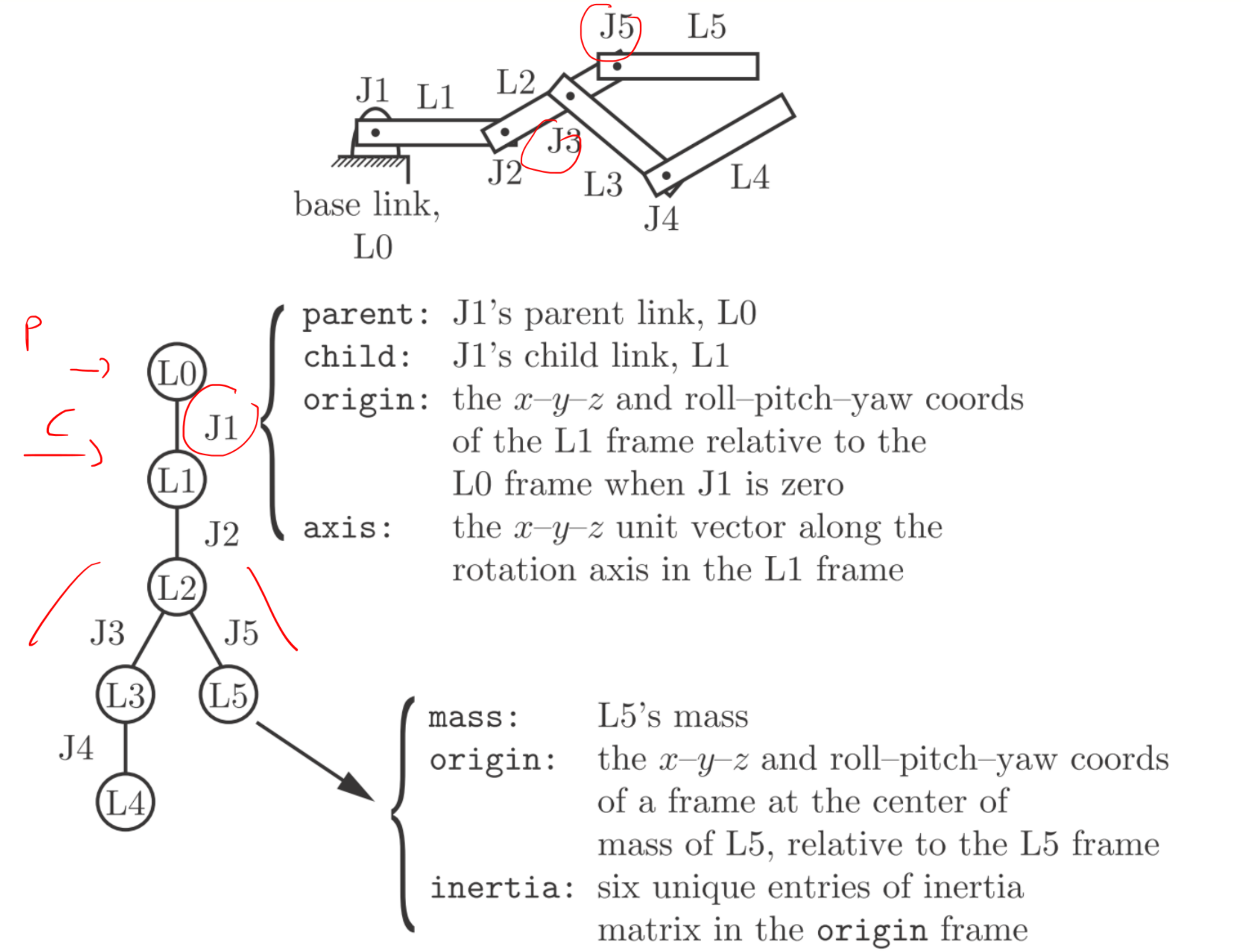
The Universal Robot Description Format (URDF) is an XML (eXtensible Markup Language) file format used by the Robot Operating System (ROS) to describe the kinematics, inertial properties, and link geometry of robots.
A URDF file describes the joints and links of a robot:

- **Joints**. Joints connect two links: a parent link and a child link.
  - ***Joint types*** include prismatic, revolute (including joint limits), continuous (revolute without joint limits), and fixed (a virtual joint that does not permit any motion).
  - Each joint has an ***origin frame*** that defines the position and orientation of the child link frame relative to the parent link frame when the joint variable is zero.
  - The ***origin*** is on the joint's axis. Each joint has an axis 3-vector, a unit vector expressed in the child link's frame, in the direction of positive rotation for a revolute joint or positive translation for a prismatic joint.

- **Links**. the links define the mass properties.
  - The elements of a link include its ***mass***;
  - An ***origin frame*** that defines the position and orientation of a frame at the link's center of mass relative to the link's joint frame
  - An ***inertia matrix***, relative to the link's center of mass frame

The Universal Robot Description Format (URDF)

A URDF file can represent any robot with a tree structure including serial-chain robot arms and robot hands

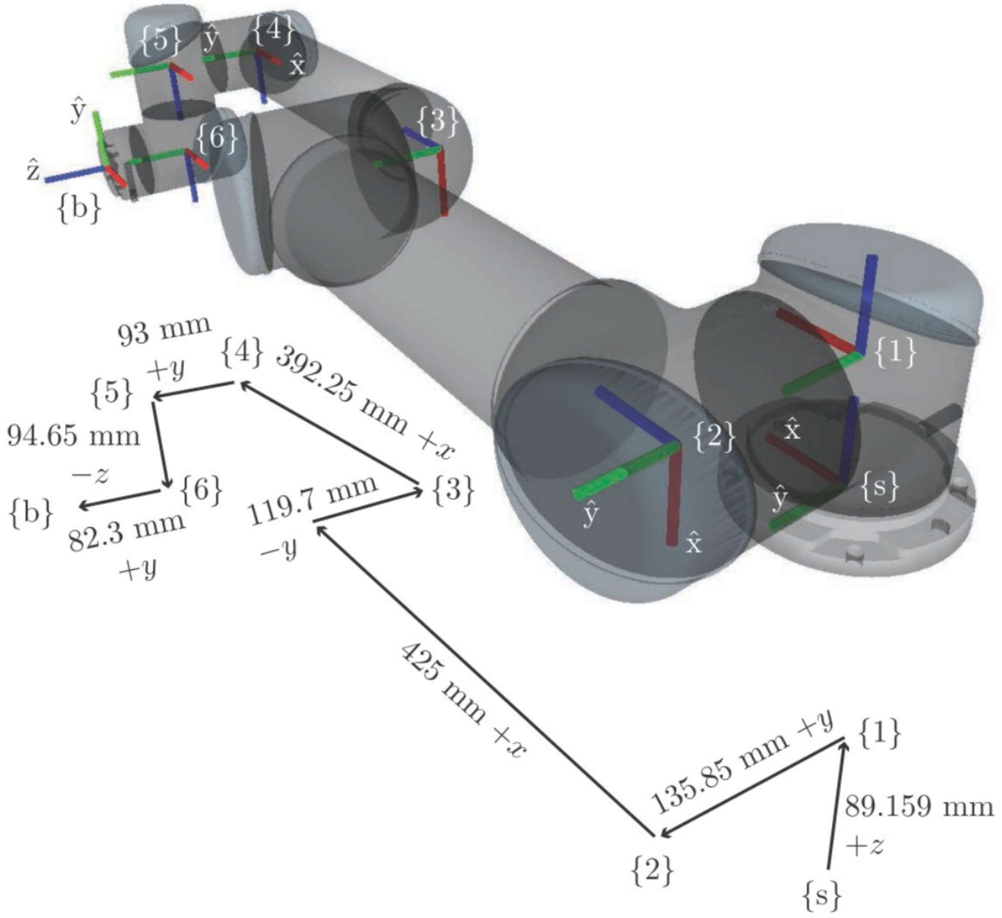


A five-link robot represented as a tree, where the nodes of the tree are the links and the edges of the tree are the joints.

URDF - Example

The UR5 URDF file (kinematics and inertial properties only).

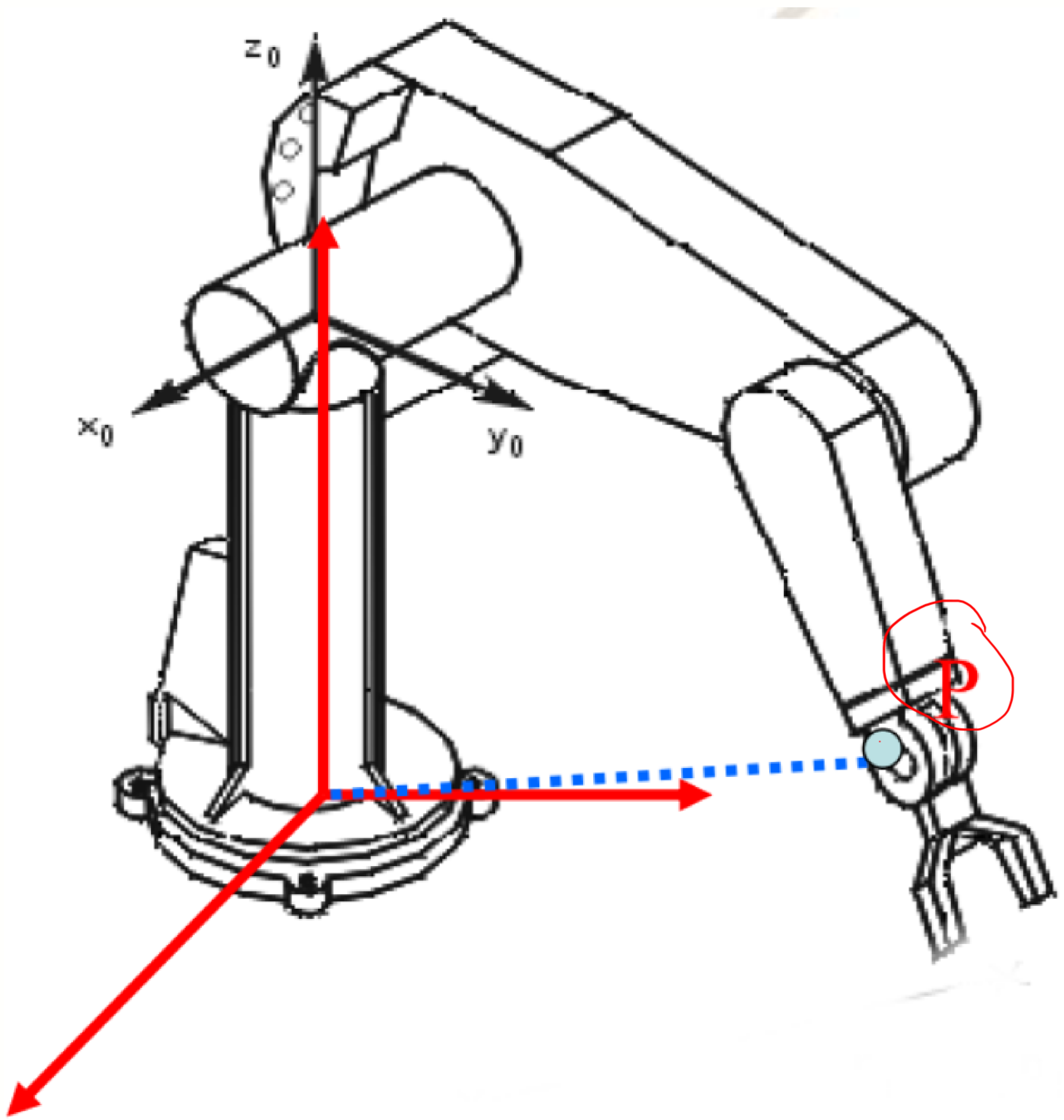
```
<?xml version="1.0" ?>
<robot name="ur5">

<!-- ********** KINEMATIC PROPERTIES (JOINTS) ********** -->
  <joint name="world_joint" type="fixed">
    <parent link="world"/>
    <child link="base_link"/>
    <origin rpy="0.0 0.0 0.0" xyz="0.0 0.0 0.0"/>
  </joint>
  <joint name="joint1" type="continuous">
    <parent link="base_link"/>
    <child link="link1"/>
    <origin rpy="0.0 0.0 0.0" xyz="0.0 0.0 0.089159"/>
    <axis xyz="0 0 1"/>
  </joint>
  <joint name="joint2" type="continuous">
    <parent link="link1"/>
    <child link="link2"/>
    <origin rpy="0.0 1.570796325 0.0" xyz="0.0 0.13585 0.0"/>
    <axis xyz="0 1 0"/>
  </joint>
  <joint name="joint3" type="continuous">
    <parent link="link2"/>
    <child link="link3"/>
    <origin rpy="0.0 0.0 0.0" xyz="0.0 -0.1197 0.425"/>
    <axis xyz="0 1 0"/>
  </joint>
  <joint name="joint4" type="continuous">
    <parent link="link3"/>
    <child link="link4"/>
    <origin rpy="0.0 1.570796325 0.0" xyz="0.0 0.0 0.39225"/>
    <axis xyz="0 1 0"/>
  </joint>
  <joint name="joint5" type="continuous">
    <parent link="link4"/>
    <child link="link5"/>
    <origin rpy="0.0 0.0 0.0" xyz="0.0 0.093 0.0"/>
    <axis xyz="0 0 1"/>
  </joint>
  <joint name="joint6" type="continuous">
    <parent link="link5"/>
    <child link="link6"/>
    <origin rpy="0.0 0.0 0.0" xyz="0.0 0.0 0.09465"/>
    <axis xyz="0 1 0"/>
  </joint>
  <joint name="ee_joint" type="fixed">
    <origin rpy="-1.570796325 0 0" xyz="0 0.0823 0"/>
    <parent link="link6"/>
    <child link="ee_link"/>
  </joint>

<!-- ********** INERTIAL PROPERTIES (LINKS) ********** -->
  <link name="world"/>
  <link name="base_link">
    <inertial>
      <mass value="4.0"/>
      <origin rpy="0 0 0" xyz="0.0 0.0 0.0"/>
      <inertia ixx="0.00443333156" ixy="0.0" ixz="0.0"
               iyy="0.00443333156" iyz="0.0" izz="0.0072"/>
    </inertial>
  </link>
  <link name="link1">
    <inertial>
      <mass value="3.7"/>
      <origin rpy="0 0 0" xyz="0.0 0.0 0.0"/>
      <inertia ixx="0.010267495893" ixy="0.0" ixz="0.0"
               iyy="0.010267495893" iyz="0.0" izz="0.00666"/>
    </inertial>
  </link>
  <link name="link2">
    <inertial>
      <mass value="8.393"/>
      <origin rpy="0 0 0" xyz="0.0 0.0 0.28"/>
      <inertia ixx="0.22689067591" ixy="0.0" ixz="0.0"
               iyy="0.22689067591" iyz="0.0" izz="0.0151074"/>
    </inertial>
  </link>
  <link name="link3">
    <inertial>
      <mass value="2.275"/>
      <origin rpy="0 0 0" xyz="0.0 0.0 0.25"/>
      <inertia ixx="0.049443313556" ixy="0.0" ixz="0.0"
               iyy="0.049443313556" iyz="0.0" izz="0.004095"/>
    </inertial>
  </link>
  <link name="link4">
    <inertial>
      <mass value="1.219"/>
      <origin rpy="0 0 0" xyz="0.0 0.0 0.0"/>
      <inertia ixx="0.111172755531" ixy="0.0" ixz="0.0"
               iyy="0.111172755531" iyz="0.0" izz="0.21942"/>
    </inertial>
  </link>
  <link name="link5">
    <inertial>
      <mass value="1.219"/>
      <origin rpy="0 0 0" xyz="0.0 0.0 0.0"/>
      <inertia ixx="0.111172755531" ixy="0.0" ixz="0.0"
               iyy="0.111172755531" iyz="0.0" izz="0.21942"/>
    </inertial>
  </link>
  <link name="link6">
    <inertial>
      <mass value="0.1879"/>
      <origin rpy="0 0 0" xyz="0.0 0.0 0.0"/>
      <inertia ixx="0.0171364731454" ixy="0.0" ixz="0.0"
               iyy="0.0171364731454" iyz="0.0" izz="0.033822"/>
    </inertial>
  </link>
  <link name="ee_link"/>
</robot>
```



Denavit-Hartenberg (D-H) representation

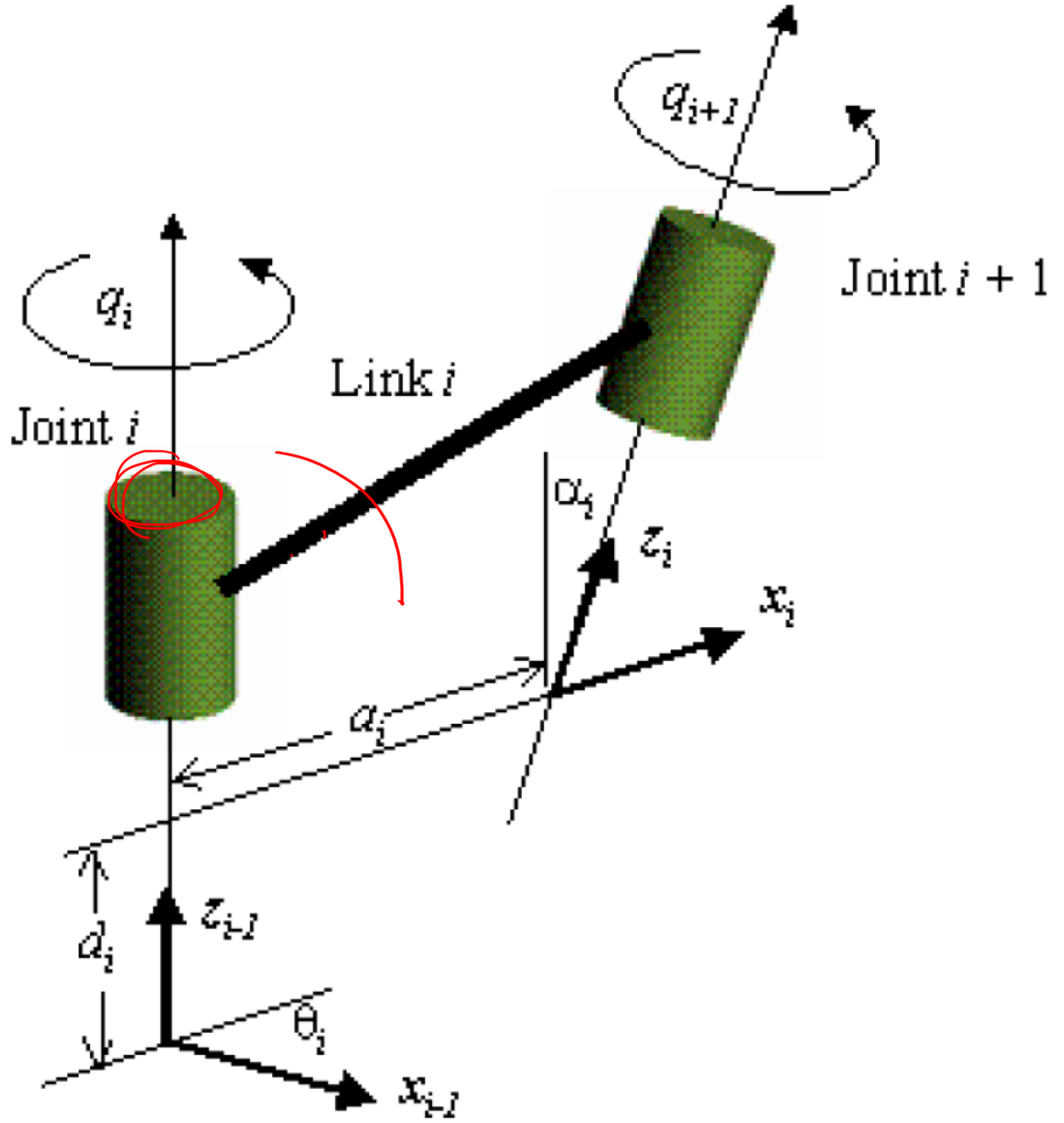
The basic idea underlying the Denavit–Hartenberg approach to forward kinematics is to attach reference frames to each link of the open chain and then to derive the forward kinematics from the knowledge of the relative displacements between adjacent link frames.

- 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



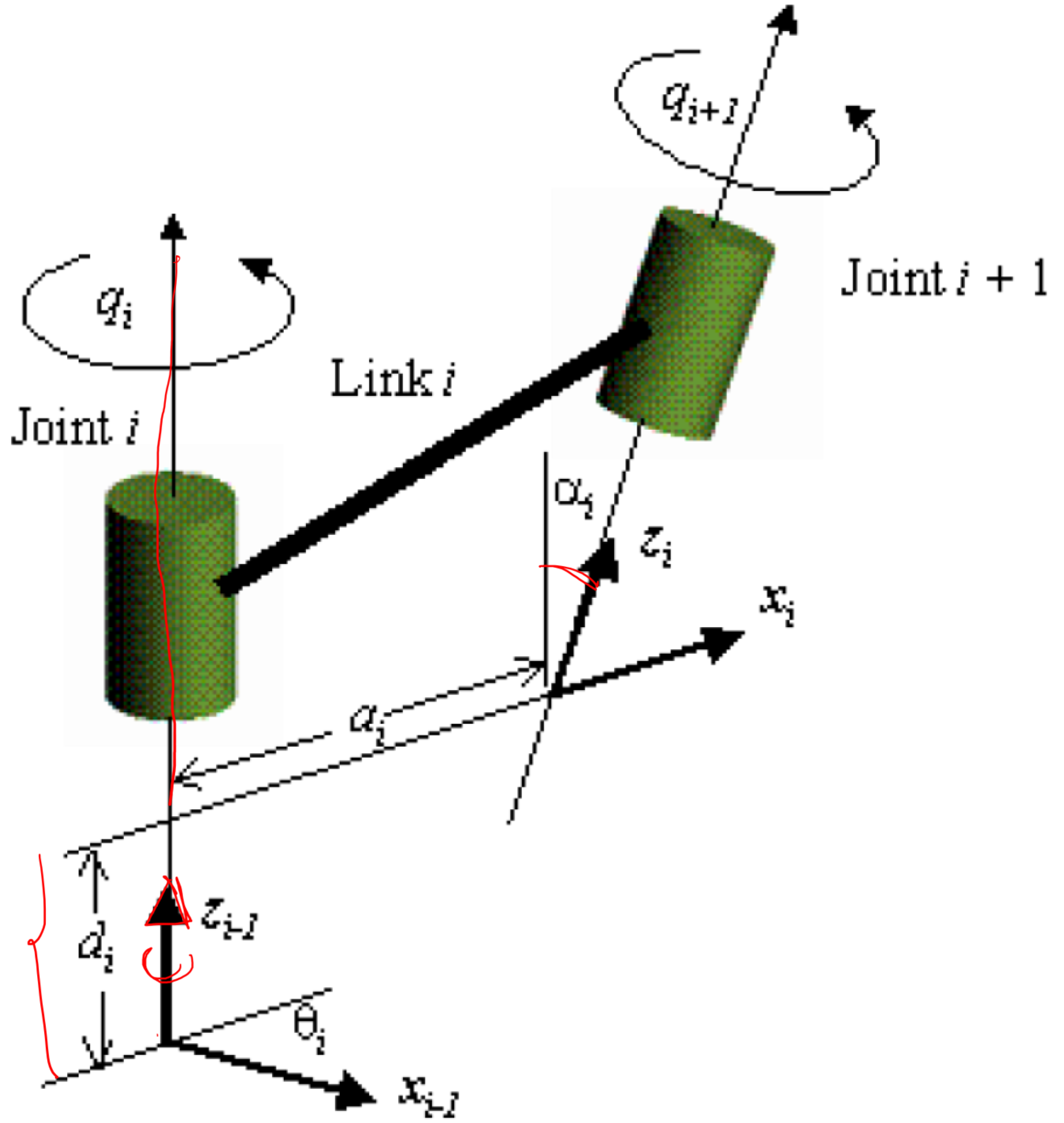
Link coordinate frames and their geometric parameters

- 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**
- 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.



Link coordinate frames and their geometric parameters

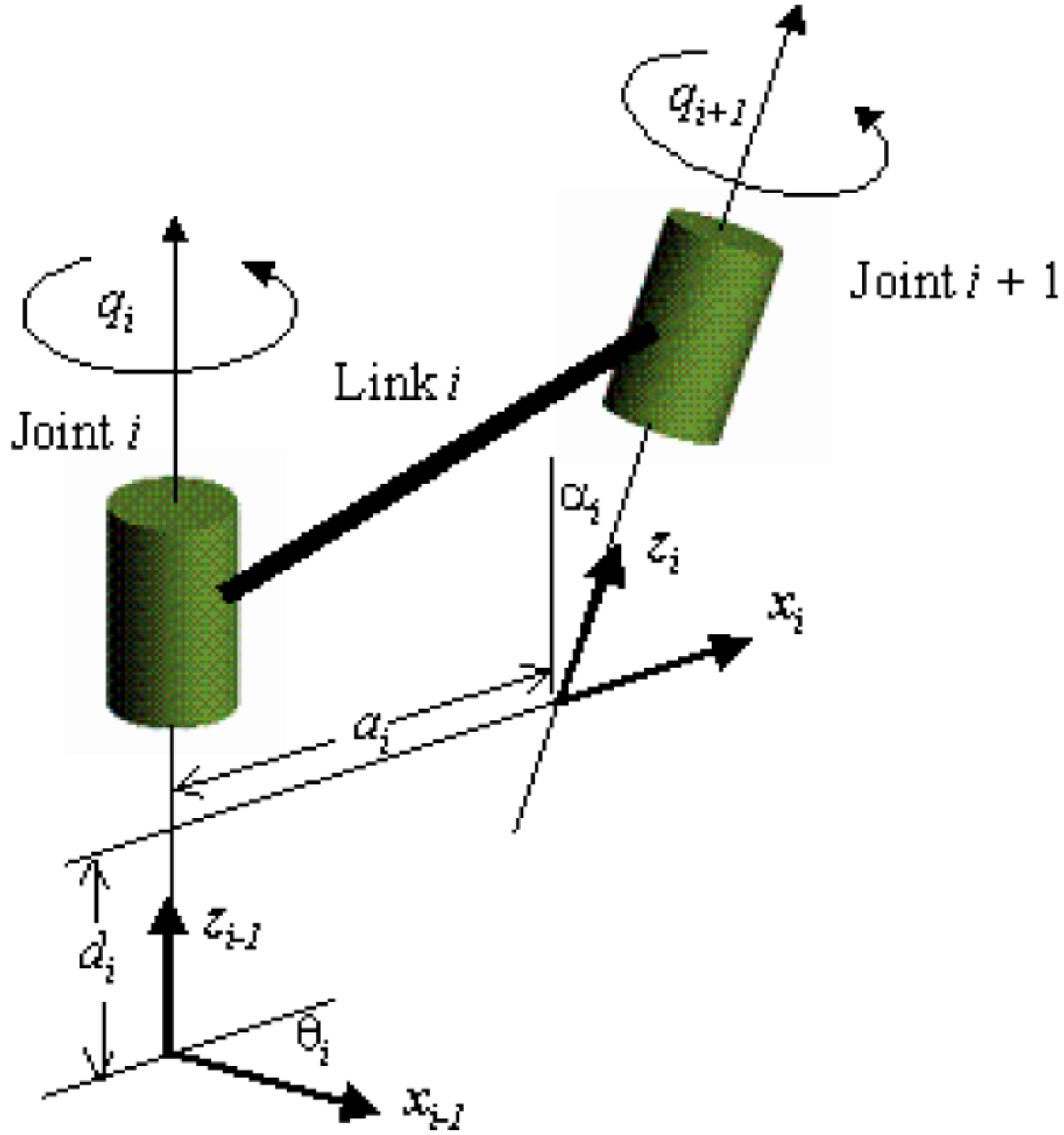
- From the kinematics viewpoint, a link keeps a fixed configuration between 2 joints (link structure).
- The structure of link *i* can be characterized through the length and the angle of the rotation axis of joint *i*.
- a_i = minimum distance along the common normal line between the two joint axes – **LINK LENGHT**
- α_i = angle between the two joint axes on a plane normal to a_i $-$ **TWIST**
- 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 adjacent links
- d_i = distance between normal lines, along the i-th joint axis – **LINK OFFSET**
- ϑ_i = angle between two normal lines, on a plane normal to the axis – **JOINT ANGLE**



Denavit-Hartenberg (D-H) representation

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

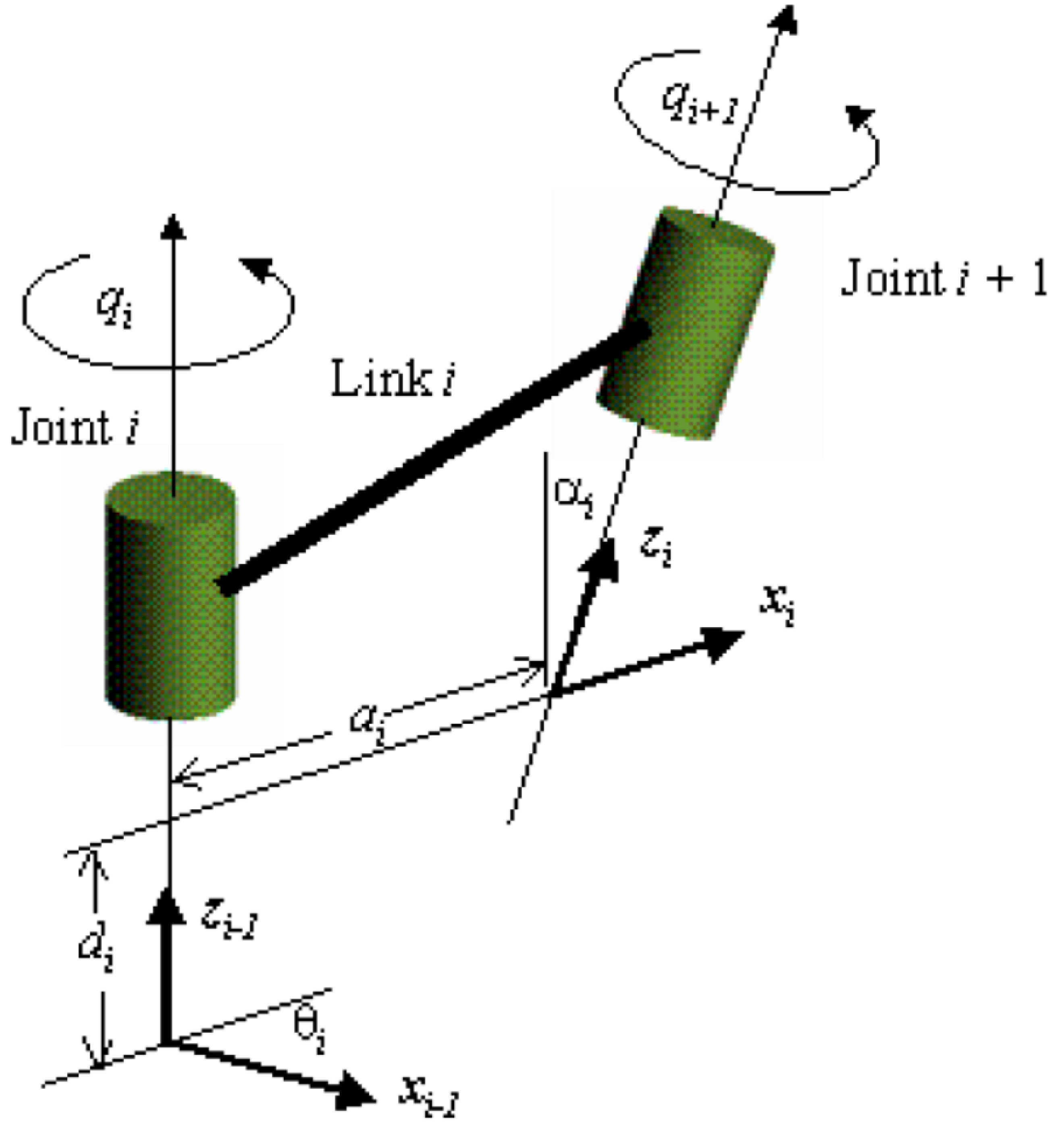
Algorithm:

I. Fix a base coordinate frame (0)

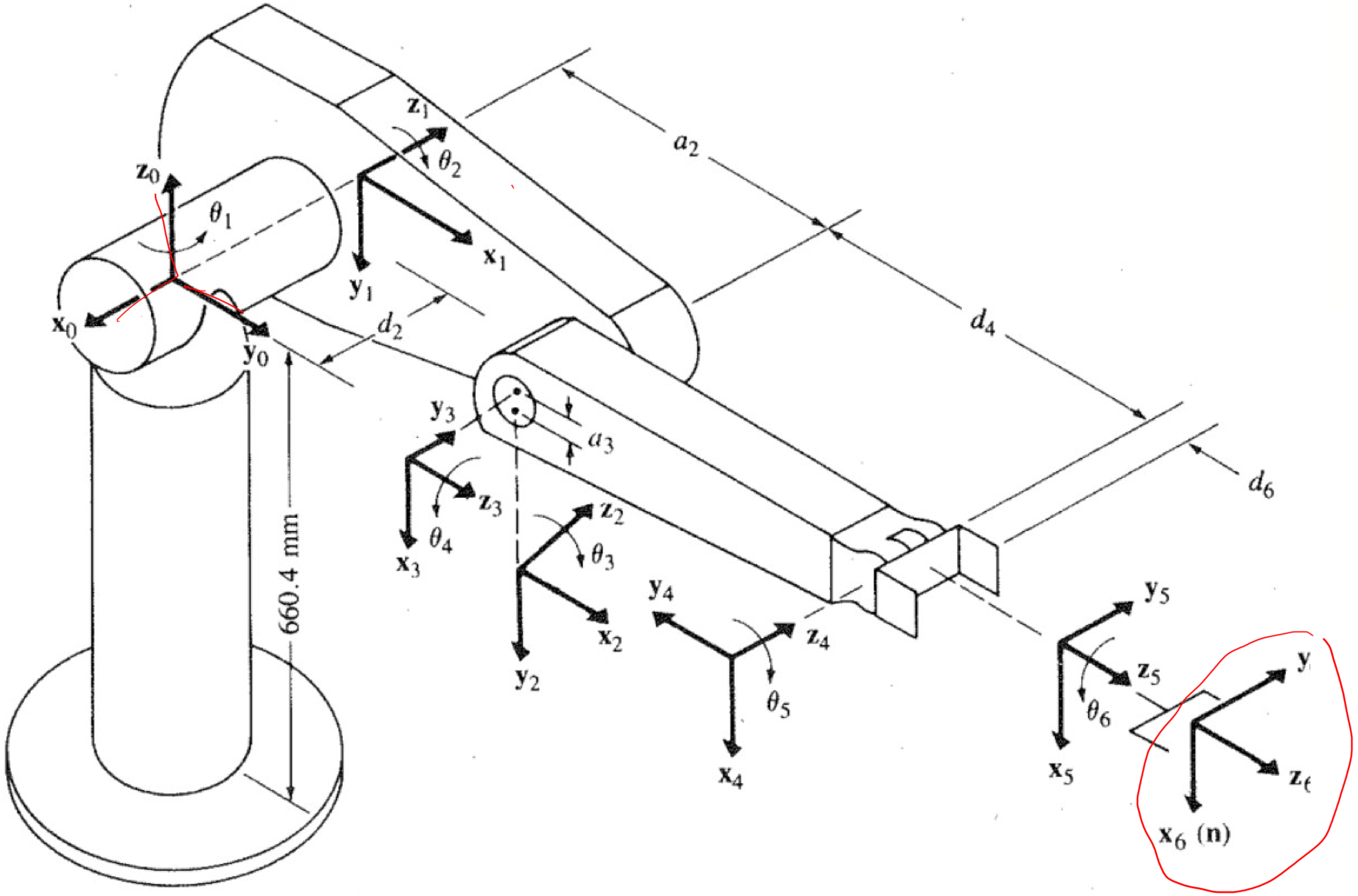
II. For each joint (1 a 5, for a 6-DOF robot), set:
- the joint axis,
- the origin of the coordinate frame,
- the x axis,
- the y axis.

III. Fix the end-effector coordinate frame.

IV. For each joint and for each link, set:
- the joint parameters
- the link parameters.



D-H for PUMA 560



Parametri delle coordinate dei link per il braccio PUMA

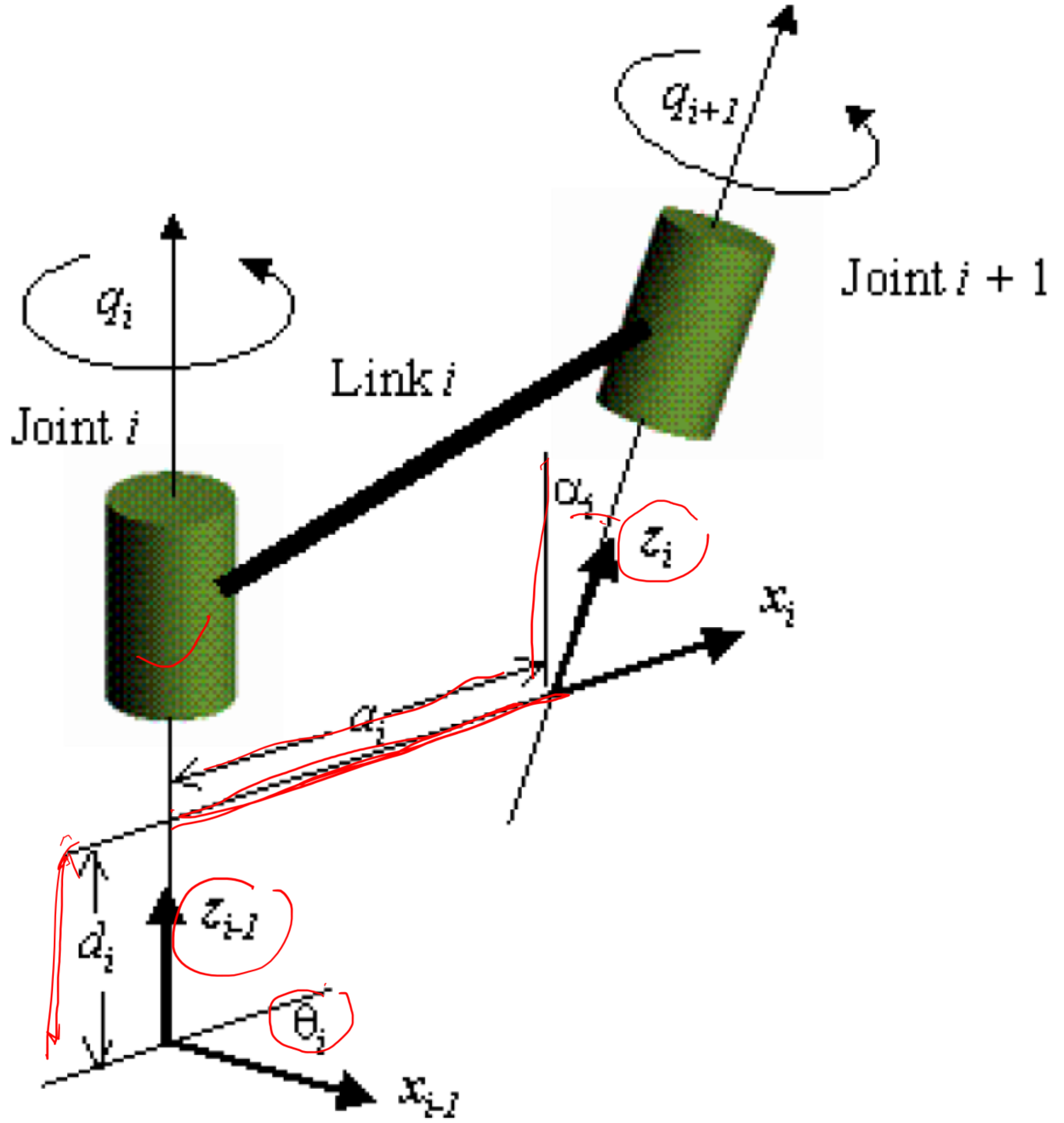
JOINT ~~Giunto~~ i	θ_i	α_i	a_i	d_i	Escursione del giunto
1	90	−90	0	0	−160 to +160
2	0	0	431.8 mm	149.09 mm	−225 to 45
3	90	90	−20.32 mm	0	−45 to 225
4	0	−90	0	433.07 mm	−110 to 170
5	0	90	0	0	−100 to 100
6	0	0	0	56.25 mm	−266 to 266

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 align 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 align the x axes



Denavit-Hartenberg (D-H) representation

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

$$^{i-1}\boldsymbol{A}_i = \boxed{\mathrm{R}_{z,\theta}\mathrm{T}_{z,d}\mathrm{T}_{x,a}\mathrm{R}_{x,\alpha}}$$

$$r_{i-1} = {}^{i-1}\boldsymbol{A}_i p_1 = \begin{bmatrix} \cos(\vartheta_i) & -\cos(\alpha_i)\sin(\vartheta_i) & \sin(\alpha_i)\sin(\vartheta_i) & a_i\cos(\vartheta_i) \\ \sin(\vartheta_i) & \cos(\alpha_i)\cos(\vartheta_i) & -\sin(\alpha_i)\cos(\vartheta_i) & a_i\sin(\alpha_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

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 $\boldsymbol{\vartheta_i}$ varies.**
- For a **prismatic joint:** ϑ_i, a_i , α_i are the joint parameters, constant for a given robot. **Only $\boldsymbol{d_i}$ 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 ${}^{i-1}\boldsymbol{A}_i$, expressed as:

$$ {}^{0}\boldsymbol{T}_n = {}^{0}\boldsymbol{A}_1\,{}^{1}\boldsymbol{A}_2 \ldots {}^{n-1}\boldsymbol{A}_n $$

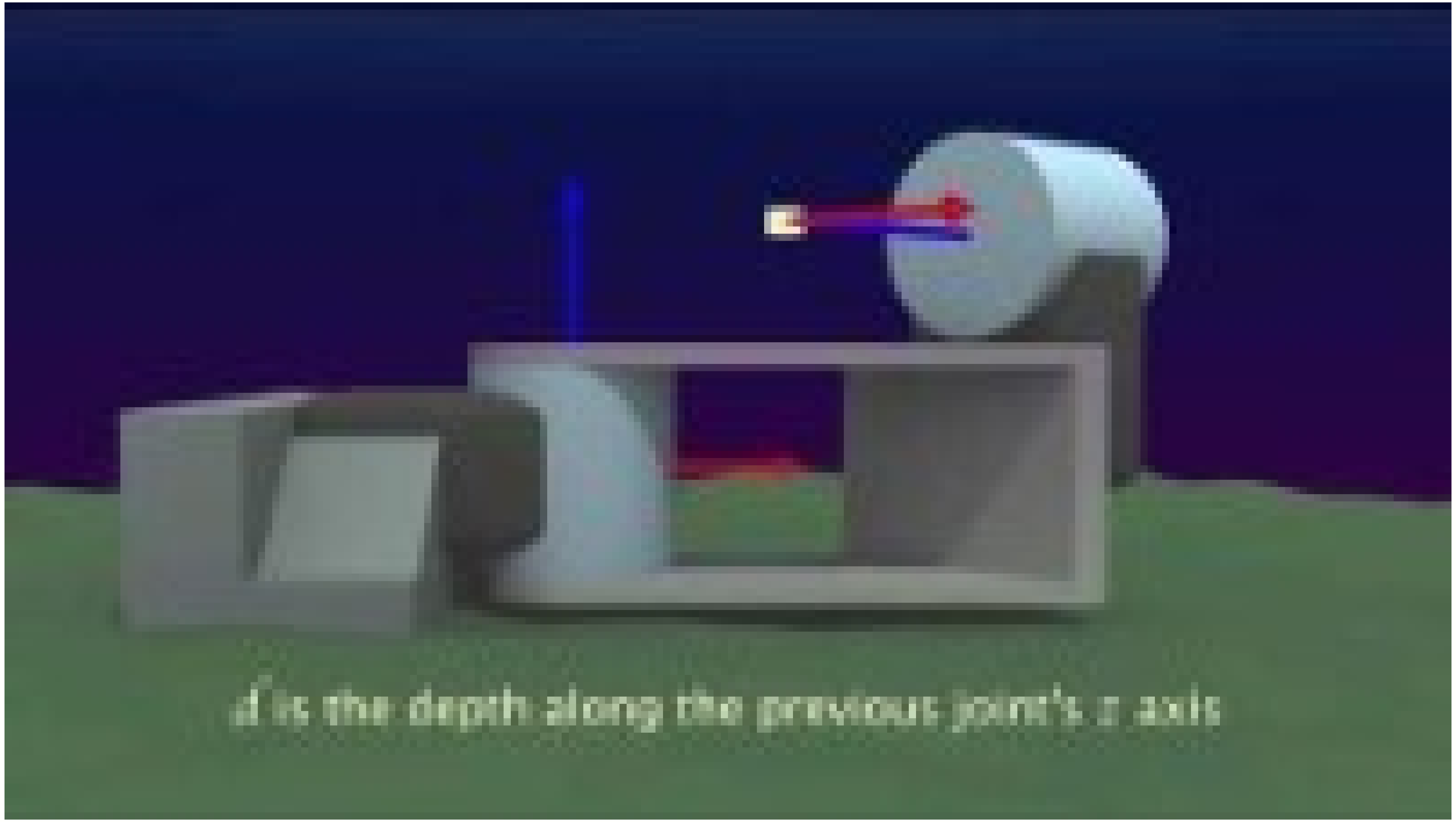
$$ {}^{0}\boldsymbol{T}_n = \begin{bmatrix} \boldsymbol{X}_i & \boldsymbol{Y}_i & \boldsymbol{Z}_i & \boldsymbol{p}_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad {}^{0}\boldsymbol{T}_n = \begin{bmatrix} \overset{3\times3}{\boxed{{}^{0}\boldsymbol{R}_n}} & \overset{3\times1}{\boxed{{}^{0}\boldsymbol{p}_n}} \\ 0 & 1 \end{bmatrix} $$

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

$\boldsymbol{P_i}$ is the position vector pointing from the origin of the base frame to the origin of the $n-th$ frame

$\boldsymbol{R}$ is the matrix describing the roll, pitch and yaw angles

Denavit-Hartenberg (D-H) representation



https://youtu.be/rA9tm0gTln8

SUMMARY

We may summarize the above procedure based on the D-H convention in the following algorithm for deriving the forward kinematics for any manipulator.

Step 1: Locate and label the joint axes $z_0, \ldots, z_{n-1}$.

Step 2: Establish the base frame. Set the origin anywhere on the z_0-axis. The x_0 and y_0 axes are chosen conveniently to form a right-hand frame.

For $i = 1, \ldots, n-1$, perform Steps 3 to 5.

Step 3: Locate the origin o_i where the common normal to z_i and z_{i-1} intersects z_i. If z_i intersects z_{i-1} locate o_i at this intersection. If z_i and z_{i-1} are parallel, locate o_i in any convenient position along z_i.

Step 4: Establish x_i along the common normal between z_{i-1} and z_i through o_i, or in the direction normal to the $z_{i-1} - z_i$ plane if z_{i-1} and z_i intersect.

Step 5: Establish y_i to complete a right-hand frame.

Step 6: Establish the end-effector frame $o_n x_n y_n z_n$. Assuming the n-th joint is revolute, set $z_n = \boldsymbol{a}$ along the direction z_{n-1}. Establish the origin o_n conveniently along z_n, preferably at the center of the gripper or at the tip of any tool that the manipulator may be carrying. Set $y_n = \boldsymbol{s}$ in the direction of the gripper closure and set $x_n = \boldsymbol{n}$ as $\boldsymbol{s} \times \boldsymbol{a}$. If the tool is not a simple gripper set x_n and y_n conveniently to form a right-hand frame.

Step 7: Create a table of link parameters a_i, d_i, α_i, θ_i.

$a_i =$ distance along x_i from o_i to the intersection of the x_i and z_{i-1} axes.

$d_i =$ distance along z_{i-1} from o_{i-1} to the intersection of the x_i and z_{i-1} axes. d_i is variable if joint i is prismatic.

$\alpha_i =$ the angle between z_{i-1} and z_i measured about x_i (see Figure 3.3).

$\theta_i =$ the angle between x_{i-1} and x_i measured about z_{i-1} (see Figure 3.3). θ_i is variable if joint i is revolute.

Step 8: Form the homogeneous transformation matrices A_i by substituting the above parameters into (3.10).

Step 9: Form $T_n^0 = A_1 \cdots A_n$. This then gives the position and orientation of the tool frame expressed in base coordinates.

$$
\begin{aligned}
A_i &= Rot_{z,\theta_i} \mathrm{Trans}_{z,d_i} \mathrm{Trans}_{x,a_i} Rot_{x,\alpha_i} \qquad (3.10)\\
&= \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} & 0 & 0 \\ s_{\theta_i} & c_{\theta_i} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_{\alpha_i} & -s_{\alpha_i} & 0 \\ 0 & s_{\alpha_i} & c_{\alpha_i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\\
&= \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} c_{\alpha_i} & s_{\theta_i} s_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i} c_{\alpha_i} & -c_{\theta_i} s_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}
$$

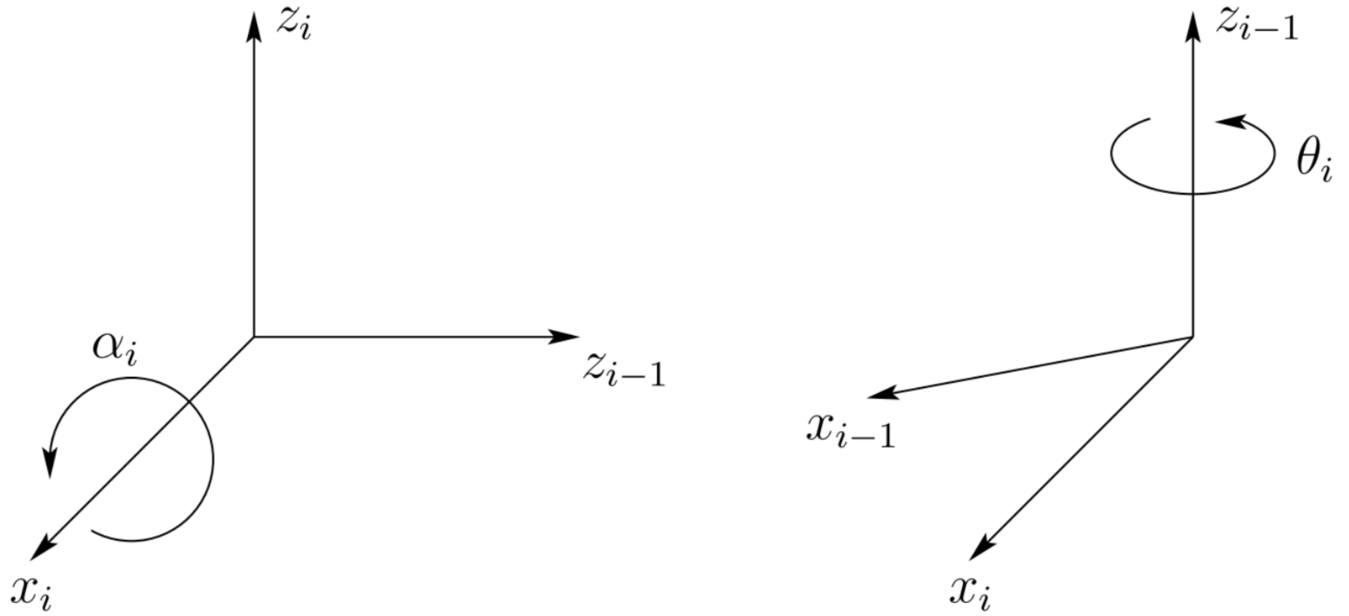


Figure 3.3: Positive sense for α_i and θ_i.

Denavit-Hartenberg (D-H) representation

The direct kinematics of a 6-link manipulator can be solved by calculating $T = {}^{0}\boldsymbol{A}_6$ 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 $\vartheta_i = q_i$

For a given $q = (q_0, q_1, q_2, q_3, q_4, q_5)$ it is possible to find $(x, y, z, roll, pitch, yaw)$

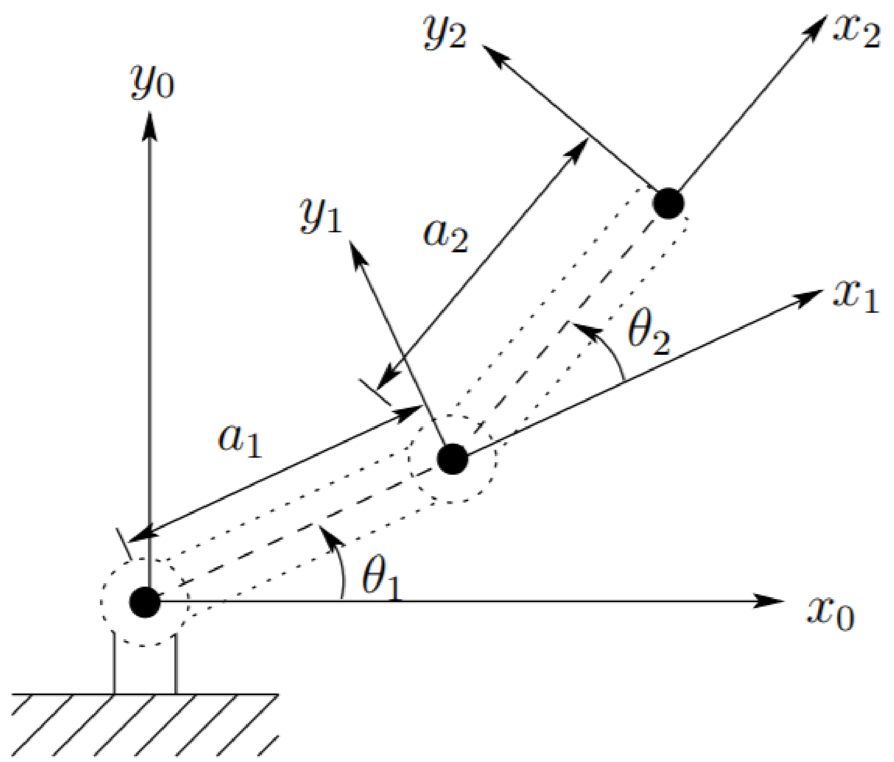
$$x = K(q) = T(q)$$

Denavit-Hartenberg

- In the D-H convention the only variable angle is ϑ, so we simplify notation by writing c_i for cos θ_i, etc.
- We also denote $\theta_1 + \theta_2$by θ_{12}, and $\cos(\theta_1 + \theta_2)$ by c_{12}, and so on.
- In the following examples it is important to remember that the D-H convention, while systematic, still allows considerable freedom in the choice of some of the manipulator parameters.
- This is particularly true in the case of parallel joint axes or when prismatic joints are involved.

Planar Elbow Manipulator

- Consider the two-link planar arm of Figure 3.6. The joint axes z_0 and z_1 are normal to the page.
- We establish the base frame $o_0\ x_0\ y_0\ z_0$ as shown.
- The origin is chosen at the point of intersection of the z_0 axis with the page and the direction of the x_0 axis is completely arbitrary.
- Once the base frame is established, the $o_1\ x_1\ y_1\ z_1$ frame is fixed as shown by the *D-H* convention, where the origin O_1 has been located at the intersection of z_1 and the page.
- The final frame $o_2\ x_2\ y_2\ z_2$ is fixed by choosing the origin O_2 at the end of link 2 as shown.



Two-link planar manipulator. The z-axes all point out of the page, and are not shown in the figure.

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

Planar Elbow Manipulator

$$^{i-1}\boldsymbol{A}_i = \mathrm{R}_{z,\theta}\mathrm{T}_{z,d}\mathrm{T}_{x,a}\mathrm{R}_{x,\alpha}$$

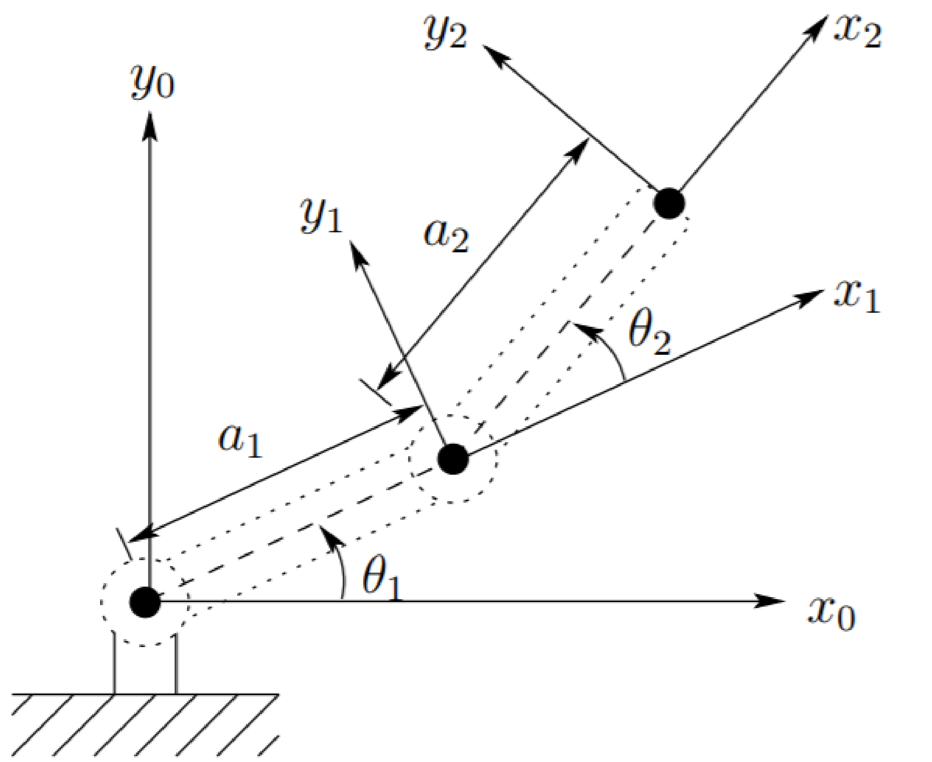
$$r_{i-1} = {}^{i-1}\boldsymbol{A}_i p_1 = \begin{bmatrix} \cos(\vartheta_i) & -\cos(\alpha_i)\sin(\vartheta_i) & \sin(\alpha_i)\sin(\vartheta_i) & a_i\cos(\vartheta_i) \\ \sin(\vartheta_i) & \cos(\alpha_i)\cos(\vartheta_i) & -\sin(\alpha_i)\cos(\vartheta_i) & a_i\sin(\alpha_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- The link parameters are shown in Table. The A-matrices are

Link parameters for 2-link planar manipulator.

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

* variable



Two-link planar manipulator. The z-axes all point out of the page, and are not shown in the figure.

Planar Elbow Manipulator

- The link parameters are shown in Table. The A-matrices are

Link parameters for 2-link planar manipulator.

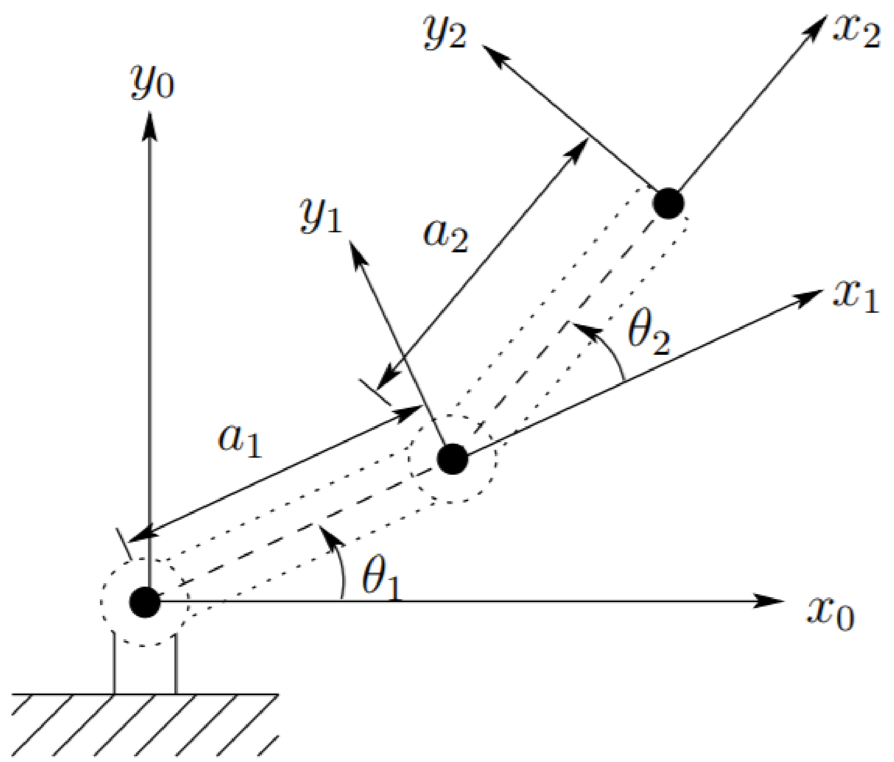
Link	a_i	α_i	d_i	θ_i
1	a_1	0	0	θ_1^*
2	a_2	0	0	θ_2^*

* variable

- The A-matrices are determined as :

$$A_1 = \begin{bmatrix} c_1 & -s_1 & 0 & a_1c_1 \\ s_1 & c_1 & 0 & a_1s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

$$A_2 = \begin{bmatrix} c_2 & -s_2 & 0 & a_2c_2 \\ s_2 & c_2 & 0 & a_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Two-link planar manipulator. The z-axes all point out of the page, and are not shown in the figure.

Planar Elbow Manipulator

- The T-matrices are thus given by

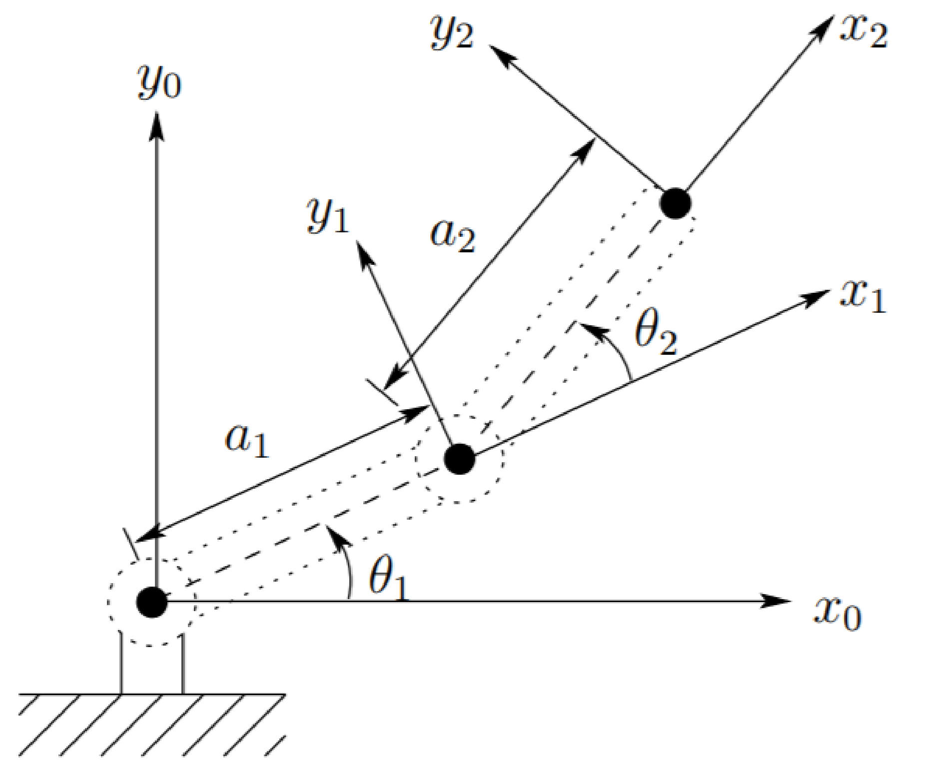
$$T_1^0 = A_1.$$

$$T_2^0 = A_1 A_2 = \begin{bmatrix} c_{12} & -s_{12} & 0 & a_1c_1 + a_2c_{12} \\ s_{12} & c_{12} & 0 & a_1s_1 + a_2s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

- Notice that the first two entries of the last column of T_2^0 are the x and y components of the origin O_2 in the base frame; that is,

$$x = a_1c_1 + a_2c_{12}$$
$$y = a_1s_1 + a_2s_{12}$$

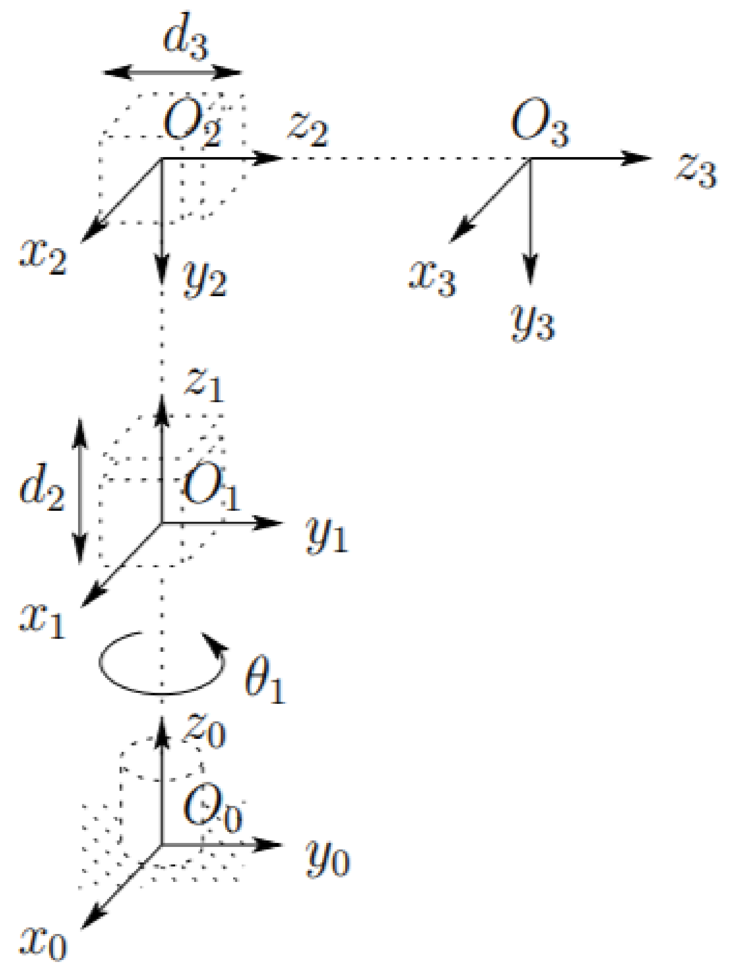
are the coordinates of the end-effector in the base frame. The rotational part of T_2^0 gives the orientation of the frame $o_2\ x_2\ y_2\ z_2$ relative to the base frame.



Two-link planar manipulator. The z-axes all point out of the page, and are not shown in the figure.

Three-Link Cylindrical Robot

- Consider now the three-link cylindrical robot represented symbolically by Figure .
- We establish O_0 as shown at joint 1. Note that the placemen of the origin O_0 along z_0 as well as the direction of the x_0 axis are arbitrary.
- Our choice of O_0 is the most natural, but O_0 could just as well be placed at joint 2.
- The axis x_0 is chosen normal to the page. Next, since z_0 and
- z_1 coincide, the origin O_1 is chosen at joint 1 as shown.
- The x_1 axis is normal to the page when $\vartheta_1 = 0$ but, o course its direction will change since ϑ_1 is variable.
- Since z_2 and z_1 intersect, the origin O_2 is placed at this
- intersection. The direction of z_2 is chosen parallel to x_1 so that ϑ_2 is zero.
- Finally, the third frame is chosen at the end of link 3 as shown.



Three-link cylindrical manipulator.

Three-Link Cylindrical Robot

- The link parameters are now shown in Table

Link parameters for 3-link cylindrical manipulator.

Link	a_i	α_i	d_i	θ_i
1	0	0	d_1	θ_1^*
2	0	-90	d_2^*	0
3	0	0	d_3^*	0

* variable

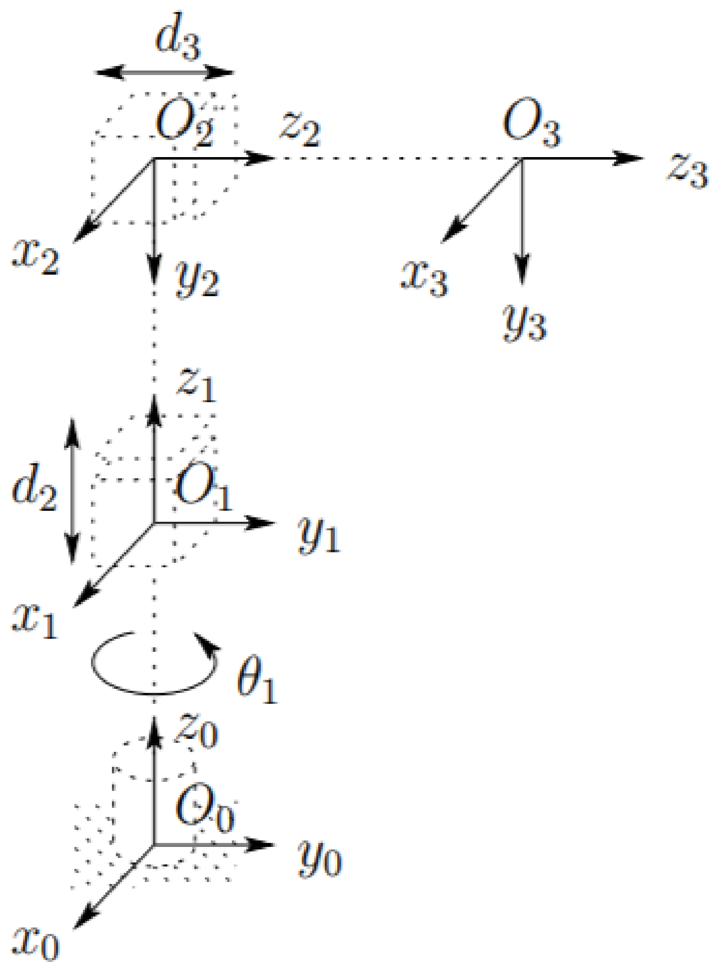
- The corresponding A and T matrices are

$$A_1 = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

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



Three-link cylindrical manipulator.

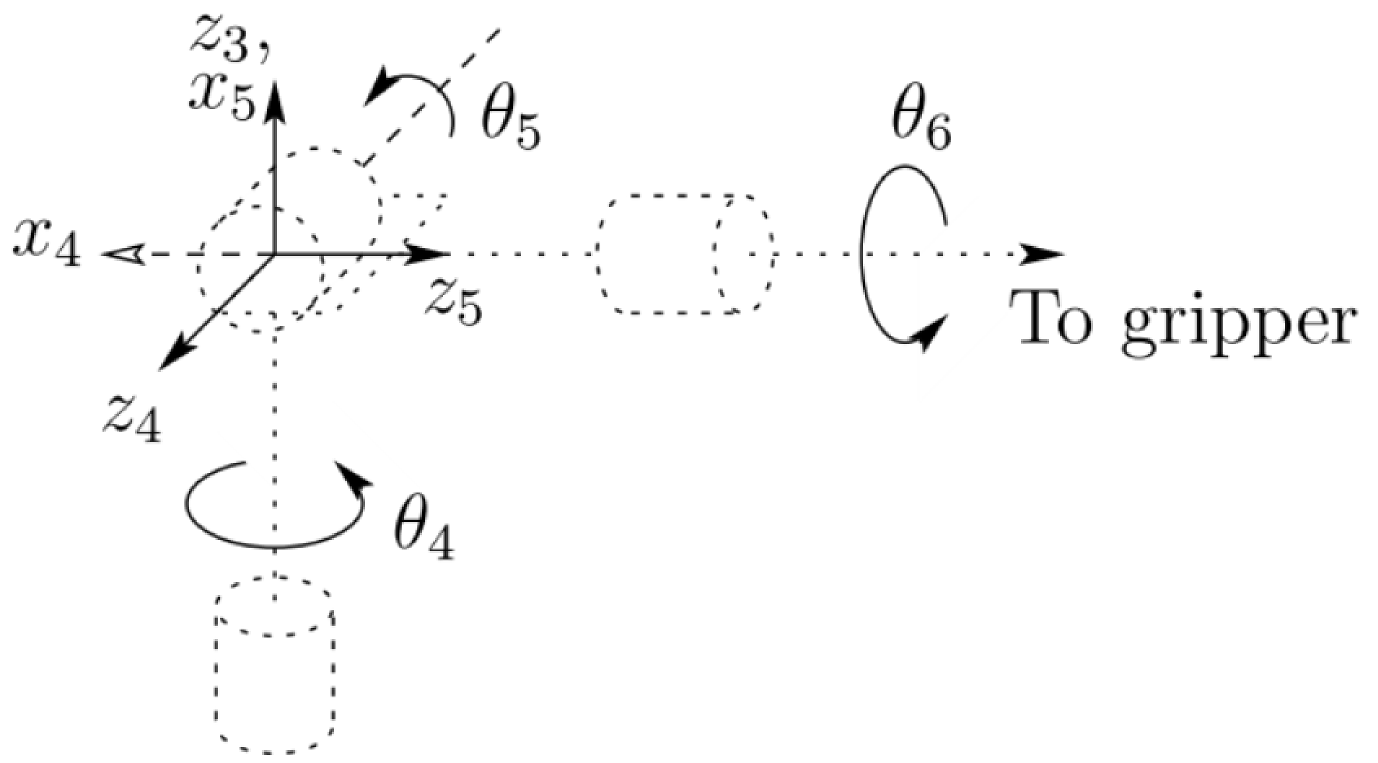
Spherical Wrist

- The spherical wrist configuration is shown in Figure, in which the joint axes z_3, z_4, z_5 intersect at O.
- The Denavit-Hartenberg parameters are shown in Table.
- The Stanford manipulator is an example of a manipulator that possesses a wrist of this type. In fact, the following analysis applies to virtually all spherical wrists.
- We show now that the final three joint variables $\theta_4, \theta_5, \theta_6$ are the Euler angles φ, θ, ψ, respectively, with respect to the coordinate frame o_3, x_3, y_3, z_3.

DH parameters for spherical wrist

Link	a_i	α_i	d_i	θ_i
4	0	−90	0	θ_4^*
5	0	90	0	θ_5^*
6	0	0	d_6	θ_6^*

* variable



The spherical wrist frame assignment.

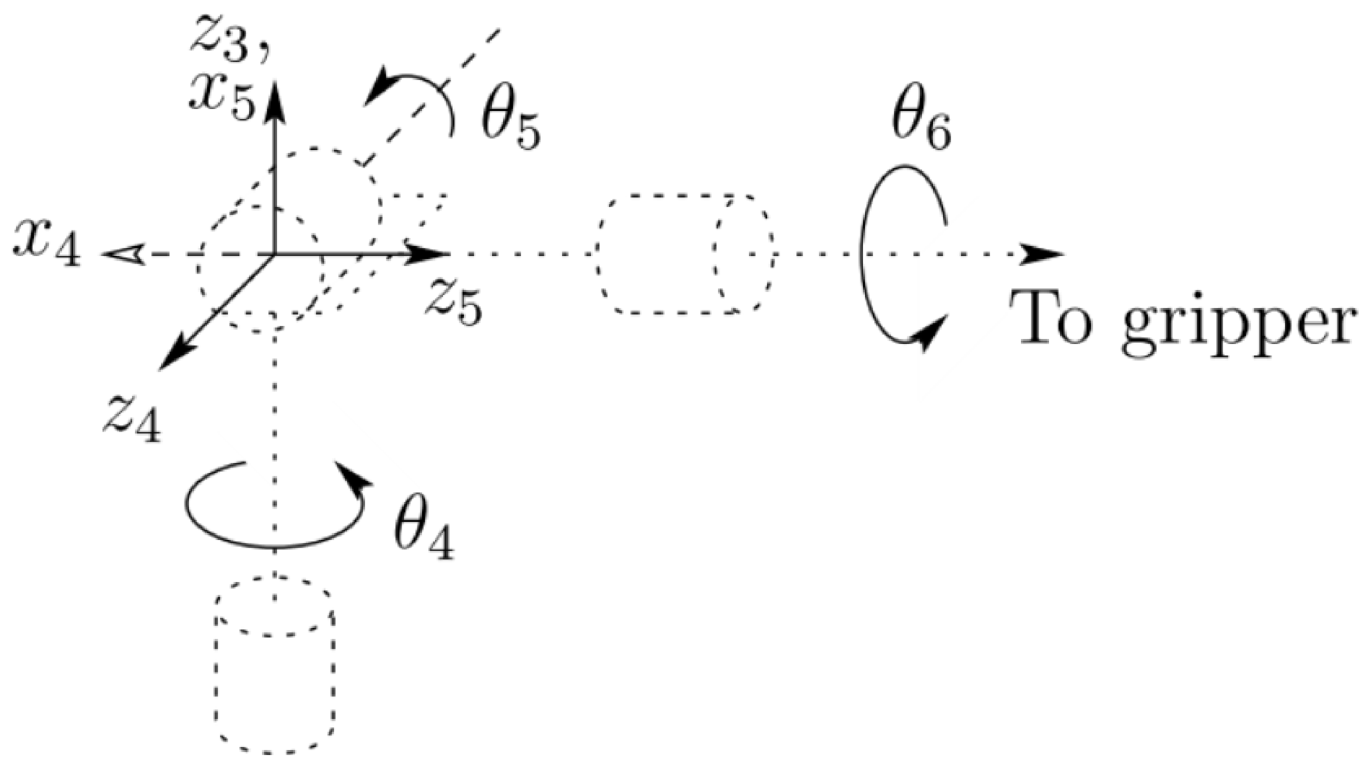
Spherical Wrist

- To see this we need only compute the matrices A_4, A_5 and A_6 using Table.

$$A_4 = \begin{bmatrix} c_4 & 0 & -s_4 & 0 \\ s_4 & 0 & c_4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_5 = \begin{bmatrix} c_5 & 0 & s_5 & 0 \\ s_5 & 0 & -c_5 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_6 = \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



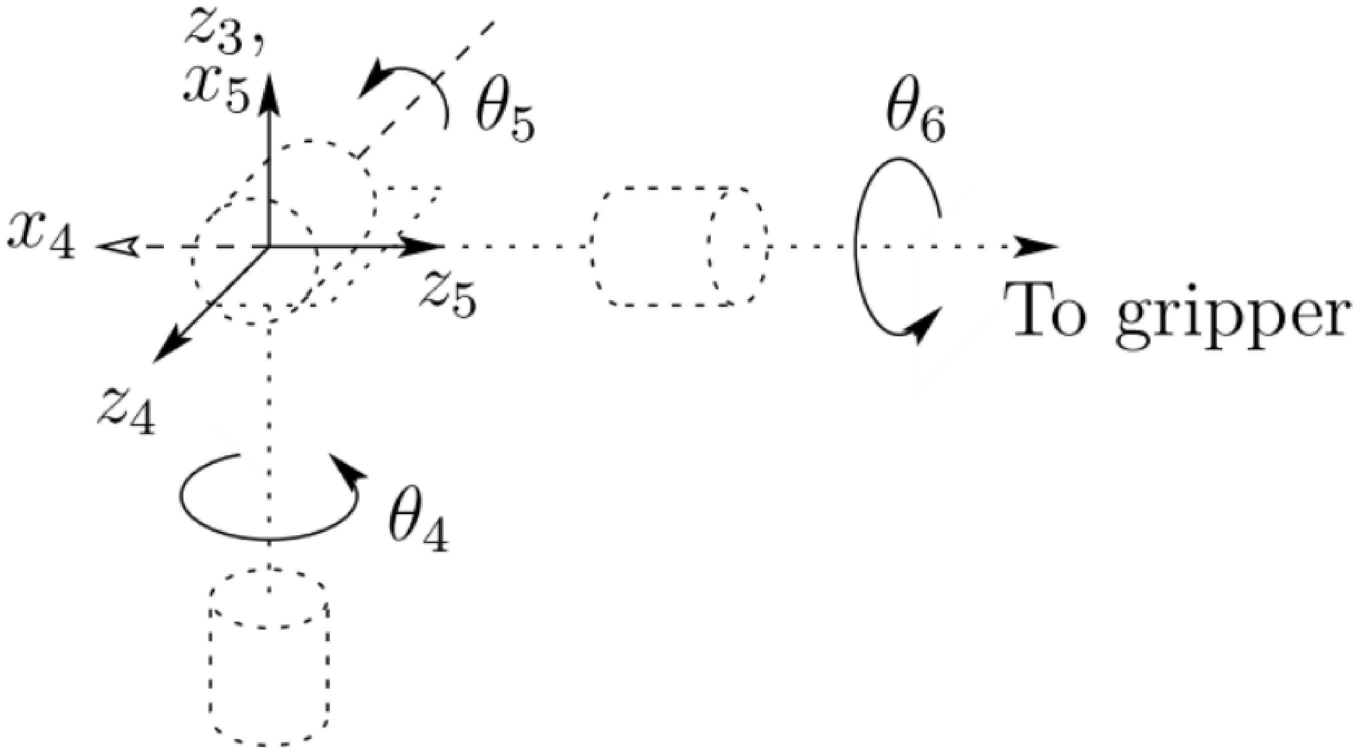
The spherical wrist frame assignment.

Spherical Wrist

- Multiplying these together yields

$$T_6^3 = A_4 A_5 A_6 = \begin{bmatrix} R_6^3 & O_6^3 \\ 0 & 1 \end{bmatrix}$$

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

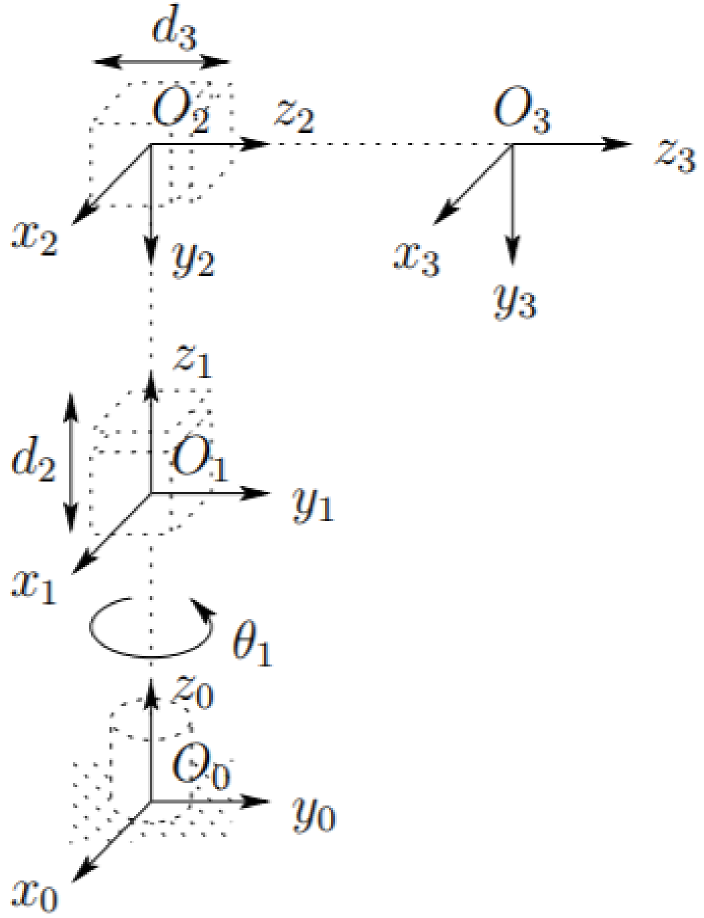


The spherical wrist frame assignment.

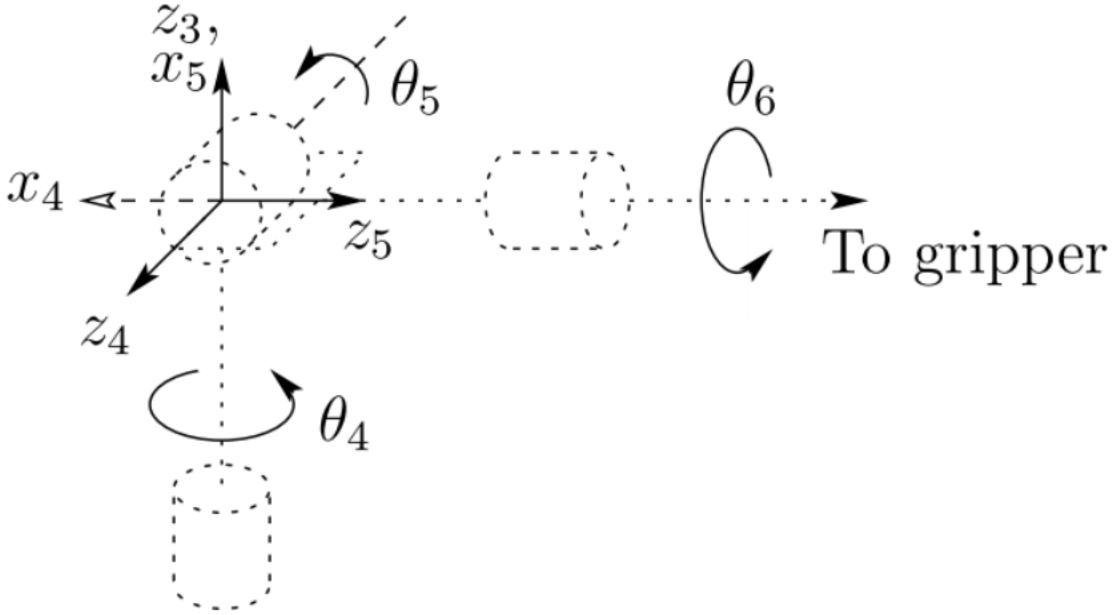
- Comparing the rotational part $R_6{}^3$ of $T_6{}^3$ with the Euler angle transformation shows that $\theta_4, \theta_5, \theta_6$ can indeed be identified as the Euler angles φ, θ and ψ with respect to the coordinate frame o_3, x_3, y_3, z_3.

Cylindrical Manipulator with Spherical Wrist

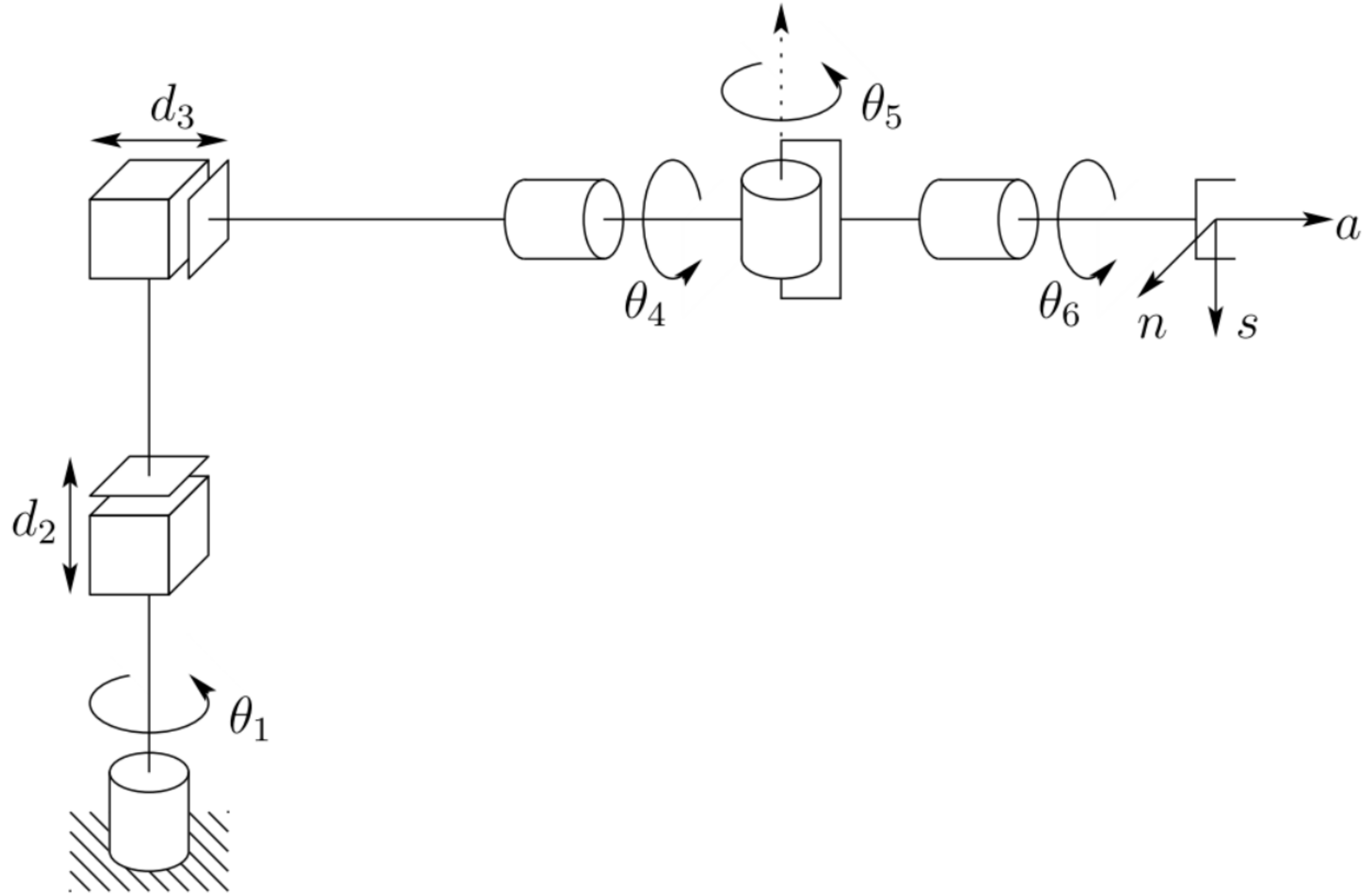
- Suppose that we now attach a spherical wrist to the cylindrical manipulator of Example 2 as shown in Figure.
- Note that the axis of rotation of joint 4 is parallel to z_2 and thus coincides with the axis z_3 of Example 2.



Three-link cylindrical manipulator.



The spherical wrist frame assignment.



Cylindrical robot with spherical wrist.

Cylindrical Manipulator with Spherical Wrist

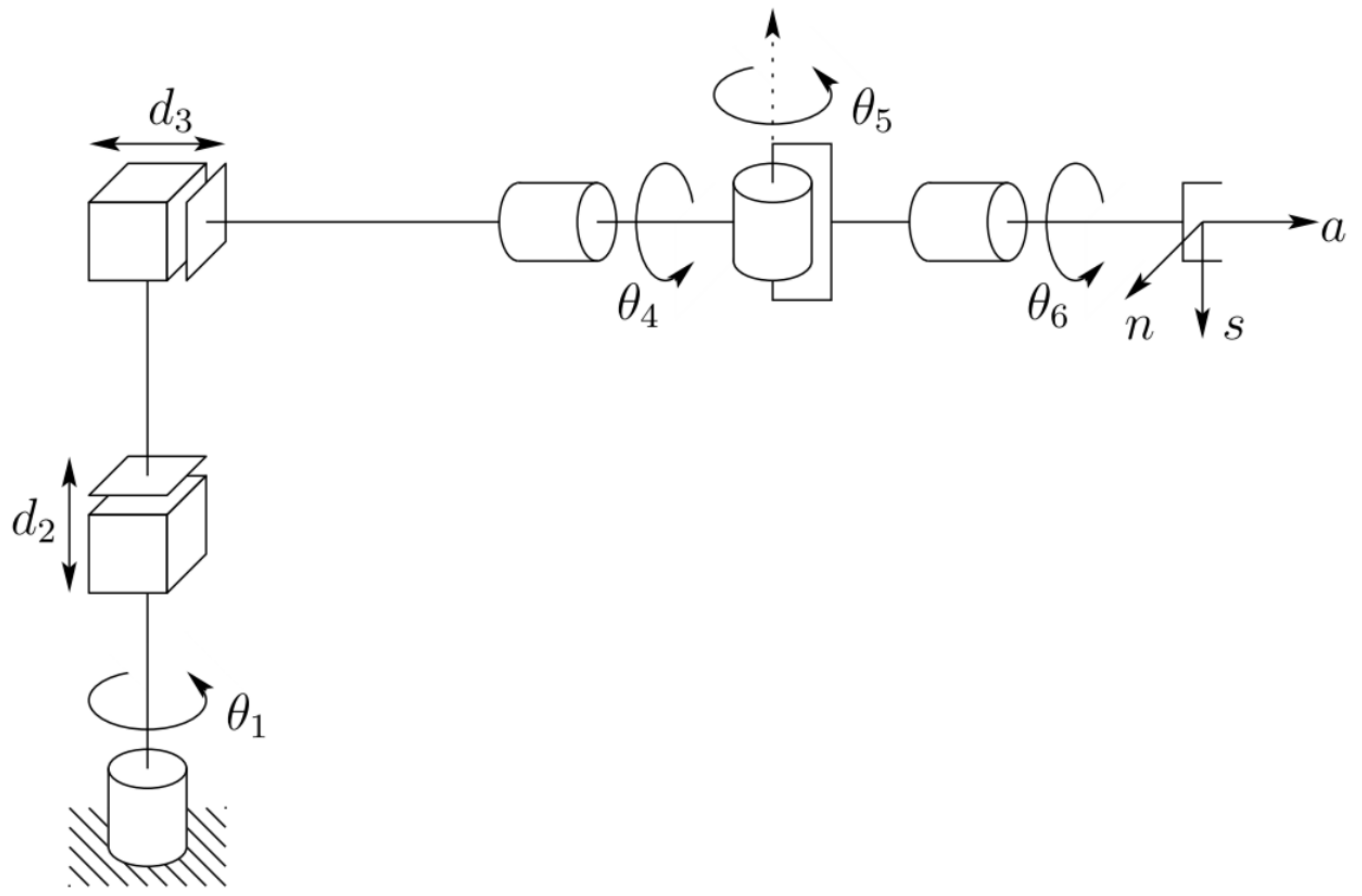
- The implication of this is that we can immediately combine the two previous expressions to derive the forward kinematics as

$$T_6^0 \quad = \quad T_3^0 T_6^3$$

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

$$T_6^3 = A_4 A_5 A_6 \quad = \begin{bmatrix} R_6^3 & O_6^3 \\ 0 & 1 \end{bmatrix}$$

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



Cylindrical robot with spherical wrist.

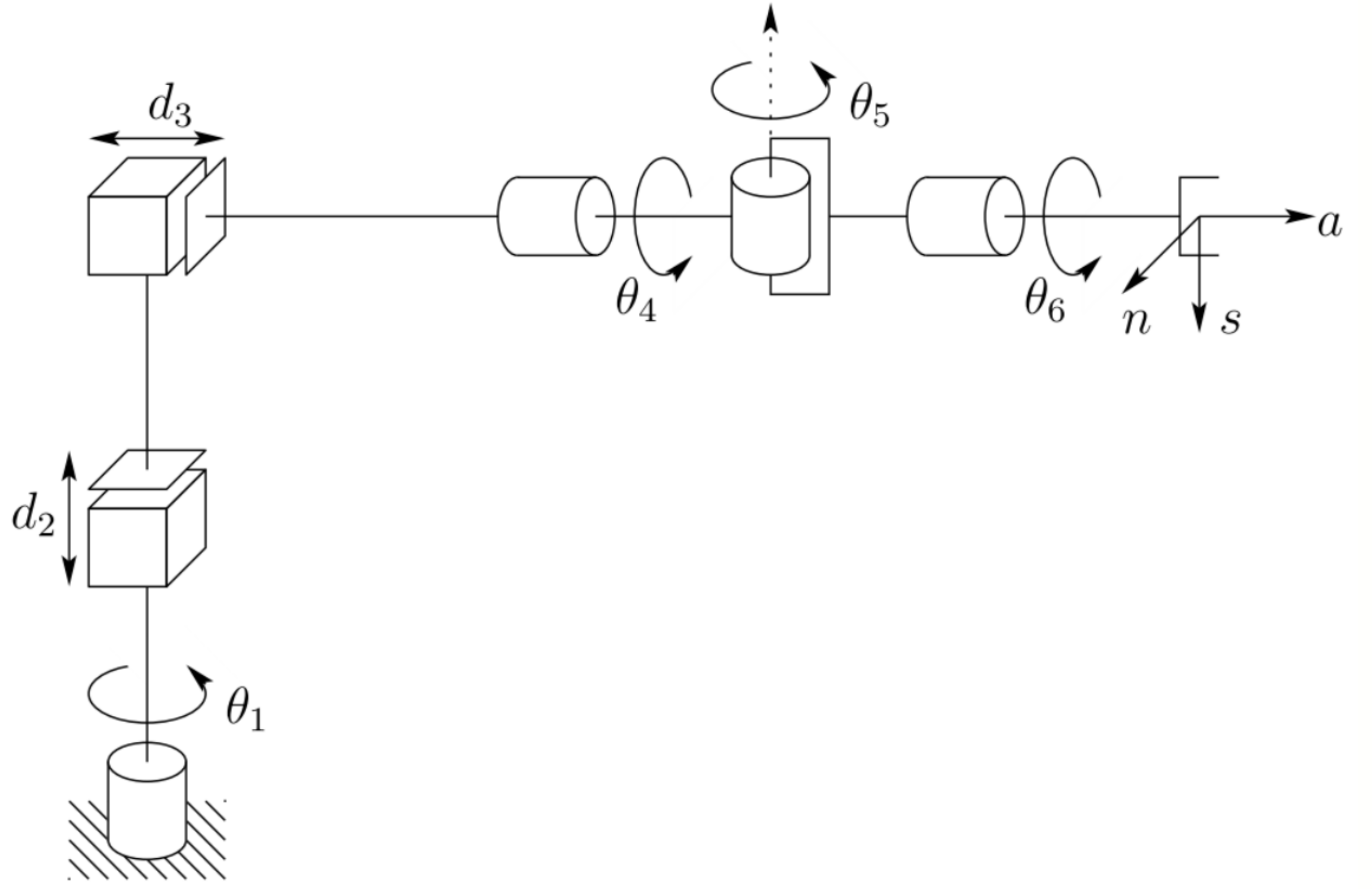
Cylindrical Manipulator with Spherical Wrist

- The implication of this is that we can immediately combine the two previous expressions to derive the forward kinematics as

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

- Therefore the forward kinematics of this manipulator is described by

$$T_6^0 = \begin{bmatrix} c_1 & 0 & -s_1 & -s_1 d_1 \\ s_1 & 0 & c_1 & c_1 d_3 \\ 0 & -1 & 0 & d_1 + d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_4 c_5 c_6 - s_4 s_6 & -c_4 c_5 s_6 - s_4 c_6 & c_4 s_5 & c_4 s_5 d_6 \\ s_4 c_5 c_6 + c_4 s_6 & -s_4 c_5 s_6 + c_4 c_6 & s_4 s_5 & s_4 s_5 d_6 \\ -s_5 c_6 & s_5 s_6 & c_5 & c_5 d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
$$= \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Cylindrical robot with spherical wrist.

Cylindrical Manipulator with Spherical Wrist

$$T_6^0 = \begin{bmatrix} c_1 & 0 & -s_1 & -s_1 d_1 \\ s_1 & 0 & c_1 & c_1 d_3 \\ 0 & -1 & 0 & d_1 + d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_4c_5c_6 - s_4s_6 & -c_4c_5s_6 - s_4c_6 & c_4s_5 & c_4s_5d_6 \\ s_4c_5c_6 + c_4s_6 & -s_4c_5s_6 + c_4c_6 & s_4s_5 & s_4s_5d_6 \\ -s_5c_6 & s_5s_6 & c_5 & c_5d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

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

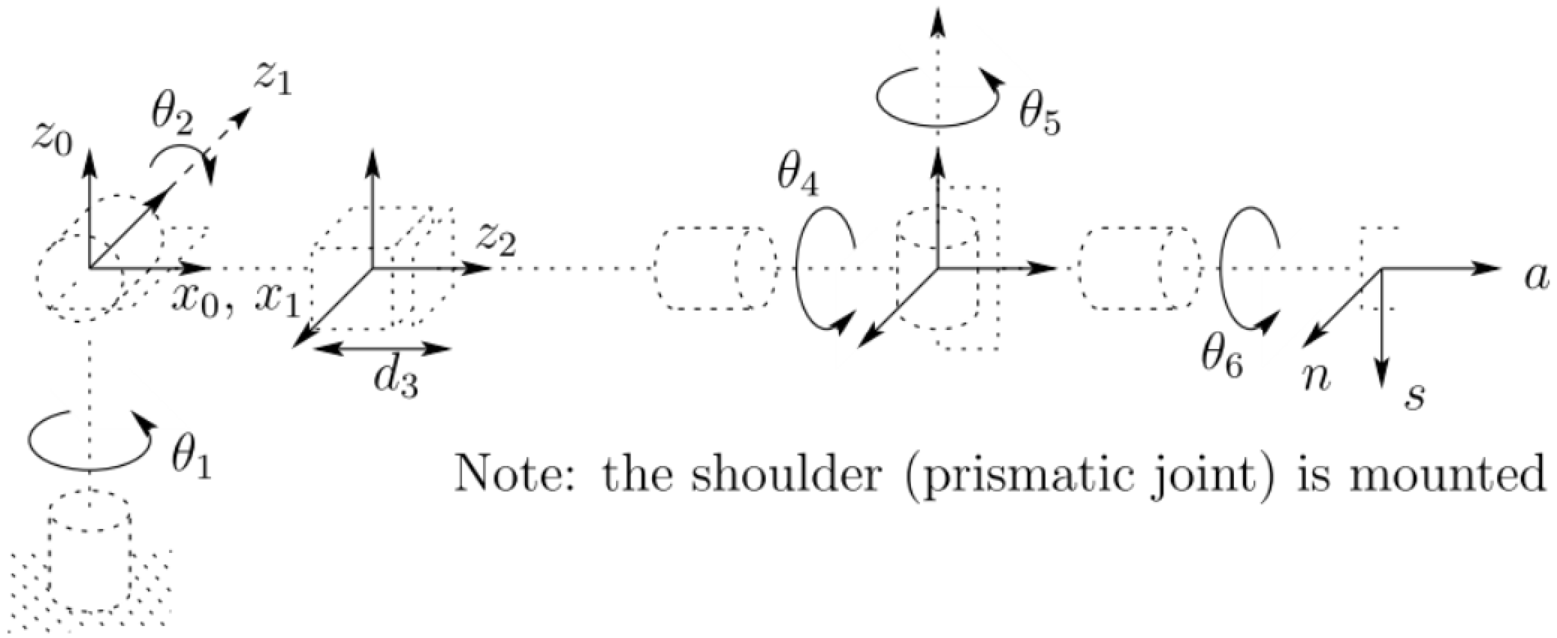
where

$$\begin{aligned}
r_{11} &= c_1c_4c_5c_6 - c_1s_4s_6 + s_1s_5c_6 \\
r_{21} &= s_1c_4c_5c_6 - s_1s_4s_6 - c_1s_5c_6 \\
r_{31} &= -s_4c_5c_6 - c_4s_6 \\
r_{12} &= -c_1c_4c_5s_6 - c_1s_4c_6 - s_1s_5c_6 \\
r_{22} &= -s_1c_4c_5s_6 - s_1s_4s_6 + c_1s_5c_6 \\
r_{32} &= s_4c_5c_6 - c_4c_6 \\
r_{13} &= c_1c_4s_5 - s_1c_5 \\
r_{23} &= s_1c_4s_5 + c_1c_5 \\
r_{33} &= -s_4s_5 \\
d_x &= c_1c_4s_5d_6 - s_1c_5d_6 - s_1d_3 \\
d_y &= s_1c_4s_5d_6 + c_1c_5d_6 + c_1d_3 \\
d_z &= -s_4s_5d_6 + d_1 + d_2.
\end{aligned}$$

- Notice how most of the complexity of the forward kinematics for this manipulator results from the orientation of the end-effector while the expression for the arm position is fairly simple.
- The spherical wrist assumption not only simplifies the derivation of the forward kinematics here, but will also greatly simplify the inverse kinematics problem.

Stanford Manipulator

- Consider now the Stanford Manipulator shown in Figure.
- This manipulator is an example of a spherical (RRP) manipulator with a spherical wrist.
- This manipulator has an offset in the shoulder joint that slightly complicates both the forward and inverse kinematics problems.
- We first establish the joint coordinate frames using the *D-H* convention as shown.
- The link parameters are shown in the Table.



Note: the shoulder (prismatic joint) is mounted wrong.

DH coordinate frame assignment for the Stanford manipulator.

DH parameters for Stanford Manipulator

Link	d_i	a_i	α_i	θ_i
1	0	0	-90	$\star$
2	d_2	0	$+90$	$\star$
3	$\star$	0	0	0
4	0	0	-90	$\star$
5	0	0	$+90$	$\star$
6	d_6	0	0	$\star$

* joint variable

Stanford Manipulator

- It is straightforward to compute the matrices A_i as:

$$A_1 = \begin{bmatrix} c_1 & 0 & -s_1 & 0 \\ s_1 & 0 & c_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

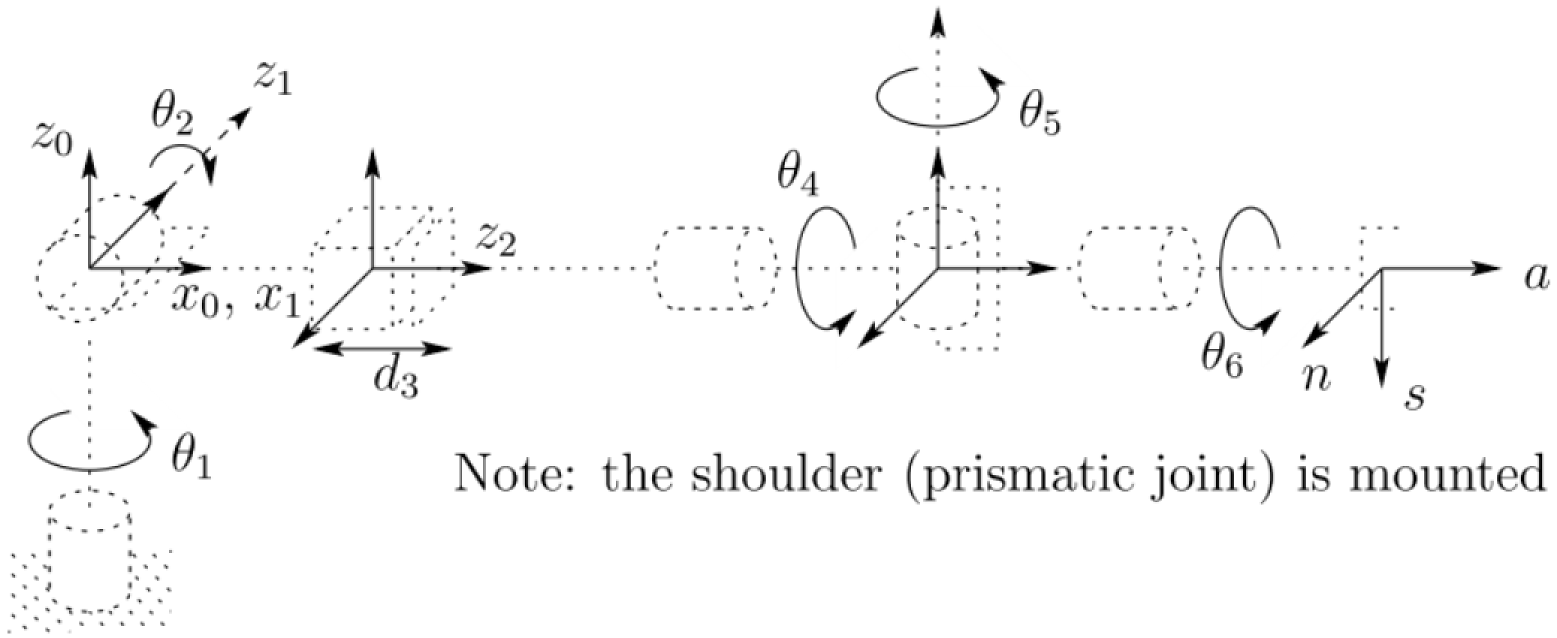
$$A_2 = \begin{bmatrix} c_2 & 0 & s_2 & 0 \\ s_2 & 0 & -c_2 & 0 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} c_4 & 0 & -s_4 & 0 \\ s_4 & 0 & c_4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_5 = \begin{bmatrix} c_5 & 0 & s_5 & 0 \\ s_5 & 0 & -c_5 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_6 = \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Note: the shoulder (prismatic joint) is mounted wrong.

DH coordinate frame assignment for the Stanford manipulator.

DH parameters for Stanford Manipulator

Link	d_i	a_i	α_i	θ_i
1	0	0	−90	⋆
2	d_2	0	+90	⋆
3	⋆	0	0	0
4	0	0	−90	⋆
5	0	0	+90	⋆
6	d_6	0	0	⋆

* joint variable

Stanford Manipulator

- It is straightforward to compute the matrices A_i as:

$$A_1 = \begin{bmatrix} c_1 & 0 & -s_1 & 0 \\ s_1 & 0 & c_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad A_5 = \begin{bmatrix} c_5 & 0 & s_5 & 0 \\ s_5 & 0 & -c_5 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} c_2 & 0 & s_2 & 0 \\ s_2 & 0 & -c_2 & 0 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad A_6 = \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} c_4 & 0 & -s_4 & 0 \\ s_4 & 0 & c_4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- $T_6{}^0$ is then given as:

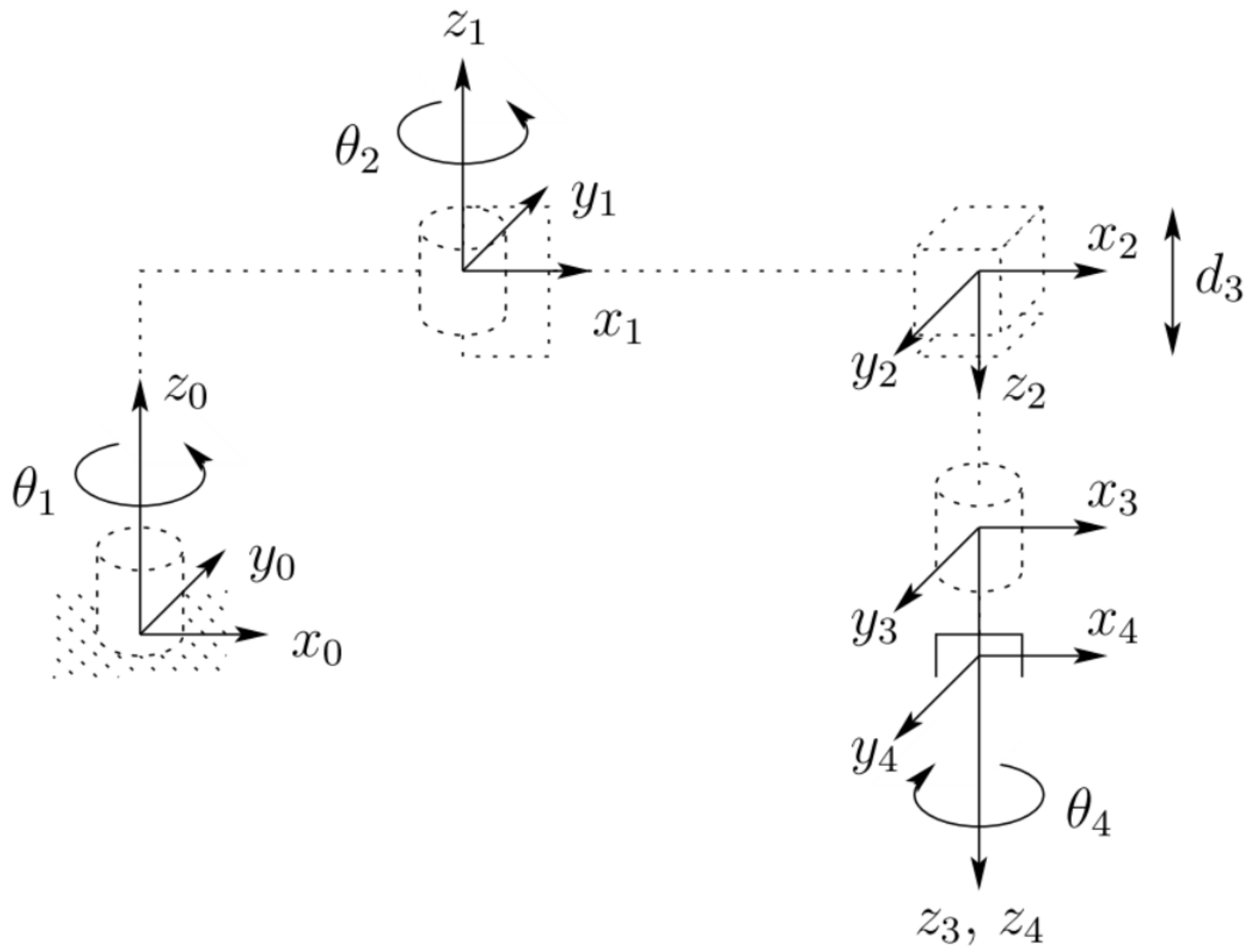
$$\begin{aligned} T_6^0 &= A_1 \cdots A_6 \\ &= \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

where

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

SCARA Manipulator

- As another example of the general procedure, consider the SCARA manipulator.
- This manipulator consists of an RRP arm and a one degree-of-freedom wrist, whose motion is a roll about the vertical axis.



DH coordinate frame assignment for the SCARA manipulator.

SCARA Manipulator

- The first step is to locate and label the joint axes as shown.
- Since all joint axes are parallel we have some freedom in the placement of the origins. The origins are placed as shown for convenience.
- We establish the x_0 axis in the plane of the page as shown. This is completely arbitrary and only affects the zero configuration of the manipulator, that is, the position of the manipulator when $\vartheta_1 = 0$.
- The joint parameters are given in Table

Joint parameters for SCARA

Link	a_i	α_i	d_i	θ_i
1	a_1	0	0	$\star$
2	a_2	180	0	$\star$
3	0	0	$\star$	0
4	0	0	d_4	$\star$

* joint variable

SCARA Manipulator

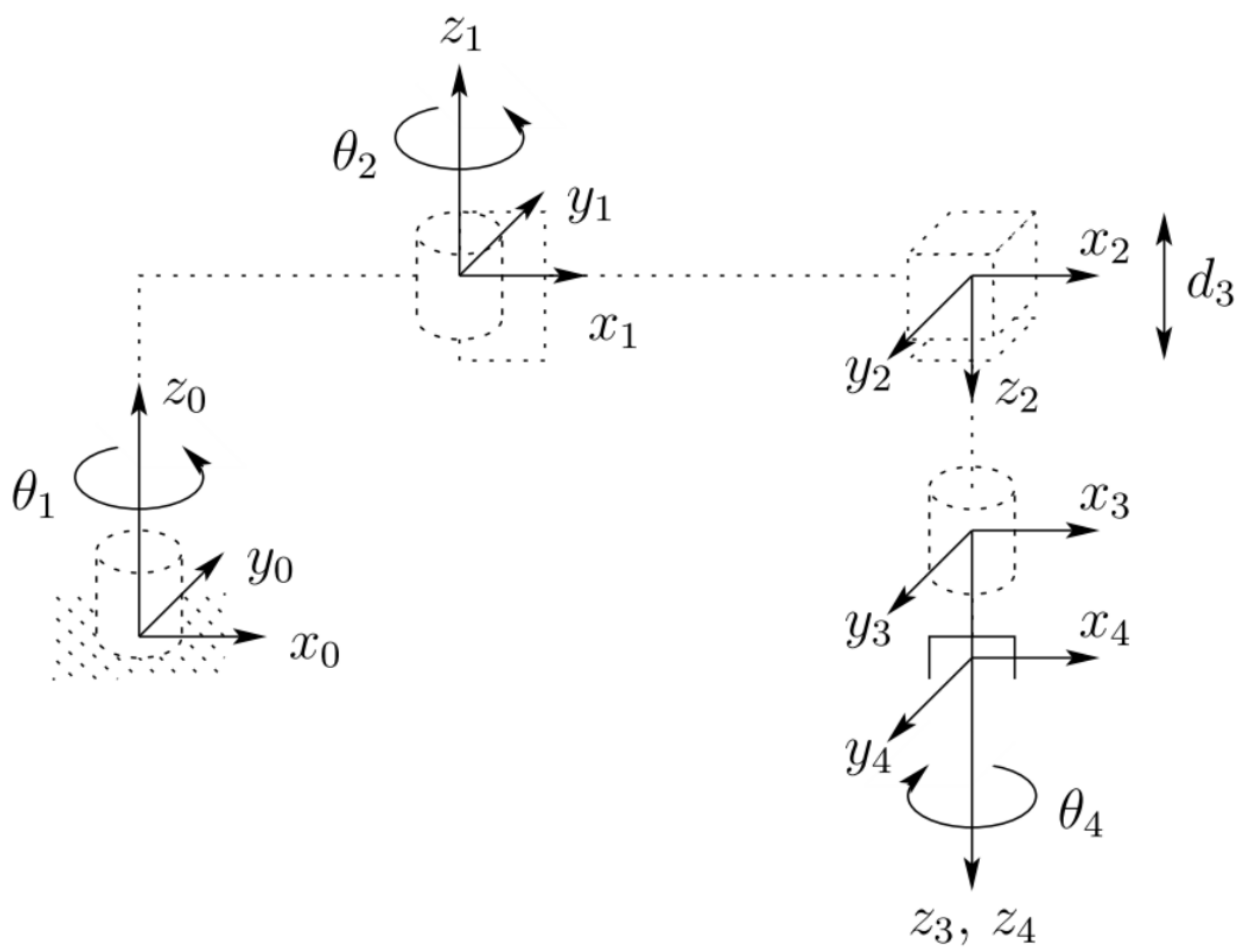
- A-matrices are as follow:

$$A_1 = \begin{bmatrix} c_1 & -s_1 & 0 & a_1c_1 \\ s_1 & c_1 & 0 & a_1s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} c_2 & s_2 & 0 & a_2c_2 \\ s_2 & -c_2 & 0 & a_2s_2 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad A_4 = \begin{bmatrix} c_4 & -s_4 & 0 & 0 \\ s_4 & c_4 & 0 & 0 \\ 0 & 0 & 1 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

- The forward kinematic equations are therefore given by

$$T_4^0 = A_1 \cdots A_4 = \begin{bmatrix} c_{12}c_4 + s_{12}s_4 & -c_{12}s_4 + s_{12}c_4 & 0 & a_1c_1 + a_2c_{12} \\ s_{12}c_4 - c_{12}s_4 & -s_{12}s_4 - c_{12}c_4 & 0 & a_1s_1 + a_2\,s_{12} \\ 0 & 0 & -1 & -d_3 - d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



DH coordinate frame assignment for the SCARA manipulator.

Joint parameters for SCARA

Link	a_i	α_i	d_i	θ_i
1	a_1	0	0	$\star$
2	a_2	180	0	$\star$
3	0	0	$\star$	0
4	0	0	d_4	$\star$

* joint variable

Inverse Kinematics

- The inverse problem of finding the joint variables in terms of the end-effector position and orientation.
- This is the problem of inverse kinematics, and it is, in general, more difficult than the forward
- kinematics problem.
- We begin by formulating the general inverse kinematics problem.
- We describe the principle of kinematic decoupling and how it can be used to
- Simplify the inverse kinematics of most modern manipulators.
- Using kinematic decoupling, we can consider the position and orientation problems independently. We describe a geometric approach for solving the positioning problem, while we exploit the Euler angle parameterization to solve the orientation problem.

Inverse Kinematics -The General Inverse Kinematics Problem

The general problem of inverse kinematics can be stated as follows. Given a 4 $\times$ 4 homogeneous transformation

$$H = \begin{bmatrix} R & o \\ 0 & 1 \end{bmatrix} \in SE(3) \tag{4.1}$$

with R ∈ SO(3), find (one or all) solutions of the equation

$$T_n^0(q_1, \ldots, q_n) = H \tag{4.2}$$

where

$$T_n^0(q_1, \ldots, q_n) = A_1(q_1) \cdots A_n(q_n). \tag{4.3}$$

Here, H represents the desired position and orientation of the end-effector, and our task is to find the values for the joint variables $q_1, \ldots, q_n$ so that $T_n^0(q_1, \ldots, q_n) = H$.

Inverse Kinematics -The General Inverse Kinematics Problem

- Furthermore, because these forward kinematic equations are in general complicated nonlinear functions of the joint variables, the solutions may be difficult to obtain even when they exist.

- In solving the inverse kinematics problem we are most interested in finding a **closed form solution** of the equations rather than a numerical solution. Finding a closed form solution means finding an explicit relationship:

$$q_k = f_k(h_{11}, \ldots, h_{34}), \qquad k = 1, \ldots, n. \tag{4.6}$$

- Closed form solutions are preferable for two reasons.
  - First, in certain applications, such as tracking a welding seam whose location is provided by a vision system, the inverse kinematic equations must be solved at a rapid rate, say every 20 milliseconds, and having closed form expressions rather than an iterative search is a practical necessity.
  - Second, the kinematic equations in general have multiple solutions. Having closed form solutions allows one to develop rules for choosing a particular solution among several.

Inverse Kinematics -The General Inverse Kinematics Problem

- Equation (4.2) results in twelve nonlinear equations in n unknown variables, which can be written as

$$T_{ij}(q_1, \ldots, q_n) = h_{ij}, \qquad i = 1, 2, 3, \quad j = 1, \ldots, 4 \qquad (4.4)$$

where $T_{\text{ij}}, h_{\text{ij}}$ refer to the twelve nontrivial entries of T_n^0 and H, respectively.
(Since the bottom row of both T_n^0 and H are (0,0,0,1), four of the sixteen equations represented by (4.2) are trivial.)

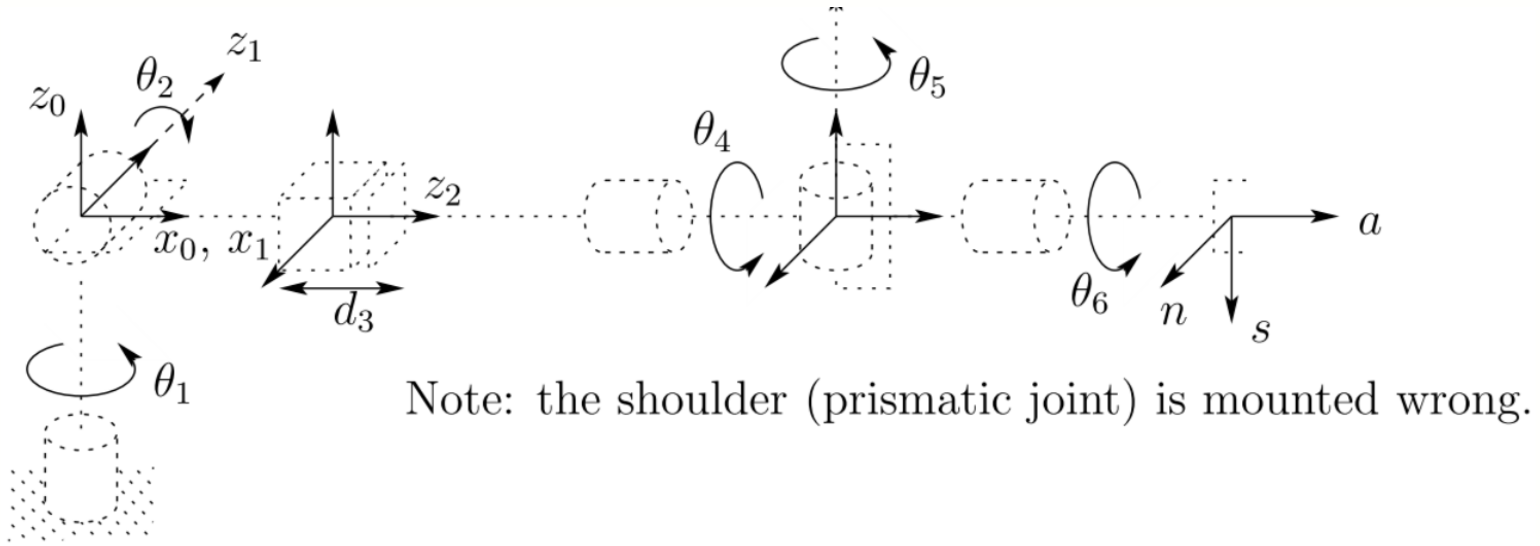
Inverse Kinematics -The General Inverse Kinematics Problem

Example 1

- Recall the Stanford manipulator, suppose that the desired position and orientation of the final frame are given by

$$H = \begin{bmatrix} r_{11} & r_{12} & r_{13} & o_x \\ r_{21} & r_{22} & r_{23} & o_y \\ r_{31} & r_{32} & r_{33} & o_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (4.5)$$

- To find the corresponding joint variables $\vartheta_1, \vartheta_2, \vartheta_3, \vartheta_4, \vartheta_5, \vartheta_6$,we must solve the following simultaneous set of nonlinear trigonometric equations



Note: the shoulder (prismatic joint) is mounted wrong.

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

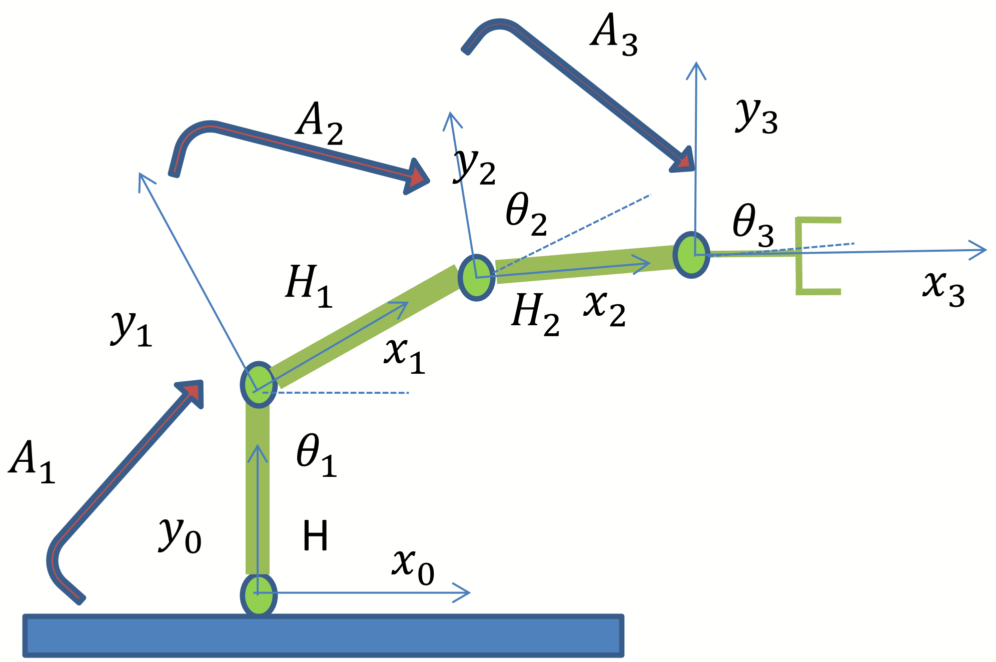
Inverse Kinematics -The General Inverse Kinematics Problem

- Equation (4.2) results in twelve nonlinear equations in n unknown variables, which can be written as

where $T_{\mathrm{ij}}, h_{\mathrm{ij}}$ refer to the twelve nontrivial entries of T_n^0 and H, respectively.
(Since the bottom row of both T_n^0 and H are (0,0,0,1), four of the sixteen equations represented by (4.2) are trivial)

- The equations in the preceding example are, of course, much too difficult to solve directly in closed form. This is the case for most robot arms.
- 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 <u>may or may not have a solution</u>.
- **Even if a solution exists, it may or may not be unique.**

Example – Inverse kinematics



$$A_1 = \begin{pmatrix} \cos\theta_1 & -sin\theta_1 & 0 \\ sin\theta_1 & \cos\theta_1 & H \\ 0 & 0 & 1 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} \cos\theta_2 & -sin\theta_2 & H_1 \\ sin\theta_2 & \cos\theta_2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$A_3 = \begin{pmatrix} \cos\theta_3 & -sin\theta_3 & H_2 \\ sin\theta_3 & \cos\theta_3 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

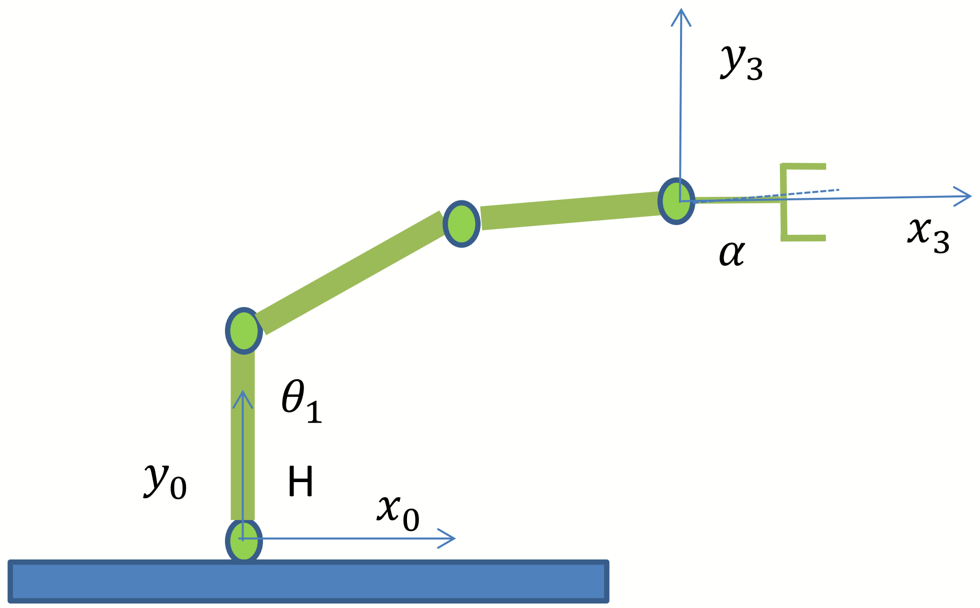
$$T_3 = A_1\, A_2\, A_3$$

$$T_3 = \begin{pmatrix} \cos(\theta_1 + \theta_2 + \theta_3) & -sin(\theta_1 + \theta_2 + \theta_3) & \cos(\theta_1 + \theta_2) H_2 + \cos(\theta_1) H_1 \\ sin(\theta_1 + \theta_2 + \theta_3) & \cos(\theta_1 + \theta_2 + \theta_3) & \mathrm{H} + \sin(\theta_1 + \theta_2) H_2 + \sin(\theta_1) H_1 \\ 0 & 0 & 1 \end{pmatrix}$$

How we can compute the inverse kinematics?

Example – Inverse kinematics

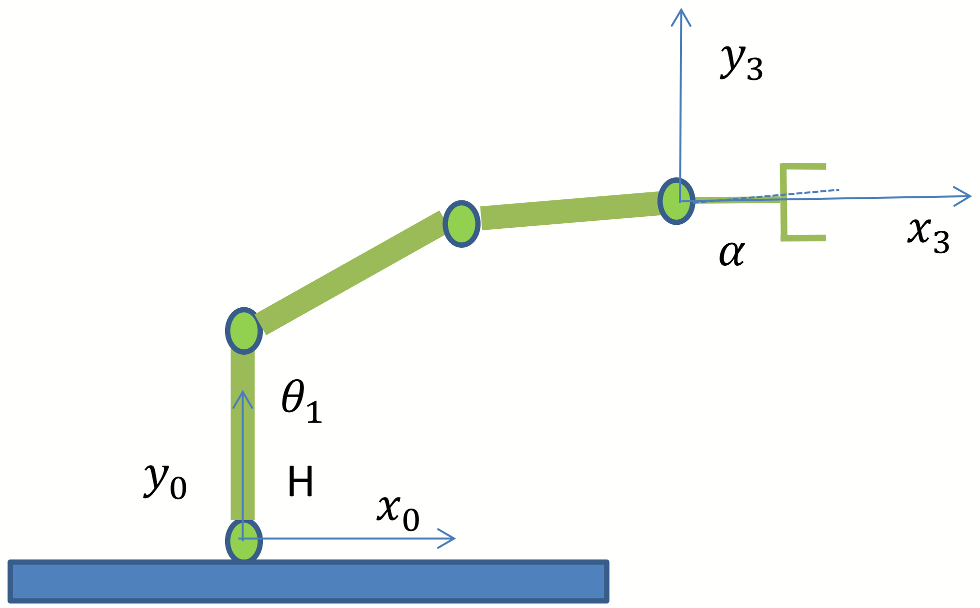
Computing end-effector coordinates with respect to the joint T^* is the end-effector transformation



$$T^* = \begin{pmatrix} \cos\alpha & -\sin\alpha & x \\ sin\ \alpha & \cos\alpha & y \\ 0 & 0 & 1 \end{pmatrix}$$

$$T_3 = \begin{pmatrix} \cos(\theta_1 + \theta_2 + \theta_3) & -sin(\theta_1 + \theta_2 + \theta_3) & \cos(\theta_1 + \theta_2) H_2 + \cos(\theta_1) H_1 \\ sin(\theta_1 + \theta_2 + \theta_3) & \cos(\theta_1 + \theta_2 + \theta_3) & \mathrm{H} + \sin(\theta_1 + \theta_2) H_2 + \sin(\theta_1) H_1 \\ 0 & 0 & 1 \end{pmatrix}$$

Example – Inverse kinematics



With $T^* = T_3$ we can obtain:

$$\begin{cases} \alpha = \theta_1 + \theta_2 + \theta_3 \\ x = \cos(\theta_1 + \theta_2) H_2 + \cos(\theta_1) H_1 \\ y - H = \sin(\theta_1 + \theta_2) H_2 + \sin(\theta_1) H_1 \end{cases}$$

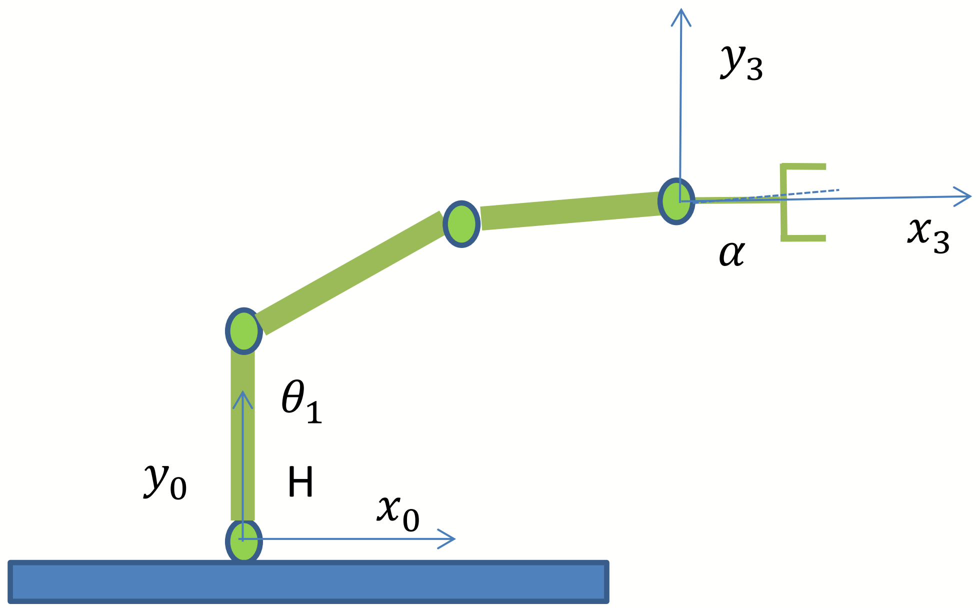
By summing the square values:

$$x^2 + (y-H)^2 = H_2^2 + H_1^2 + 2H_1 H_2 \cos\theta_2$$

$$\begin{cases} \cos\theta_2 = (x^2 + (y-H)^2 - H_2^2 - H_1^2)/2H_1 H_2 \\ \sin\theta_2 = \pm\sqrt{1 - (\cos\theta_2)^2} \end{cases}$$

Then using θ_2 it is possible to compute the values of θ_1 e θ_3

Example – Inverse kinematics



$$T^* = \begin{pmatrix} \cos\alpha & -\sin\alpha & x \\ \sin\alpha & \cos\alpha & y \\ 0 & 0 & 1 \end{pmatrix}$$

$$T_3 = \begin{pmatrix} \cos(\theta_1 + \theta_2 + \theta_3) & -\sin(\theta_1 + \theta_2 + \theta_3) & \cos(\theta_1 + \theta_2) H_2 + \cos(\theta_1) H_1 \\ \sin(\theta_1 + \theta_2 + \theta_3) & \cos(\theta_1 + \theta_2 + \theta_3) & \mathrm{H} + \sin(\theta_1 + \theta_2) H_2 + \sin(\theta_1) H_1 \\ 0 & 0 & 1 \end{pmatrix}$$

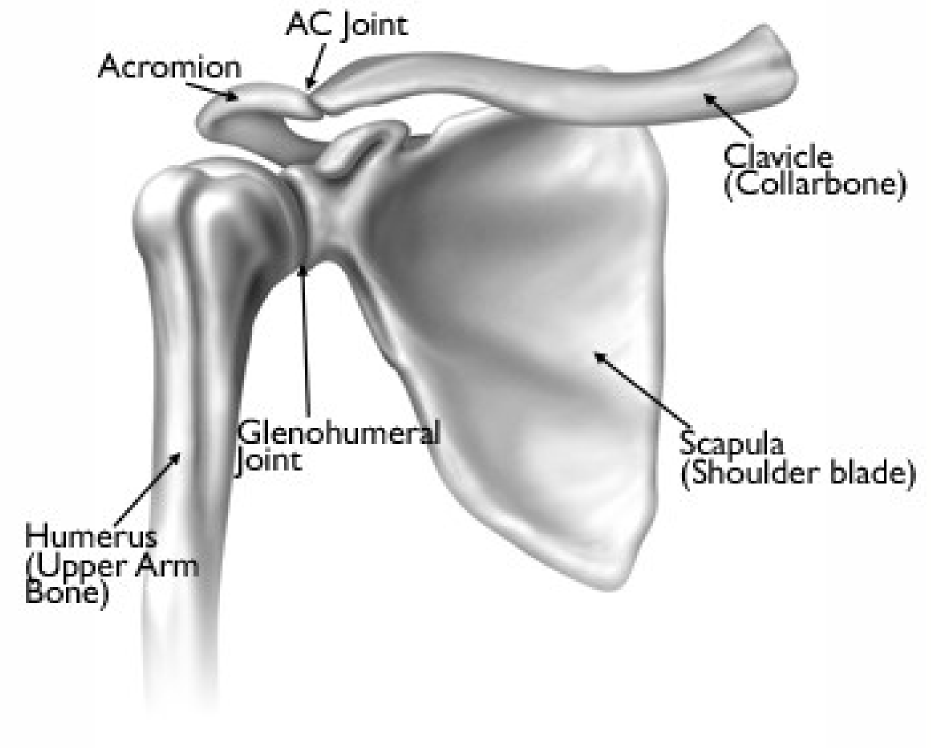
With $T^* = T_3$ we can obtain:

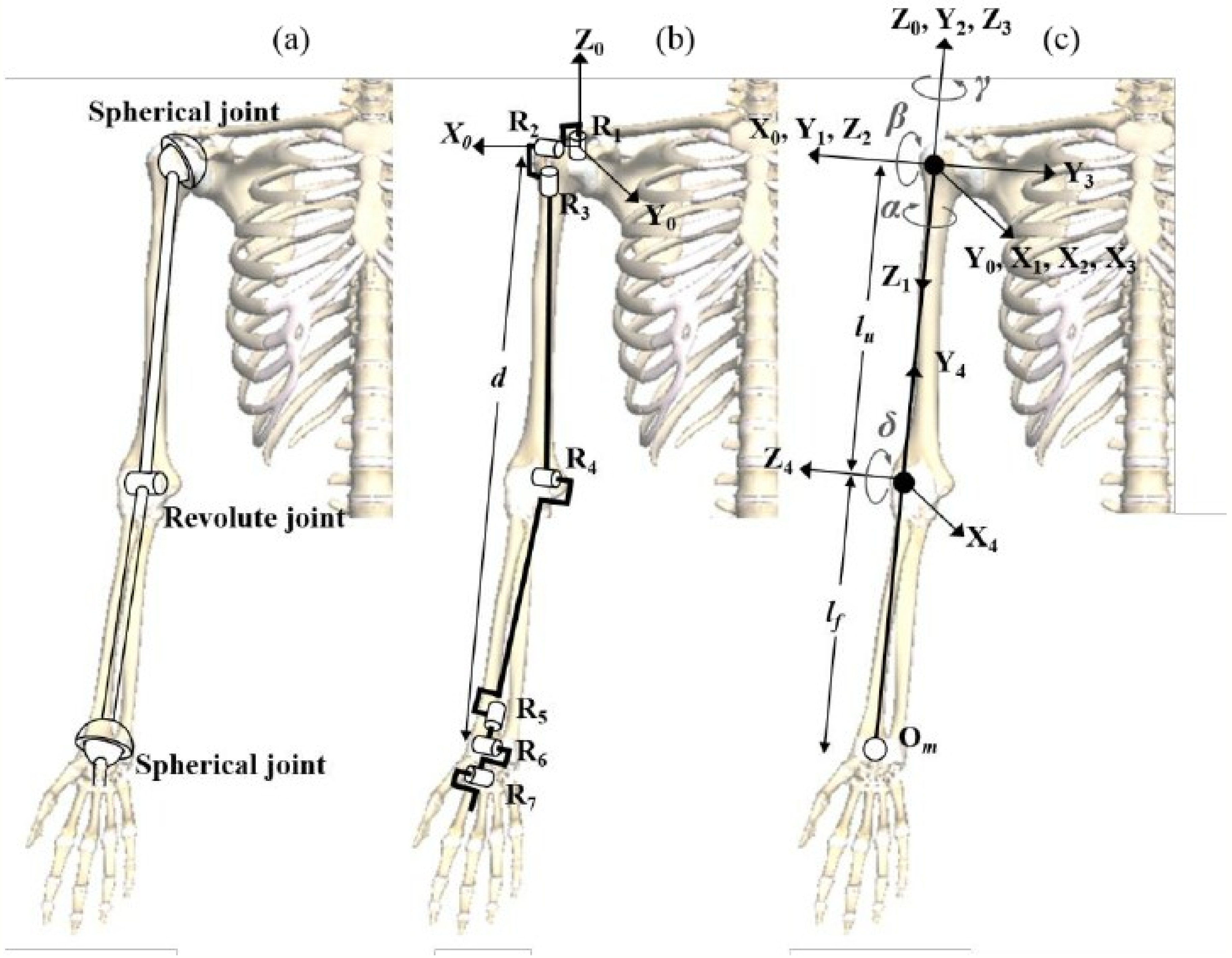
$$\begin{cases} \alpha = \theta_1 + \theta_2 + \theta_3 \\ x = \cos(\theta_1 + \theta_2) H_2 + \cos(\theta_1) H_1 \\ y - H = \sin(\theta_1 + \theta_2) H_2 + \sin(\theta_1) H_1 \end{cases}$$

Kinematic model of the human arm

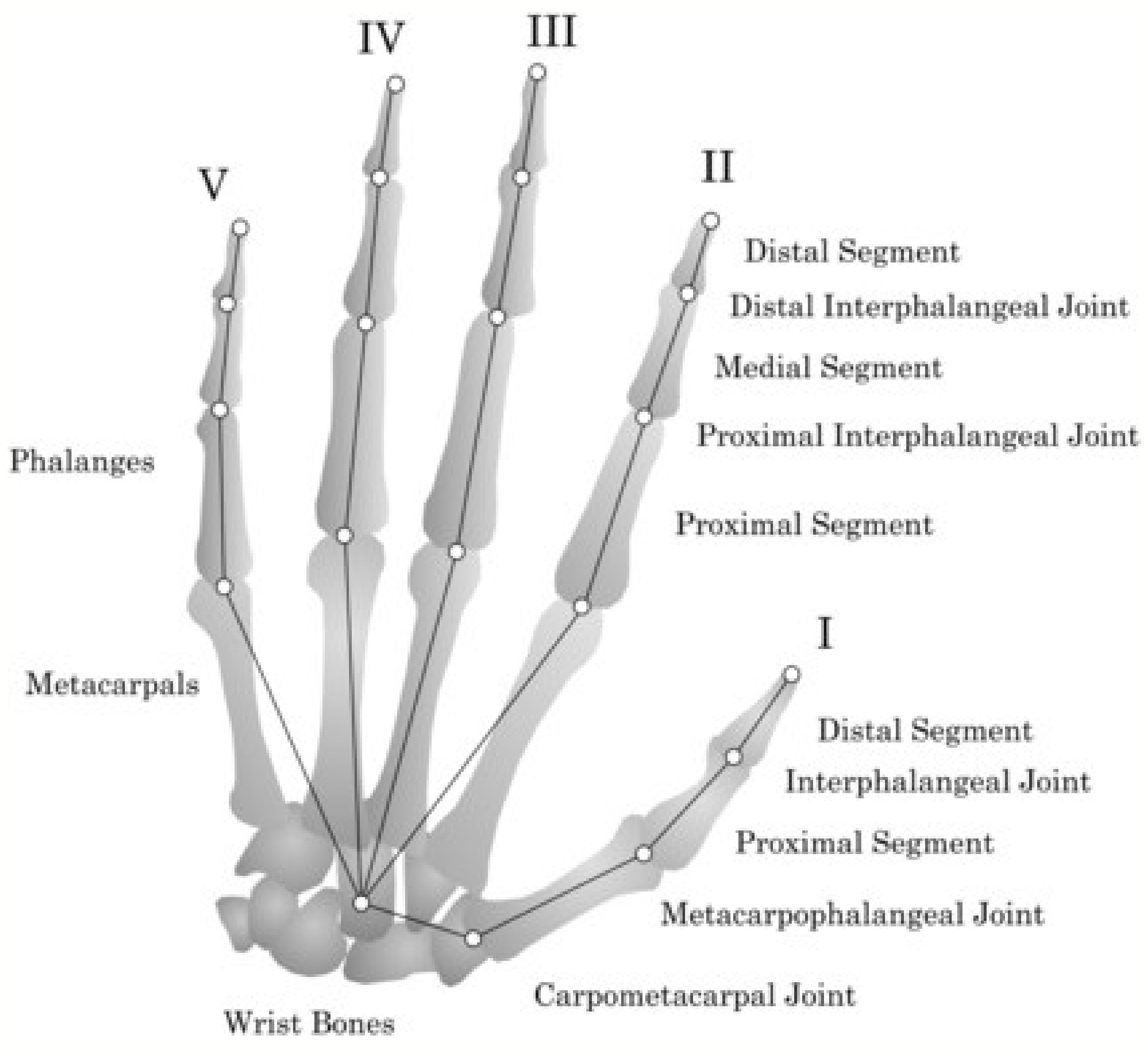
Typically we have used a 7-DOF model for the upper extremity:

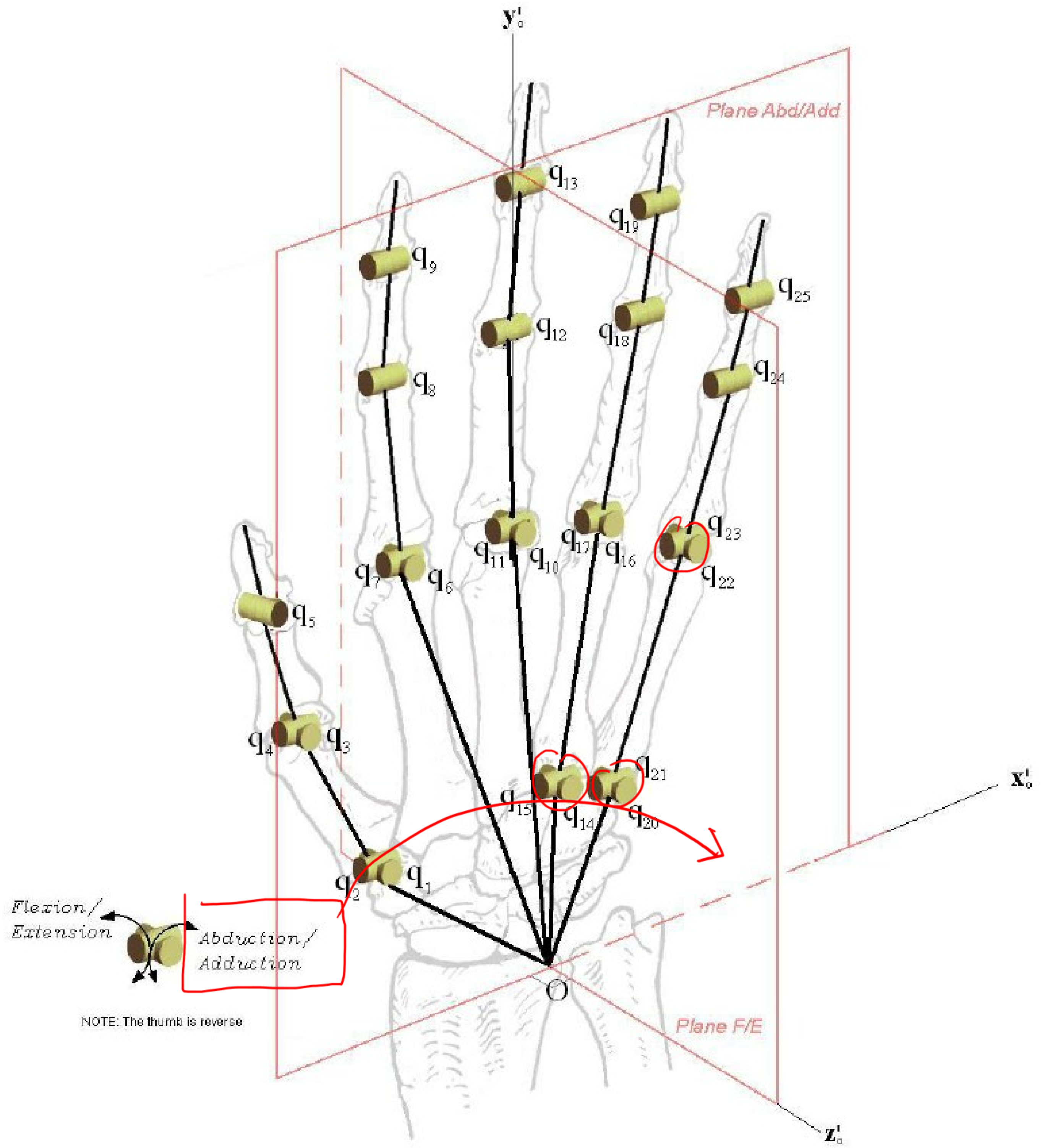
- Wrist (3-DOF),
- Elbow (1-DOF),
- Shoulder (3-DOF) + (2-DOF while accounting for the glohumeral joint and the scapula translational joint)





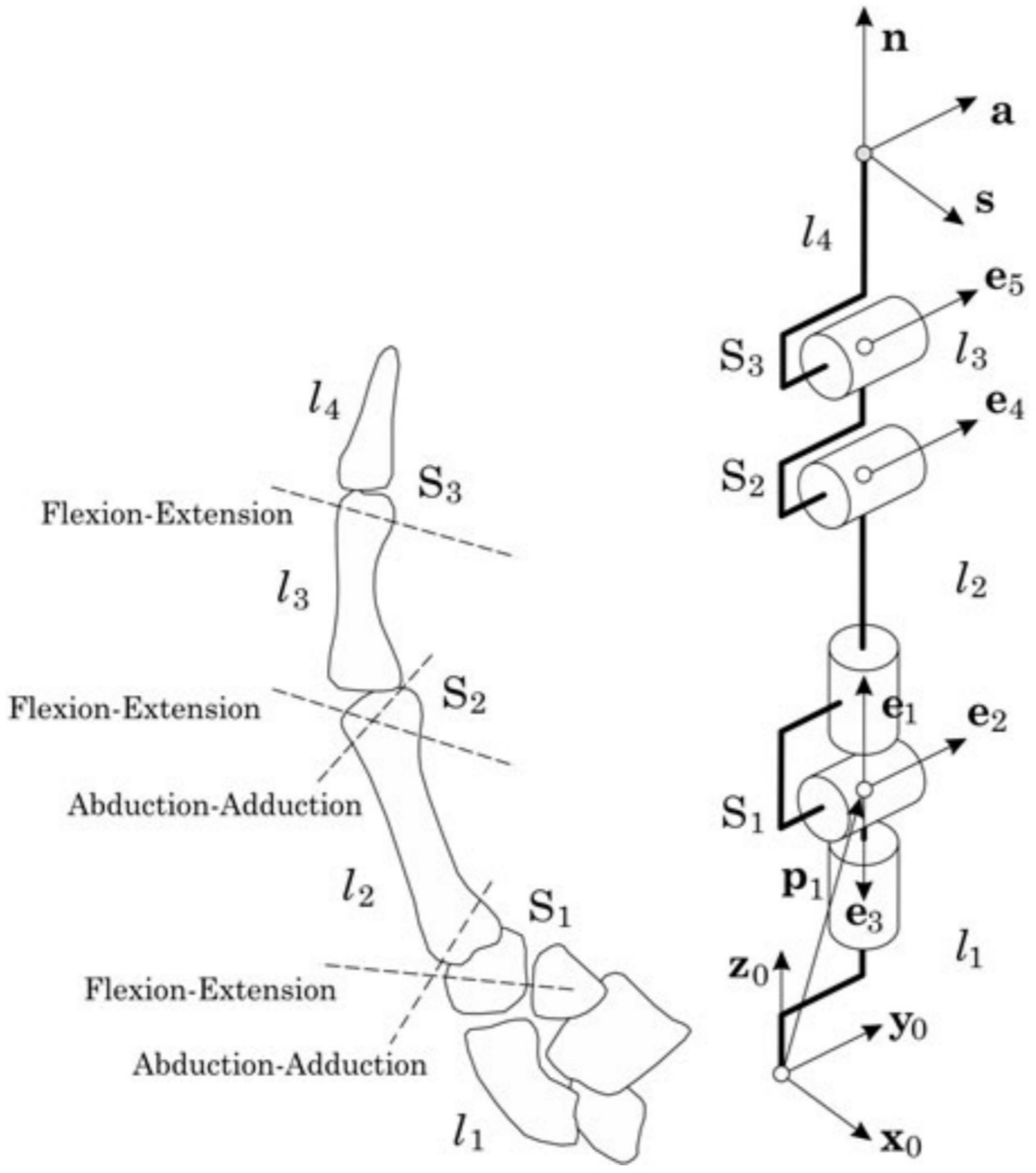
Kinematic model of the human hand (25-DOF)

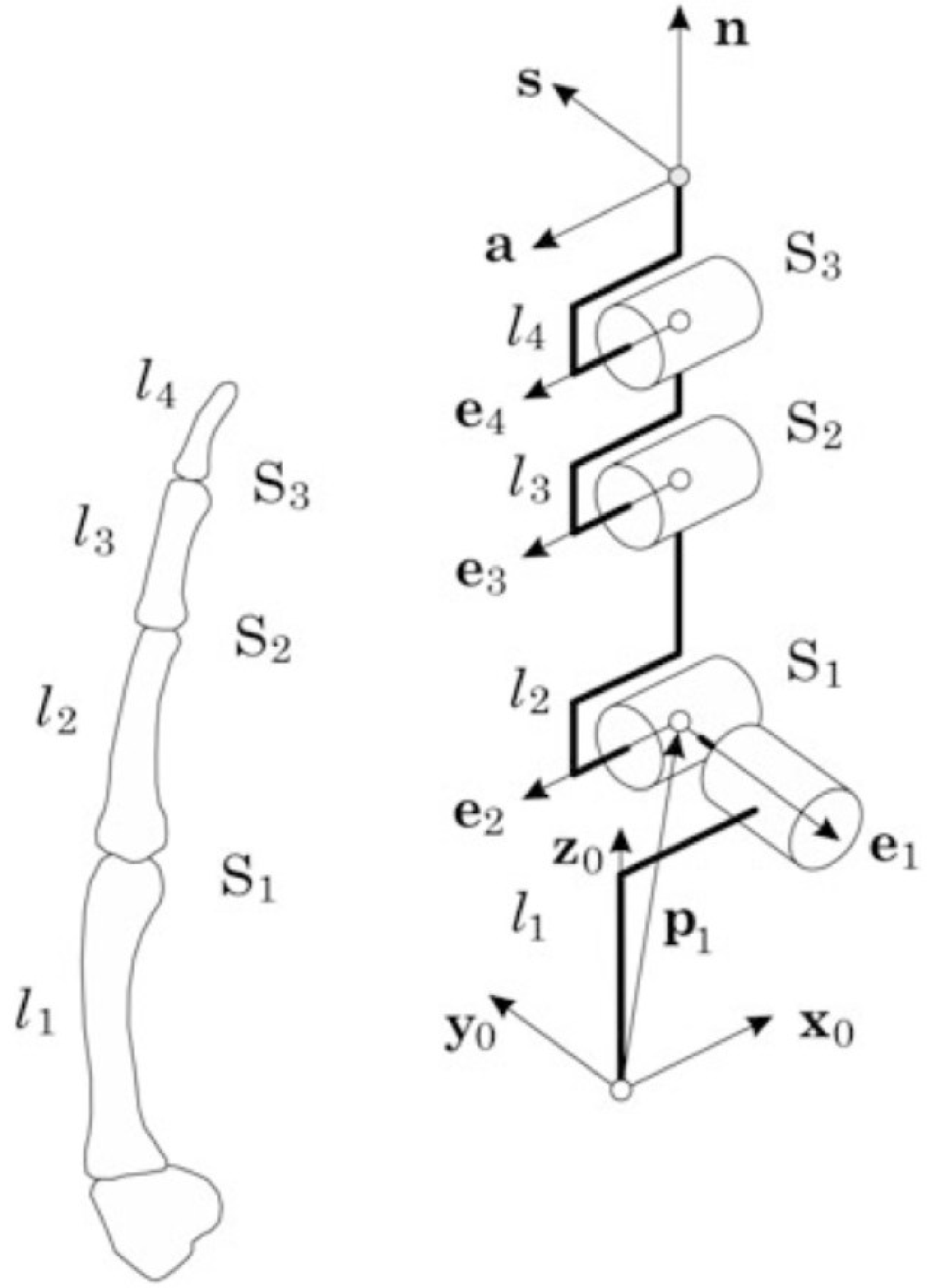




Lenarčič J., Bajd T., Stanišić M.M. (2013) Kinematic Model of the Human Hand. In: Robot Mechanisms. Intelligent Systems, Control and Automation: Science and Engineering, vol 60. Springer, Dordrecht. https://doi.org/10.1007/978-94-007-4522-3_10

Kinematic model of the human thumb and fingers





Lenarčič J., Bajd T., Stanišić M.M. (2013) Kinematic Model of the Human Hand. In: Robot Mechanisms. Intelligent Systems, Control and Automation: Science and Engineering, vol 60. Springer, Dordrecht. https://doi.org/10.1007/978-94-007-4522-3_10

Kinematic model of the human body

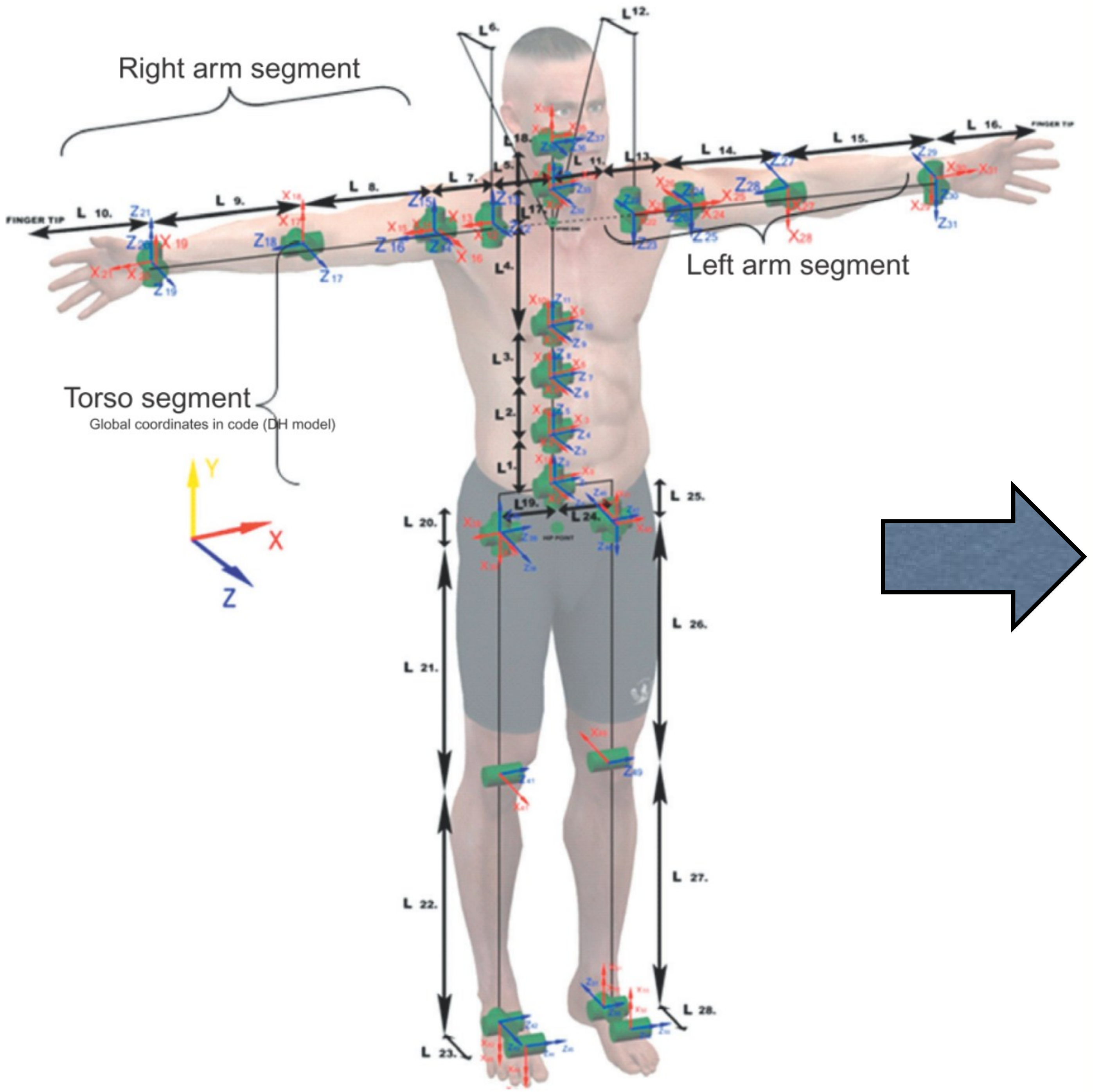
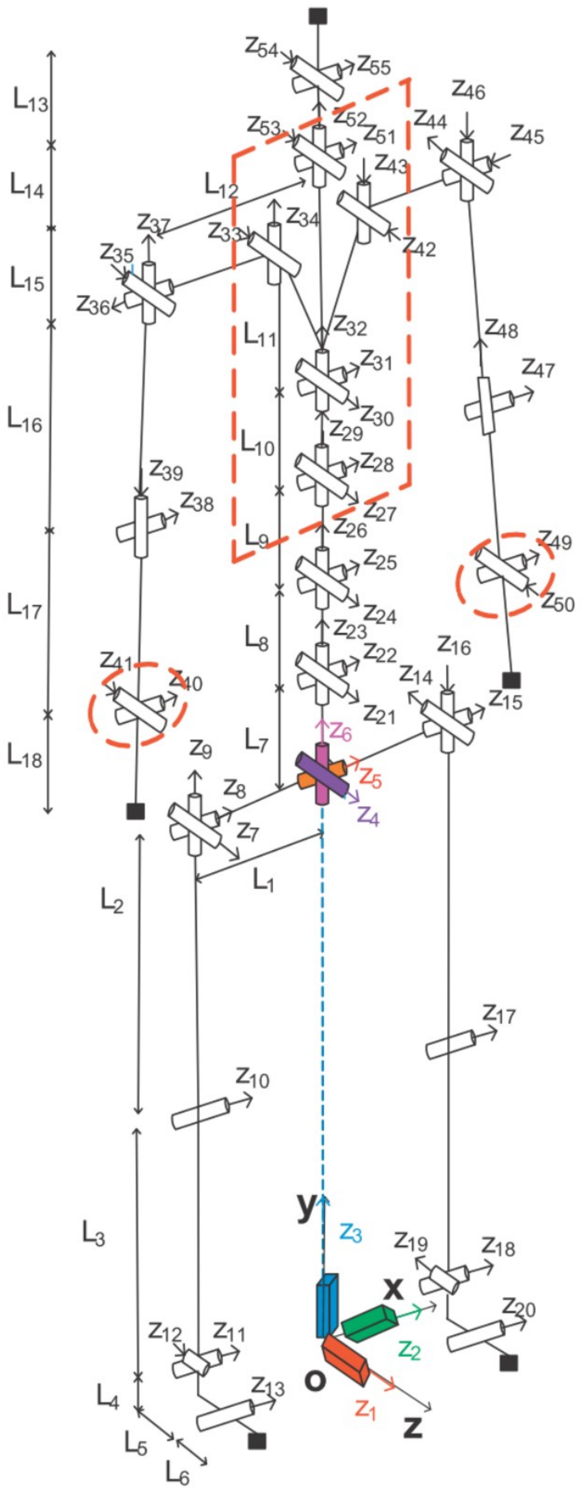


Table 1 DH table for arms, legs, and neck

#	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



Karim A. Abdel-Malek, Jasbir Singh Arora,Chapter 2 - Human Modeling: Kinematics,Editor(s): Karim A. Abdel-Malek, Jasbir Singh Arora,Human Motion Simulation,Academic Press,2013,Pages 7-40,ISBN 9780124051904,https://doi.org/10.1016/B978-0-12-405190-4.00002-7. (http://www.sciencedirect.com/science/article/pii/B9780124051904000027)

Robot kinematics and differential kinematics

Kinematics

$$x = k(q)$$
$$q = k^{-1}(x)$$

$$x = \begin{bmatrix} x \\ y \\ z \\ \varphi \\ \vartheta \\ \psi \end{bmatrix} \quad q = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \end{bmatrix}$$

$k(\cdot) = direct\ kinematics$

Differential Kinematics

$$\dot{x} = J(q)\dot{q}$$
$$\dot{q} = J^{-1}(q)\dot{x}$$

$J(q) = \text{Jacobian matrix}$

Differential kinematics

Geometric Jacobian = transformation matrix depending on the current robot configuration

$$v = \begin{bmatrix} \dot{p} \\ \omega \end{bmatrix} = J(q)\dot{q}$$

$J(q)$= geometric Jacobian
$\dot{p}$= linear velocity of the end effector
ω= angular velocity of the end effector
$\dot{q}$= joint velocity

$$J = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} = \begin{pmatrix} \frac{\partial y_1}{\partial x_1} & \cdots & \frac{\partial y_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_m}{\partial x_1} & \cdots & \frac{\partial y_m}{\partial x_n} \end{pmatrix}$$

A Jacobian is the matrix equivalent of the derivative - the derivative of a vector-valued function of a vector with respect to a vector.

If $y = f(x)$ and $x \in R^n$ and $y \in R^m$ then the Jacobian is the $m \times n$ matrix

Differential kinematics

To find the joint velocities given the end effector velocity in operational space

$$v = \begin{bmatrix} \dot{p} \\ \omega \end{bmatrix} = J(q)\dot{q}$$

$$\dot{q} = J^{-1}(q)v = J^{-1}(q)\begin{bmatrix} \dot{p} \\ \omega \end{bmatrix}$$

Integral numerical methods allows to find the q vector from the vector of joint velocities

$J(q)$= geometric Jacobian
J^{-1}= inverse Jacobian
$\dot{p}$= linear velocity of the end effector
ω= angular velocity of the end effector
$\dot{q}$= joint velocity

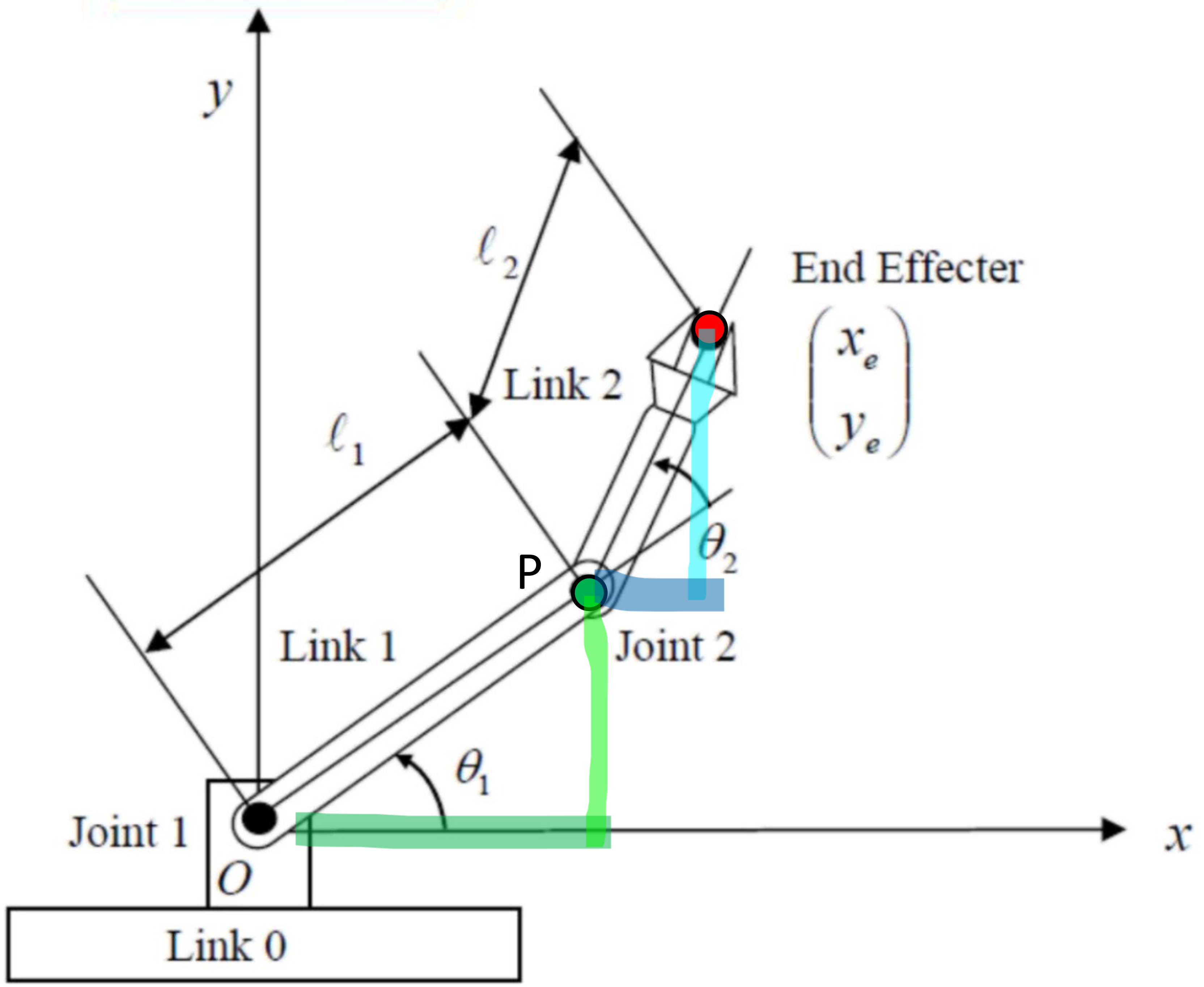
Differential kinematics

- (x_e, y_e) End Effector
- (x_{0-P}, y_{0-P}) Point P

  - $x_{0-P}(\vartheta_1, \vartheta_2) = l_1 \cos(\vartheta_1)$
  - $y_{0-P}(\vartheta_1, \vartheta_2) = l_1 \sin(\vartheta_1)$

  - $x_{P-e}(\vartheta_1, \vartheta_2) = l_2 \cos(\vartheta_1 + \vartheta_2)$
  - $y_{P-e}(\vartheta_1, \vartheta_2) = l_2 \sin(\vartheta_1 + \vartheta_2)$

  - $x_e(\vartheta_1, \vartheta_2) = x_{0-P}(\vartheta_1, \vartheta_2) + x_{P-e}(\vartheta_1, \vartheta_2)$
    $= l_1 \cos(\vartheta_1) + l_2 \cos(\vartheta_1 + \vartheta_2)$
  - $y_e(\vartheta_1, \vartheta_2) = y_{0-P}(\vartheta_1, \vartheta_2) + y_{P-e}(\vartheta_1, \vartheta_2)$
    $= l_1 \sin(\vartheta_1) + l_2 \sin(\vartheta_1 + \vartheta_2)$



Differential kinematics

$$\begin{cases} x_e(\vartheta_1, \vartheta_2) = l_1\cos(\vartheta_1) + l_2\cos(\vartheta_1 + \vartheta_2) \\ y_e(\vartheta_1, \vartheta_2) = l_1\sin(\vartheta_1) + l_2\sin(\vartheta_1 + \vartheta_2) \end{cases}$$

$$\begin{cases} x_{0-P}(\vartheta_1, \cancel{\vartheta_2}) = l_1\cos(\vartheta_1) \\ y_{0-P}(\vartheta_1, \cancel{\vartheta_2}) = l_1\sin(\vartheta_1) \end{cases}$$

$$dx_{0-P} = \frac{\partial x_{0-P}(\theta_1,\theta_2)}{\partial\theta_1} d\theta_1 + \cancel{\frac{\partial x_{0-P}(\theta_1,\theta_2)}{\partial\theta_2}} d\theta_2 = -l_1\sin(\vartheta_1) d\theta_1$$

$$dy_{0-P} = \frac{\partial y_{0-P}(\theta_1,\theta_2)}{\partial\theta_1} d\theta_1 + \cancel{\frac{\partial y_{0-P}(\theta_1,\theta_2)}{\partial\theta_2}} d\theta_2 = l_1\cos(\vartheta_1)\, d\theta_1$$

$$\begin{cases} x_{P-e}(\vartheta_1, \vartheta_2) = l_2\cos(\vartheta_1 + \vartheta_2) \\ y_{P-e}(\vartheta_1, \vartheta_2) = l_2\sin(\vartheta_1 + \vartheta_2) \end{cases}$$

$\frac{\partial}{\partial x}(\cos(x+y)) = -\sin(x+y)$ $\quad$ $\frac{\partial}{\partial x}(\sin(x+y)) = \cos(x+y)$

$$dx_{P-e} = \frac{\partial x_{P-e}(\theta_1,\theta_2)}{\partial\theta_1} d\theta_1 + \frac{\partial x_{P-e}(\theta_1,\theta_2)}{\partial\theta_2} d\theta_2 = -l_2\sin(\vartheta_1 + \vartheta_2) d\theta_1 - l_2\sin(\vartheta_1 + \vartheta_2) d\theta_2$$

$$dy_{P-e} = \frac{\partial y_{P-e}(\theta_1,\theta_2)}{\partial\theta_1} d\theta_1 + \frac{\partial y_{P-e}(\theta_1,\theta_2)}{\partial\theta_2} d\theta_2 = l_2\cos(\vartheta_1 + \vartheta_2) d\theta_1 + l_2\cos(\vartheta_1 + \vartheta_2) d\theta_2$$

Differential kinematics

$$\begin{cases} dx_e = \dfrac{\partial x_e(\theta_1,\theta_2)}{\partial\theta_1} d\theta_1 + \dfrac{\partial x_e(\theta_1,\theta_2)}{\partial\theta_2} d\theta_2 \;= [\,-l_1\sin(\vartheta_1) - l_2\sin(\vartheta_1+\vartheta_2)\,] d\theta_1 - l_2\sin(\vartheta_1+\vartheta_2) d\theta_2 \\ dy_e = \dfrac{\partial y_e(\theta_1,\theta_2)}{\partial\theta_1} d\theta_1 + \dfrac{\partial y_e(\theta_1,\theta_2)}{\partial\theta_2} d\theta_2 \;= [\,l_1\cos(\vartheta_1) + l_2\cos(\vartheta_1+\vartheta_2)\,] d\theta_1 + l_2\cos(\vartheta_1+\vartheta_2) d\theta_2 \end{cases}$$

$$d\boldsymbol{x} = \boldsymbol{J}\cdot d\boldsymbol{q} \qquad\qquad d\boldsymbol{x} = \begin{pmatrix} dx_e \\ dy_e \end{pmatrix}, d\boldsymbol{q} = \begin{pmatrix} d\theta_1 \\ d\theta_2 \end{pmatrix}$$

$$\boldsymbol{J} = \begin{bmatrix} \dfrac{\partial x_e(\theta_1,\theta_2)}{\partial\theta_1} & \dfrac{\partial x_e(\theta_1,\theta_2)}{\partial\theta_2} \\ \dfrac{\partial y_e(\theta_1,\theta_2)}{\partial\theta_1} & \dfrac{\partial y_e(\theta_1,\theta_2)}{\partial\theta_2} \end{bmatrix} = \begin{bmatrix} -l_1\sin(\vartheta_1) - l_2\sin(\vartheta_1+\vartheta_2) & -l_2\sin(\vartheta_1+\vartheta_2) \\ l_1\cos(\vartheta_1) + l_2\cos(\vartheta_1+\vartheta_2) & l_2 cos(\vartheta_1+\vartheta_2) \end{bmatrix}$$

$$\frac{d\boldsymbol{x}_e}{dt} = \boldsymbol{J}\frac{d\boldsymbol{q}}{dt}, \quad or \quad \boldsymbol{v}_e = \boldsymbol{J}\cdot\dot{\boldsymbol{q}}$$