Robotics 1

Direct kinematics

Prof. Alessandro De Luca

Dipartimento di Ingegneria Informatica
Automatica e Gestionale Antonio Ruberti

Sapienza
Università di Roma

Kinematics of robot manipulators

- study of ...

  geometric and timing aspects of robot motion, without reference to the causes producing it

- robot seen as ...

  an (open) kinematic chain of rigid bodies interconnected by (revolute or prismatic) joints

Motivations

- functional aspects
  - definition of robot workspace
  - calibration
- operational aspects

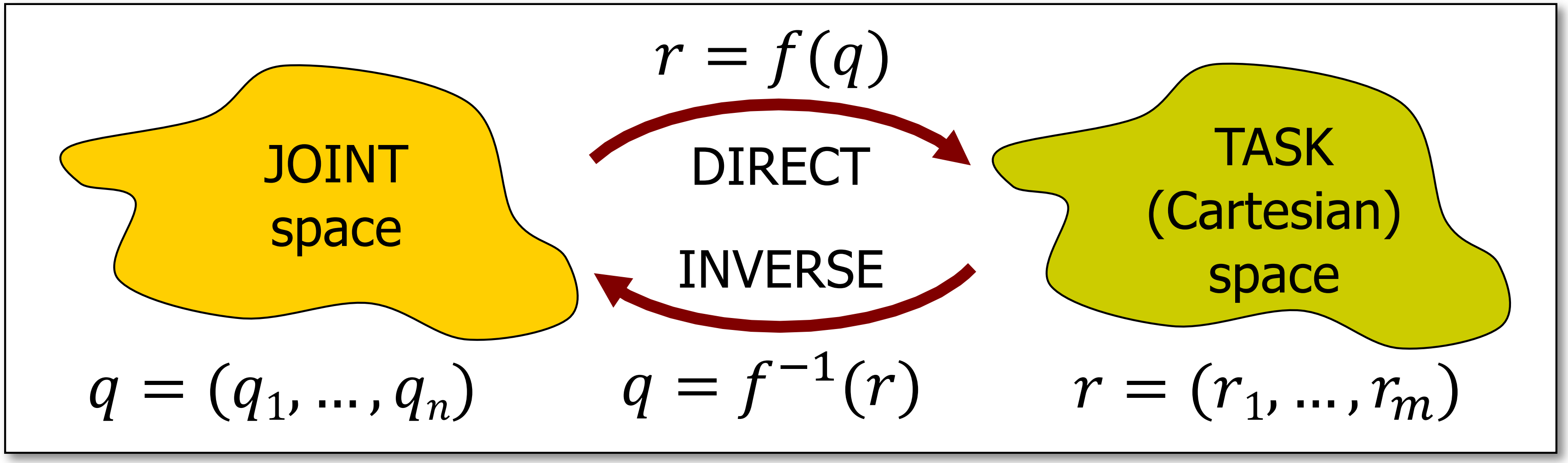
task execution (actuation by motors)

task definition and performance

two different "spaces" related by kinematic (and dynamic) maps

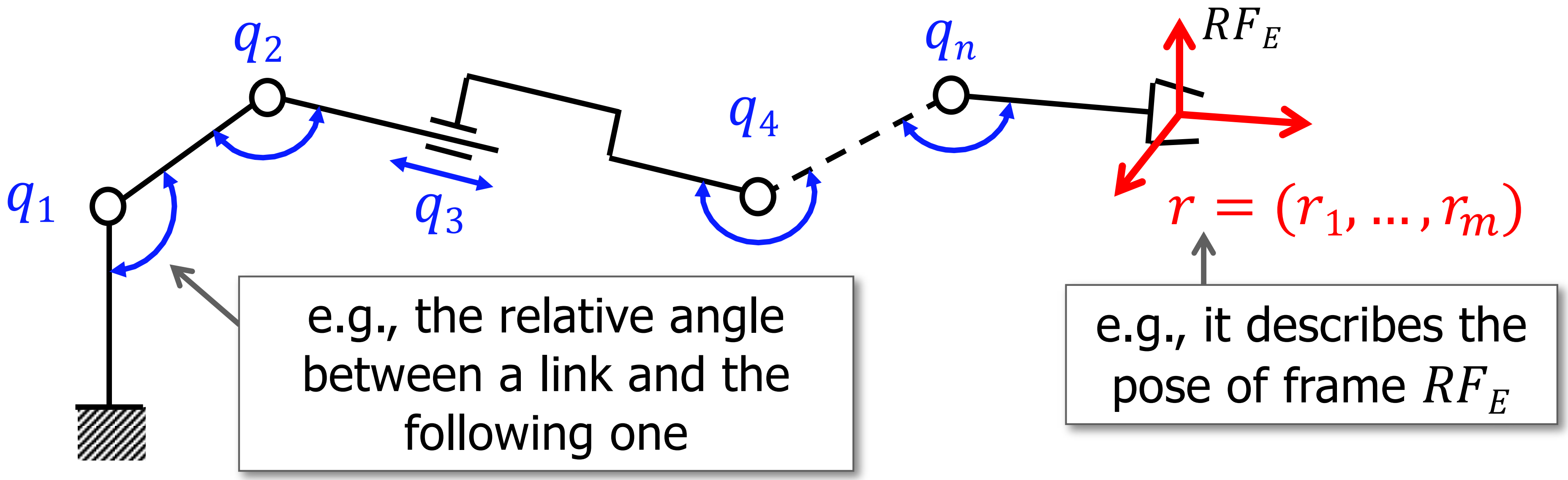
- trajectory planning
- programming
- motion control

Kinematics
formulation and parameterizations



- choice of parameterization q
  - unambiguous and minimal characterization of robot configuration
  - n = # degrees of freedom (dof) = # robot joints (rotational or translational)
- choice of parameterization r
  - compact description of position and/or orientation (pose) variables of interest to the required task
  - usually, $m \leq n$ and $m = 6$ (but none of these is strictly necessary)

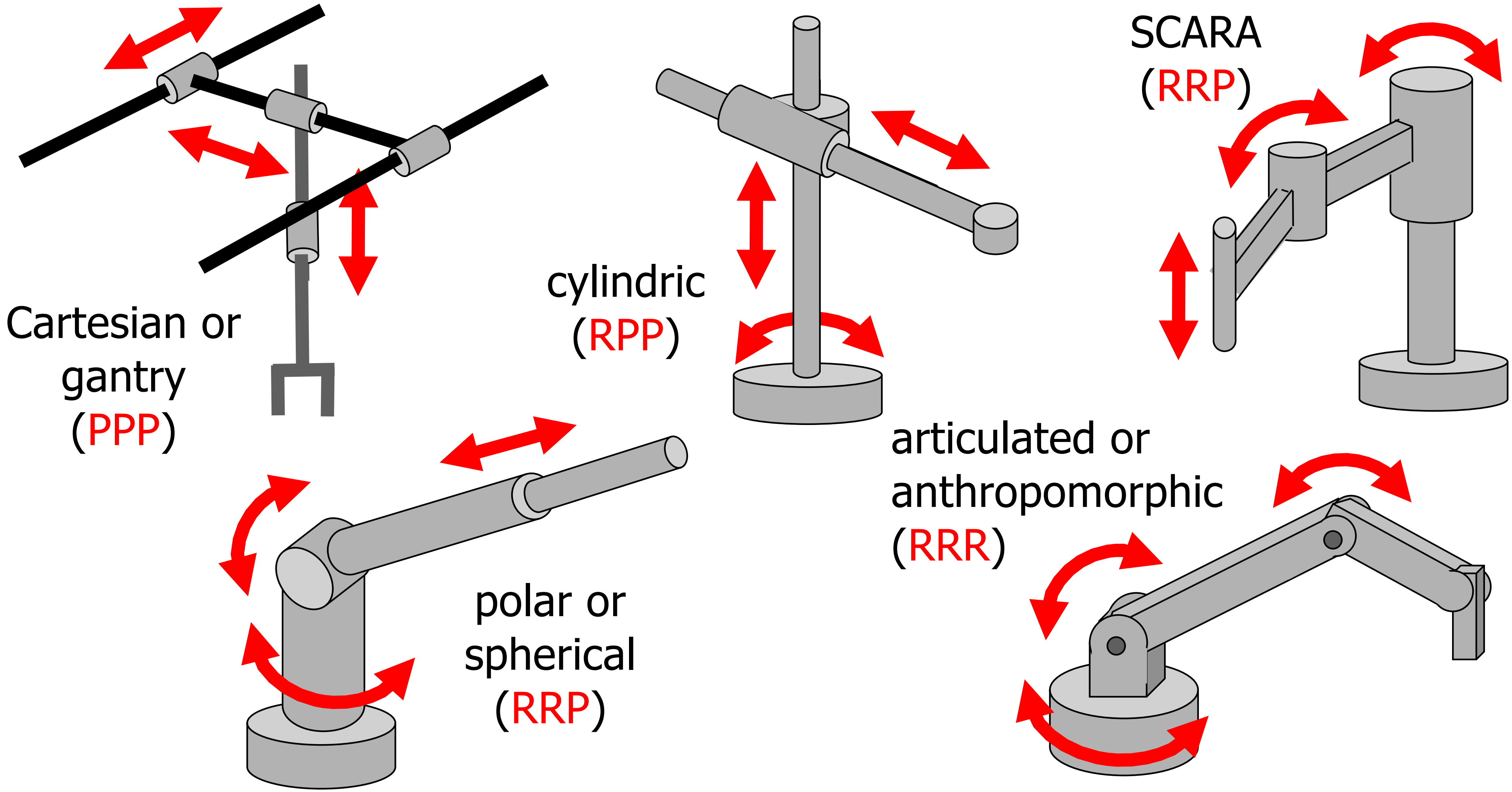
Open kinematic chains



- $m = 2$
  - pointing in space
  - positioning in the plane
- $m = 3$
  - orientation in space
  - positioning and orientation in the plane
- $m = 5$
  - positioning and pointing in space (like for spot welding)
- $m = 6$
  - positioning and orientation in space
  - positioning of two points in space (e.g., end-effector and elbow)

Classification by kinematic type

first 3 dofs only



R = 1-dof rotational (revolute) joint

P = 1-dof translational (prismatic) joint

Direct kinematic map

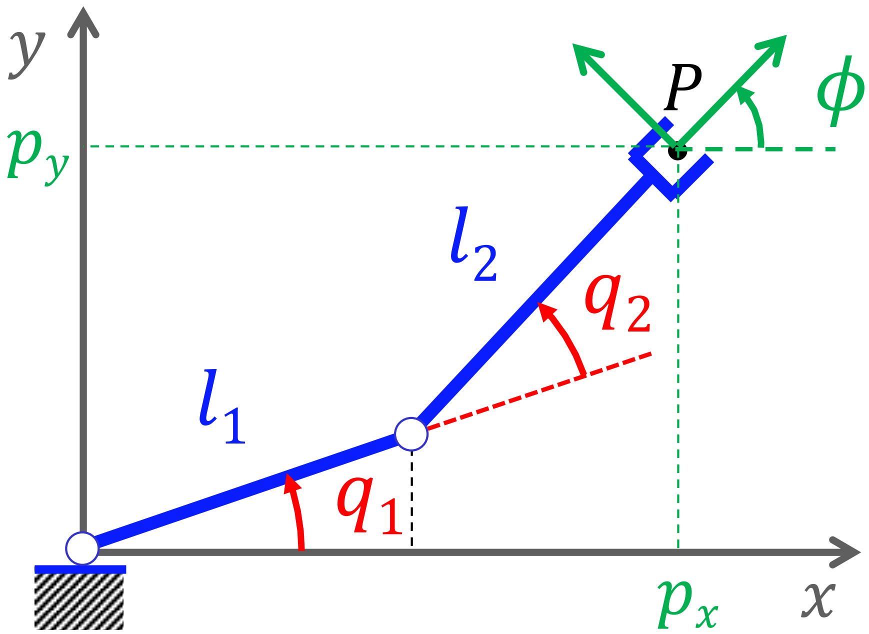
- the structure of the direct kinematics function depends on the chosen r

$$r = f_r(q)$$

- methods for computing $f_r(q)$
  - geometric/by inspection
  - systematic: assigning frames attached to the robot links and using homogeneous transformation matrices

Direct kinematics of 2R planar robot

just using inspection...



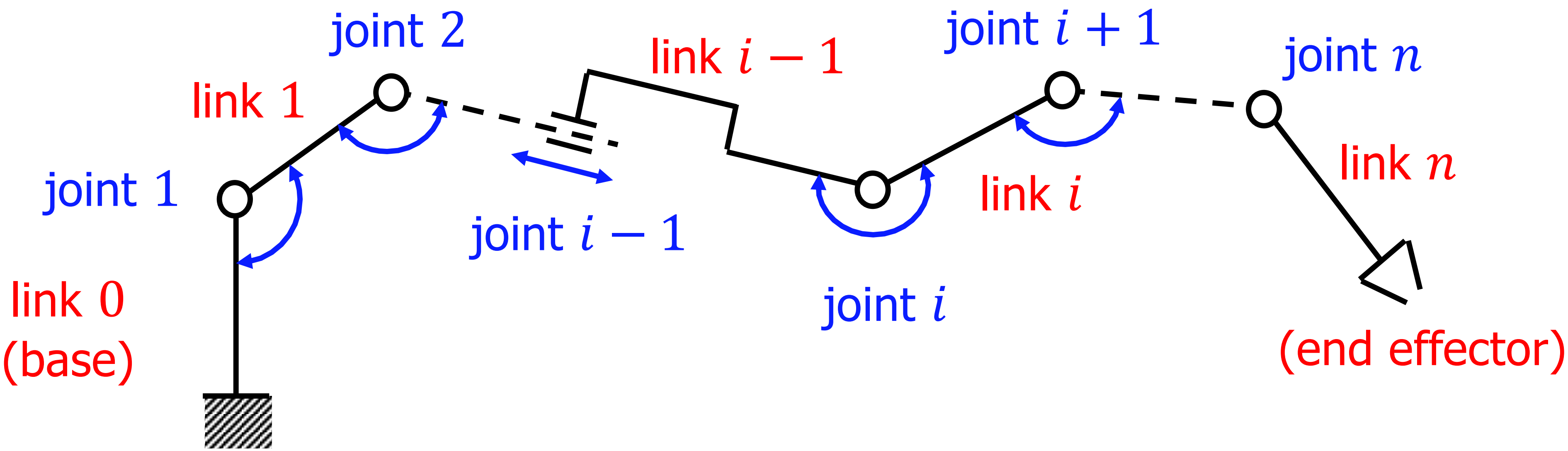
$$q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \qquad n = 2$$

$$r = \begin{bmatrix} p_x \\ p_y \\ \phi \end{bmatrix} \qquad m = 3$$

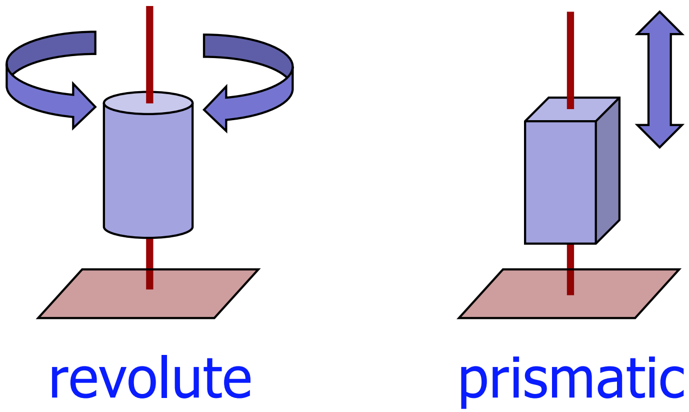
$$p_x = l_1 \cos q_1 + l_2 \cos(q_1 + q_2)$$
$$p_y = l_1 \sin q_1 + l_2 \sin(q_1 + q_2)$$
$$\phi = q_1 + q_2$$

for more general cases, we need a 'method'!

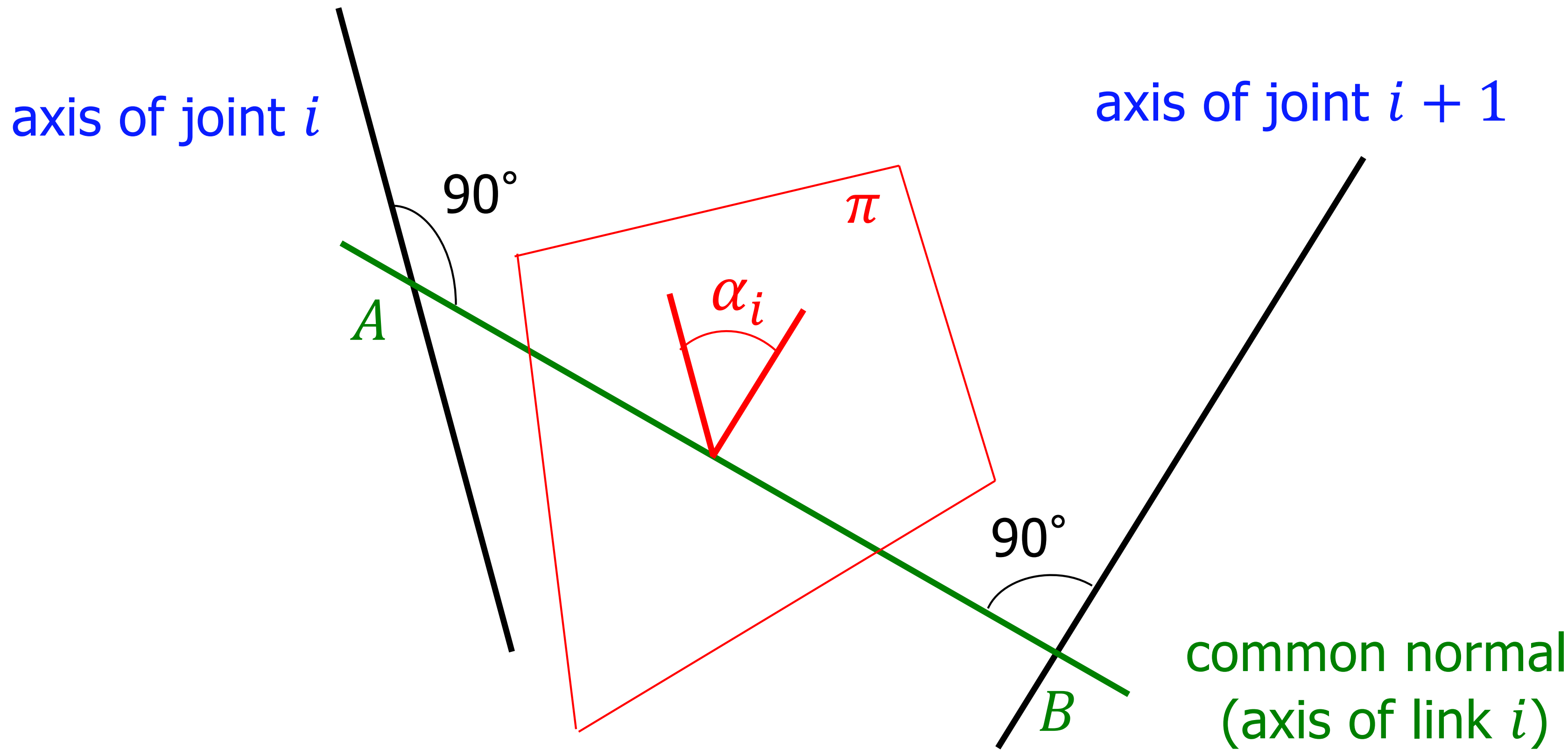
Numbering links and joints



icon representation of joint types
for the manipulator skeleton



Spatial relation between joint axes



a_i = **displacement** AB between joint axes (always well defined)

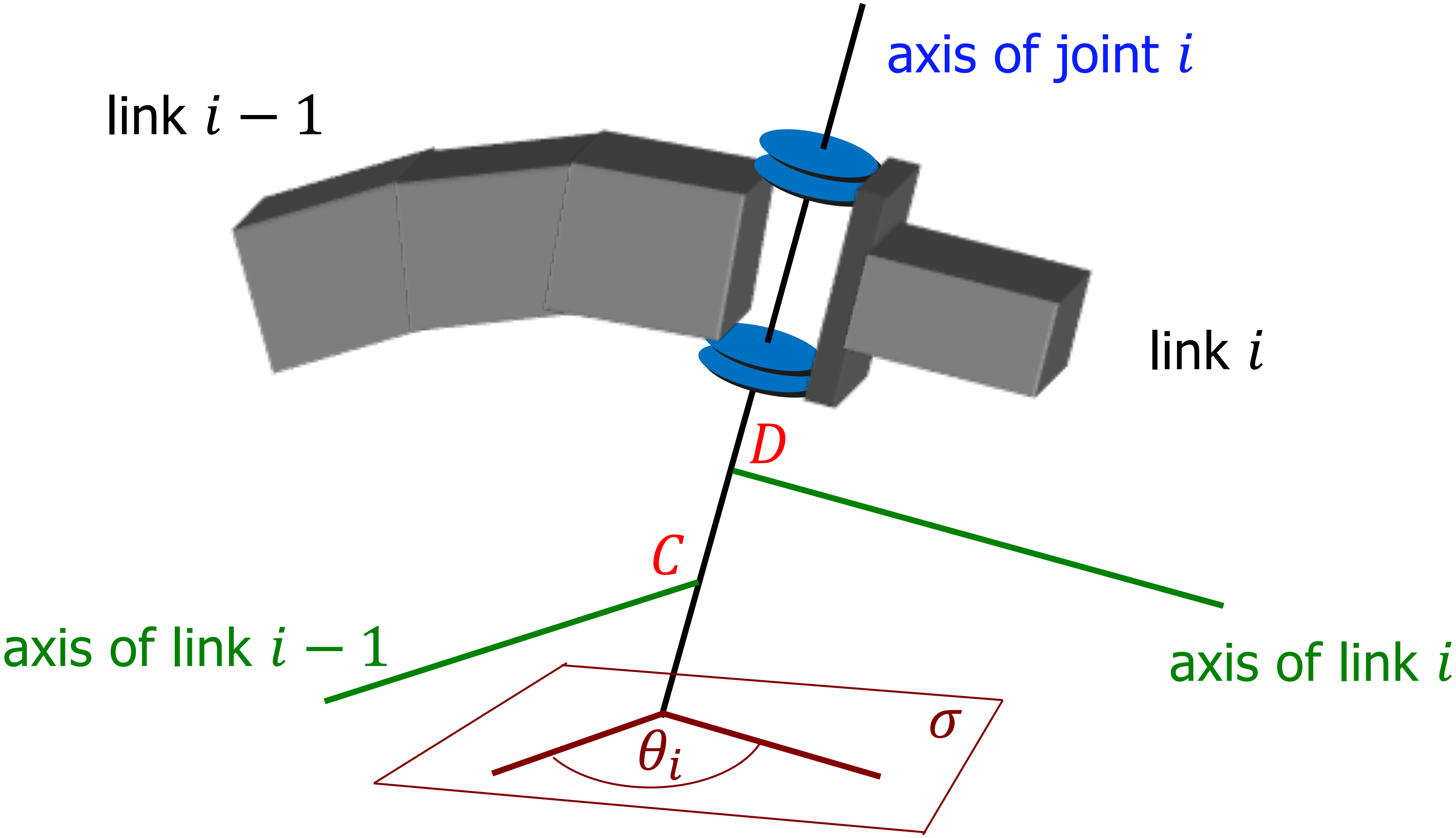
α_i = **twist angle** between joint axes

— projected on a plane π orthogonal to the link axis

} always constant!

} with sign (pos/neg)!

Spatial relation between link axes



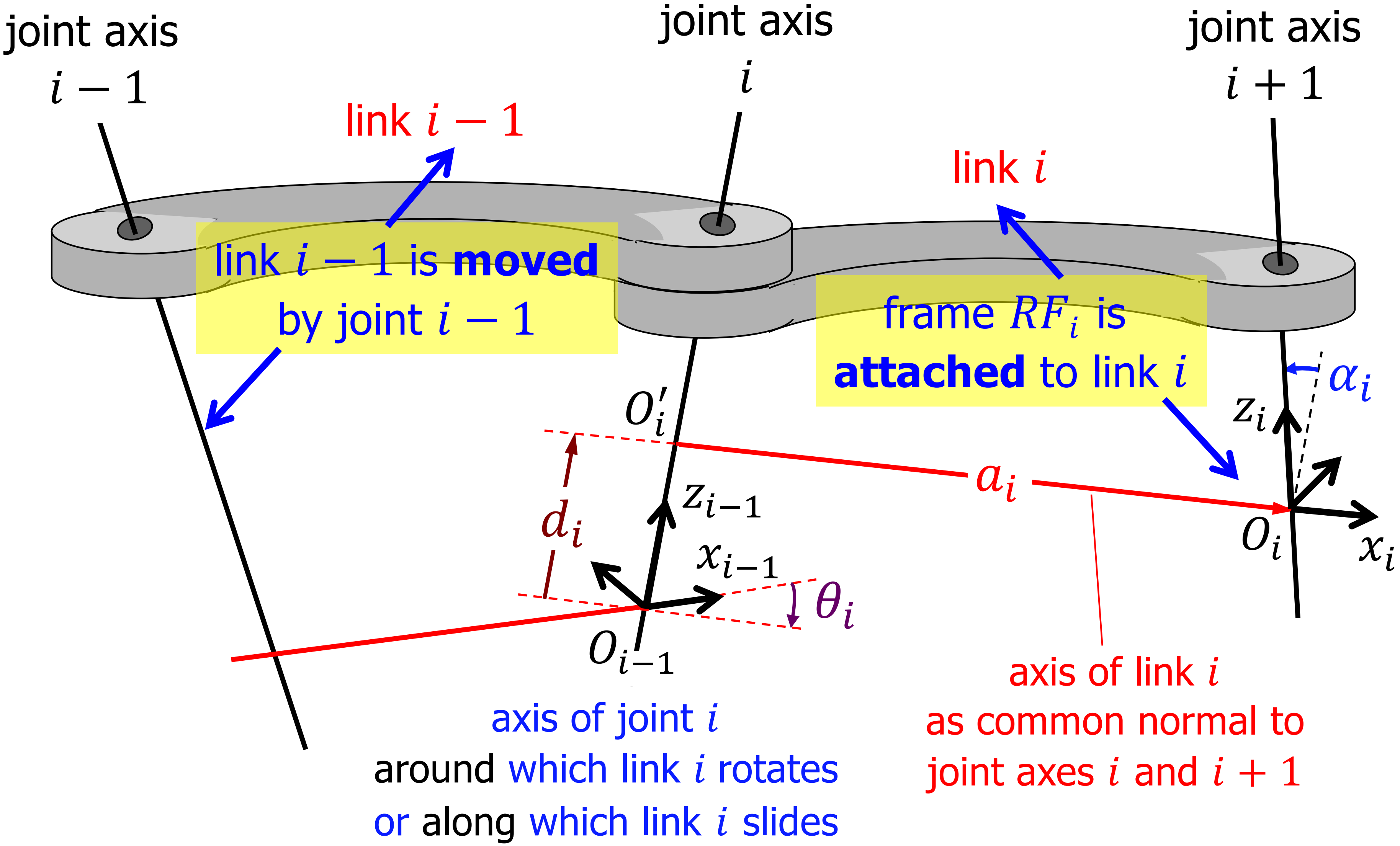
d_i = **displacement** CD (a variable if joint i is prismatic)

θ_i = **angle between link axes** (a variable if joint i is revolute)

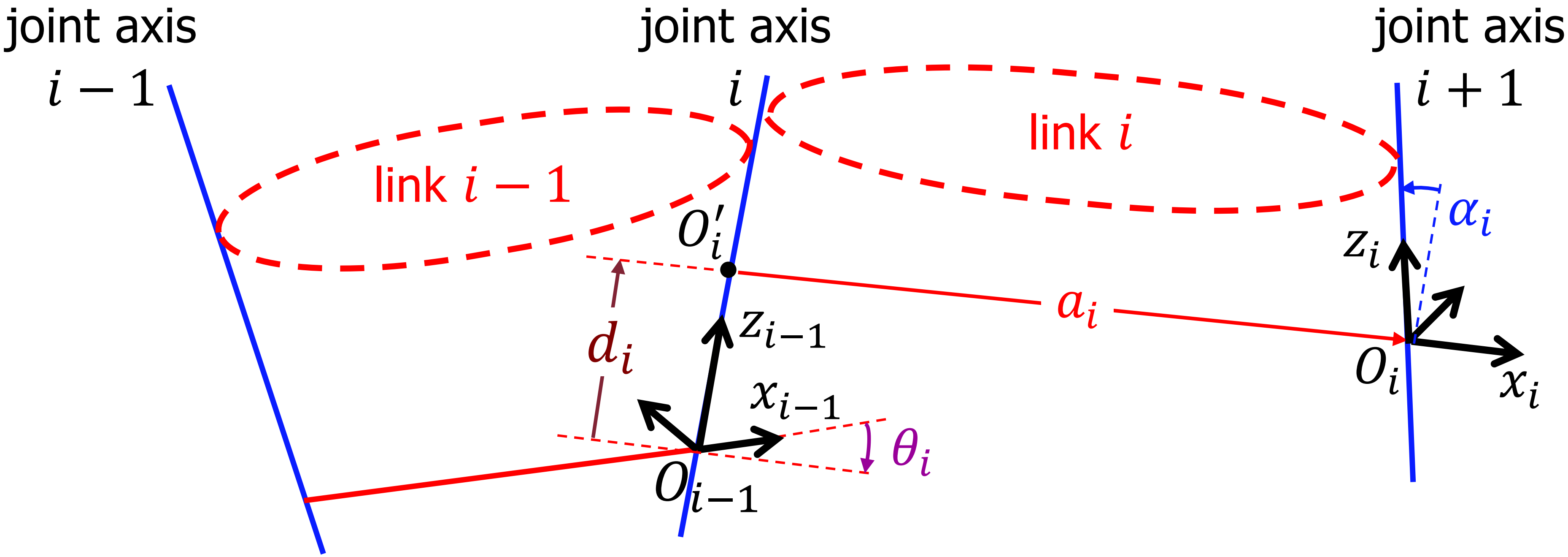
— projected on a plane σ orthogonal to the joint axis

} with sign (pos/neg)!

Denavit-Hartenberg (DH) frames



Definition of DH parameters



- unit vector z_i along axis of joint $i+1$
- unit vector x_i along the common normal to joint i and $i+1$ axes ($i \to i+1$)
- d_i = distance $O_{i-1}O_i'$, + if oriented as z_{i-1}, variable if joint i is PRISMATIC
- θ_i = angle from x_{i-1} to x_i around z_{i-1}, + if CCW, variable if joint i is REVOLUTE
- a_i = distance $O_i'O_i$, + if oriented as x_i, always constant (= 'length' of link i)
- α_i = twist angle from z_{i-1} to z_i around x_i, + if CCW, always constant

DH layout made simple

a popular 3-minute illustration...

video

https://www.youtube.com/watch?v=rA9tm0gTln8

- **note**: the author of this video uses r in place of a, and does not add subscripts!

Homogeneous transformation

between successive DH frames (from frame $i-1$ to frame i)

- roto-translation (screw motion) around and along z_{i-1}

$$
{}^{i-1}A_{i'}(q_i) = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

the product of these two matrices commutes!

rotational joint $\Rightarrow$ $q_i = \theta_i$

prismatic joint $\Rightarrow$ $q_i = d_i$

- roto-translation (screw motion) around and along x_i

$$
{}^{i'}A_i = \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & \cos\alpha_i & -\sin\alpha_i & 0 \\ 0 & \sin\alpha_i & \cos\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

← always a constant matrix

Denavit-Hartenberg matrix

J. Denavit and R.S. Hartenberg, "A kinematic notation for lower-pair mechanisms based on matrices," *Trans. ASME J. Applied Mechanics*, **23**: 215–221, 1955

$$
{}^{i-1}A_i(q_i) = {}^{i-1}A_{i'}(q_i)\ {}^{i'}A_i = \left[\begin{array}{ccc|c} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \cos\theta_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ \hline 0 & 0 & 0 & 1 \end{array}\right]
$$

compact notation: $c = \cos,\ s = \sin$

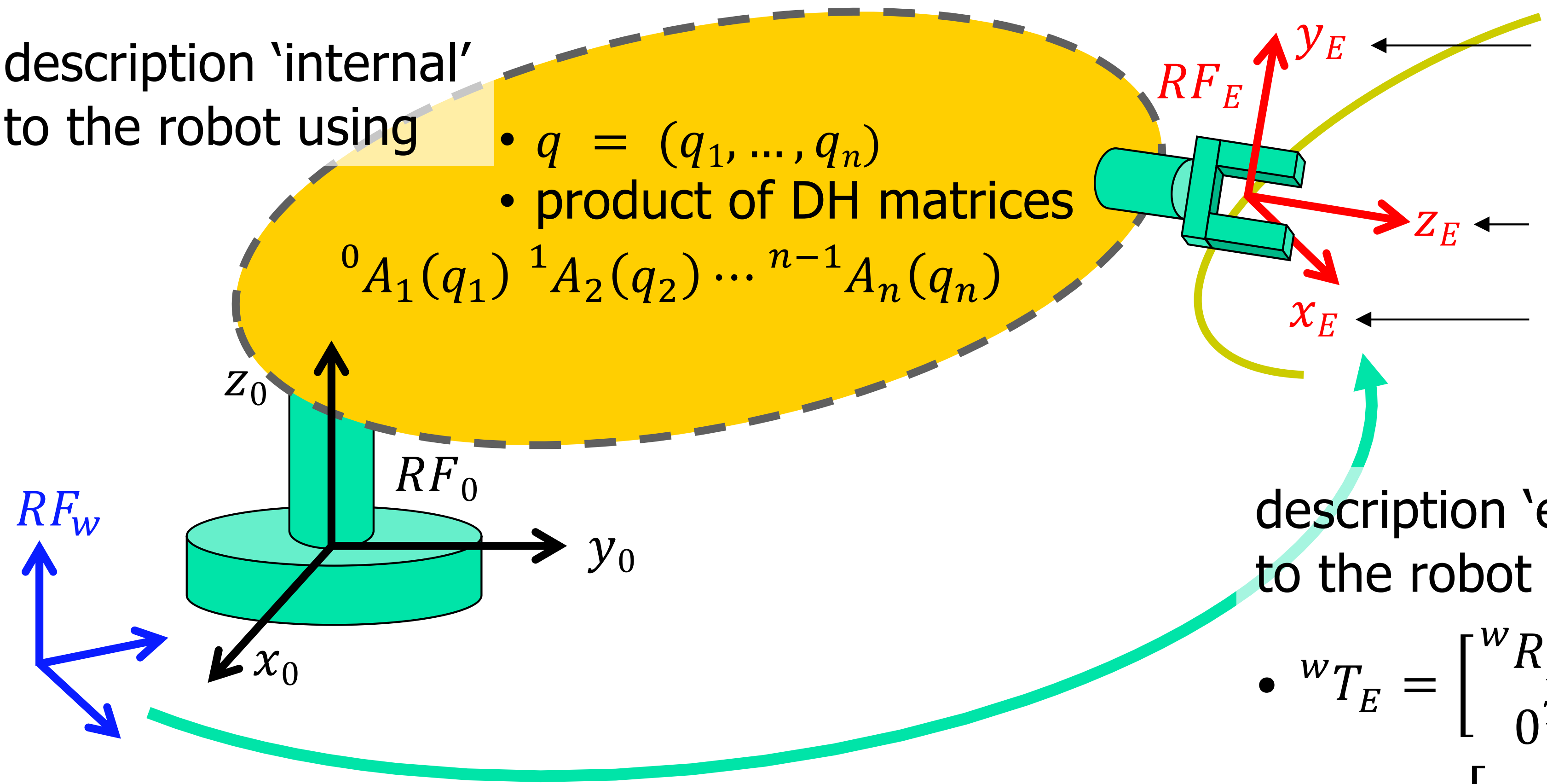
super-compact notation (if feasible): $c_i = \cos q_i,\ s_i = \sin q_i$

Direct kinematics of robot manipulators

description 'internal' to the robot using

- $q = (q_1, \ldots, q_n)$
- product of DH matrices

$$^{0}A_1(q_1)\ ^{1}A_2(q_2) \cdots\ ^{n-1}A_n(q_n)$$



description 'external' to the robot using

- $^{w}T_E = \begin{bmatrix} ^{w}R_E & ^{w}p_{wE} \\ 0^T & 1 \end{bmatrix} = \begin{bmatrix} n & s & a & p \\ & 0^T & & 1 \end{bmatrix}$
- $r = (r_1, \ldots, r_m)$

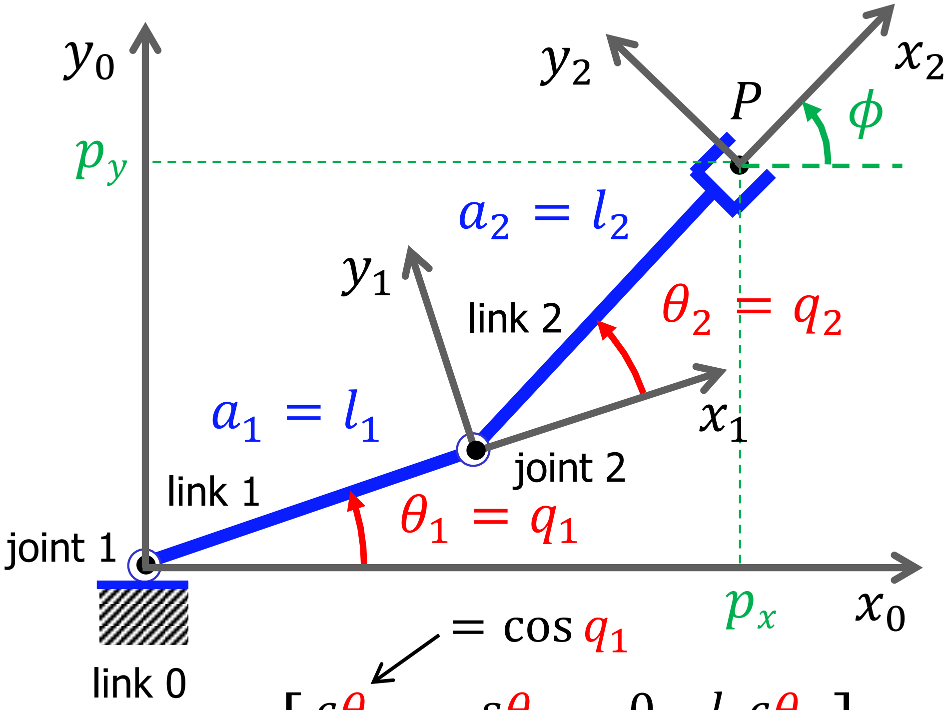
$$^{w}T_E = {}^{w}T_0\ ^{0}A_1(q_1)\ ^{1}A_2(q_2) \cdots\ ^{n-1}A_n(q_n)\ ^{n}T_E$$

$$r = f_r(q)$$

alternative representations of the **direct kinematics**

Direct kinematics of 2R planar robot

using DH frame assignment...



z_0, z_1, z_2 outgoing from plane

i	α_i	a_i	d_i	θ_i
1	0	l_1	0	q_1
2	0	l_2	0	q_2

$= \cos q_1$

$$^0A_1(\theta_1) = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & l_1 c\theta_1 \\ s\theta_1 & c\theta_1 & 0 & l_1 s\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^1A_2(\theta_2) = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & l_2 c\theta_2 \\ s\theta_2 & c\theta_2 & 0 & l_2 s\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$= \cos(q_1 + q_2)$

$$^0A_2(q) = \begin{bmatrix} c_{12} & -s_{12} & 0 & l_1 c_1 + l_2 c_{12} \\ s_{12} & c_{12} & 0 & l_1 s_1 + l_2 s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

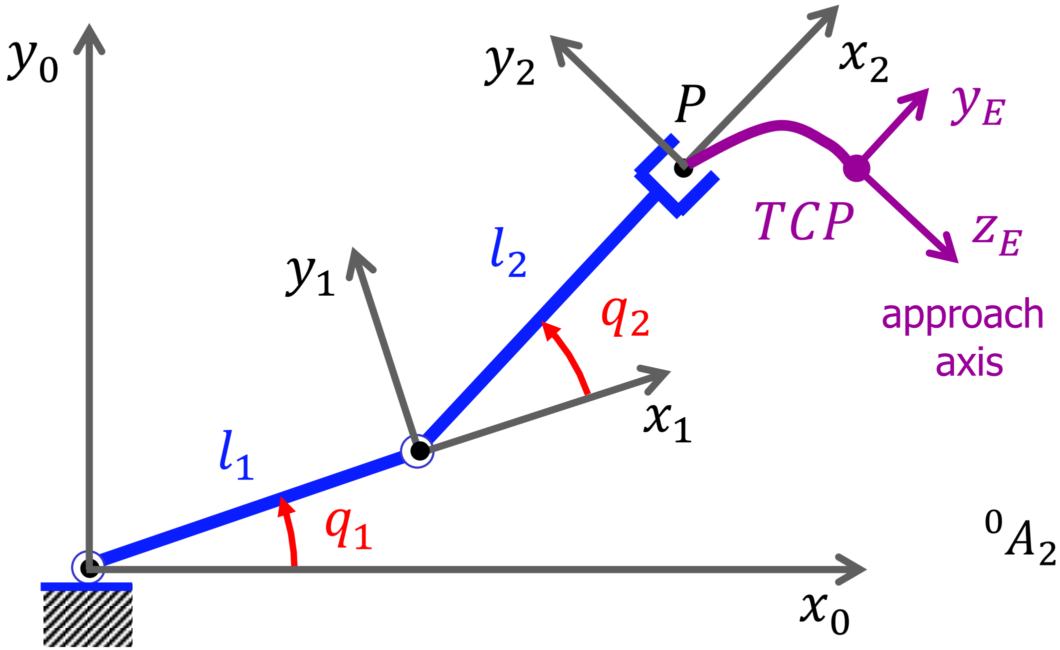
$^0\boldsymbol{p}_{\text{hom}}$ ↓ $\quad$ $^2\boldsymbol{p}_{\text{hom}}$ ↗

$$\begin{bmatrix} p \\ 1 \end{bmatrix} = \begin{bmatrix} p_x \\ p_y \\ 0 \\ 1 \end{bmatrix} = {}^0A_2(q) \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} l_1 c_1 + l_2 c_{12} \\ l_1 s_1 + l_2 s_{12} \\ 0 \\ 1 \end{bmatrix}$$

$\phi = q_1 + q_2$ (extracted from $^0R_2(q)$)

Direct kinematics of 2R planar robot

TCP location on the robot end effector



i	α_i	a_i	d_i	θ_i
1	0	l_1	0	q_1
2	0	l_2	0	q_2

$$^0A_2(q) = \begin{bmatrix} c_{12} & -s_{12} & 0 & l_1c_1 + l_2c_{12} \\ s_{12} & c_{12} & 0 & l_1s_1 + l_2s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Tool Center Point TCP and associated end-effector frame RF_E

$$^2T_E = \begin{bmatrix} 0 & 1 & 0 & ^2TCP_x \\ 0 & 0 & -1 & ^2TCP_y \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \begin{bmatrix} ^0TCP(q) \\ 1 \end{bmatrix} = \begin{bmatrix} ^0TCP_x(q) \\ ^0TCP_y(q) \\ 0 \\ 1 \end{bmatrix} = {}^0A_2(q) \begin{bmatrix} ^2TCP_x \\ ^2TCP_y \\ 0 \\ 1 \end{bmatrix} = {}^0T_E(q) \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$= {}^0A_2(q)\ {}^2T_E$$

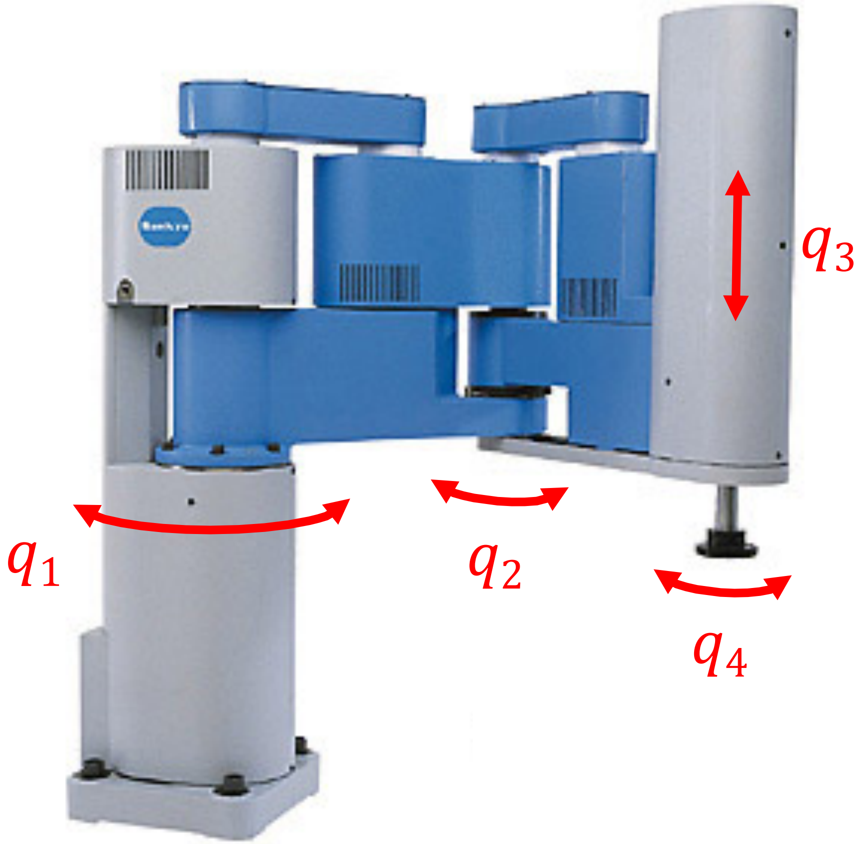
Ambiguities in defining DH frames

- frame 0: origin and x_0 axis are arbitrary (z_0 on first joint axis!)
- frame n: choose conveniently the origin, z_n axis is not specified
  - however, x_n must intersect and be chosen orthogonal to z_{n-1}
- positive direction of z_{i-1} (up/down on axis of joint i) is arbitrary
  - choose one, and try to 'avoid flipping over' to the next one
- positive direction of x_i (back/forth on axis of link i) is arbitrary
  - if successive joint axes are incident, we often take $x_i = z_{i-1} \times z_i$
  - when natural, follow the direction 'from base to tip'
- if z_{i-1} and z_i are parallel (common normal not uniquely defined)
  - O_i is chosen arbitrarily along z_i, still trying to 'zero out' parameters
- if z_{i-1} and z_i are coincident, normal x_i axis can be chosen at will
  - this case occurs only if the two joints are of different kind (P/R or R/P)
  - again, try using 'simple values' (e.g., 0 or $\pm\pi/2$) for constant angles

DH assignment for a SCARA robot

video

Sankyo SCARA 8438



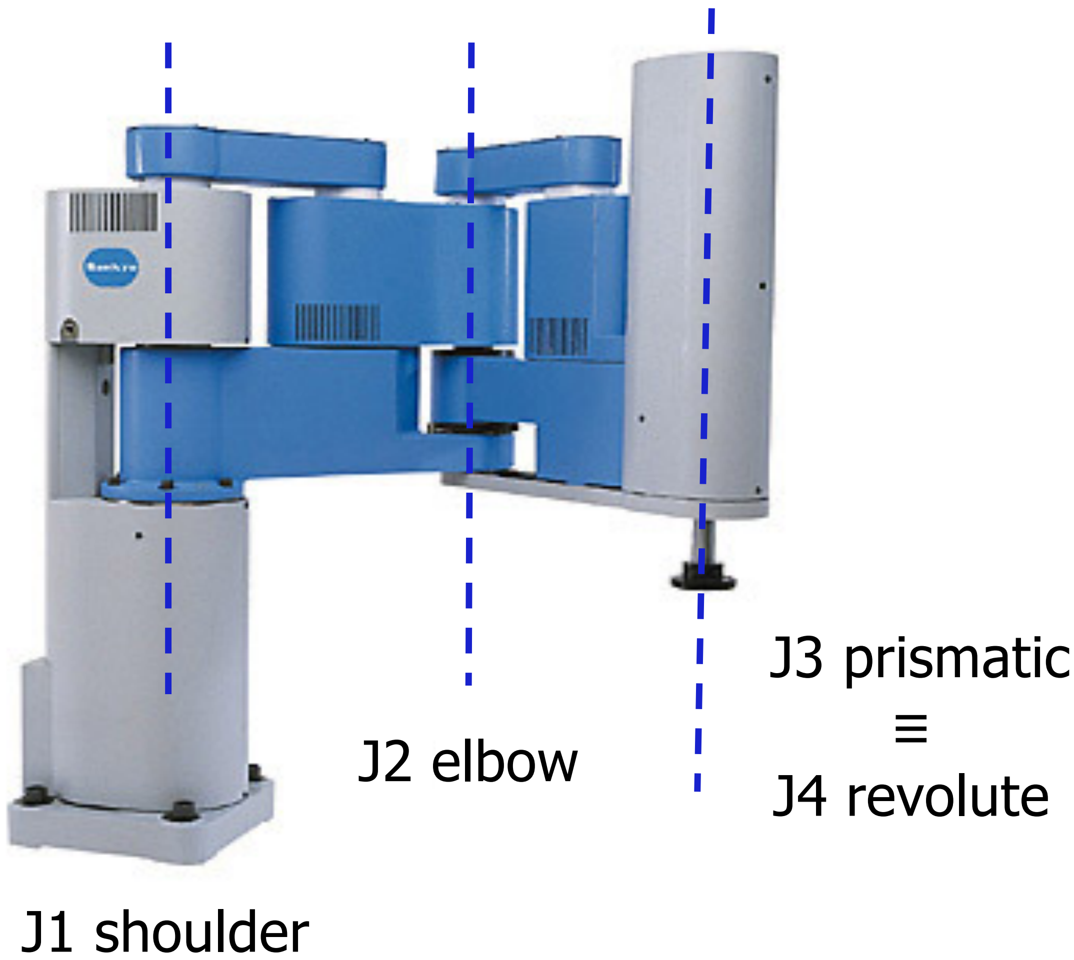
Sankyo SCARA SR 8447

Step 1: joint axes

all parallel
(or coincident)

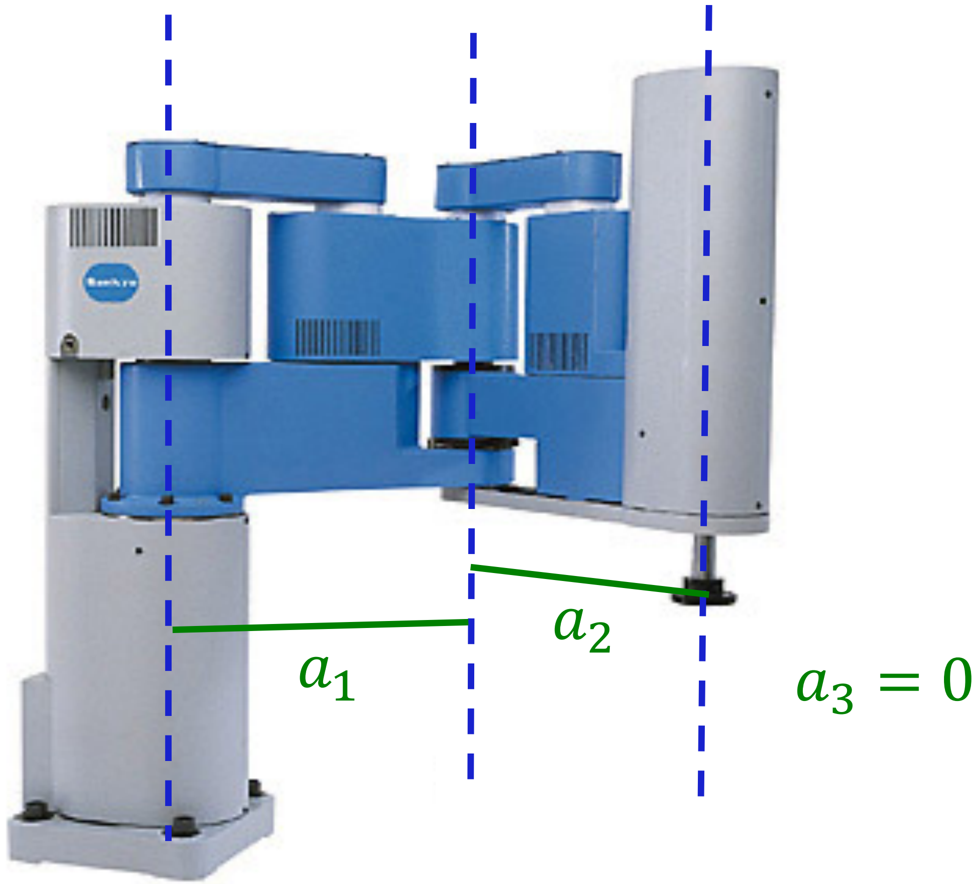
⬇

twist angles
$\alpha_i = 0$ or π

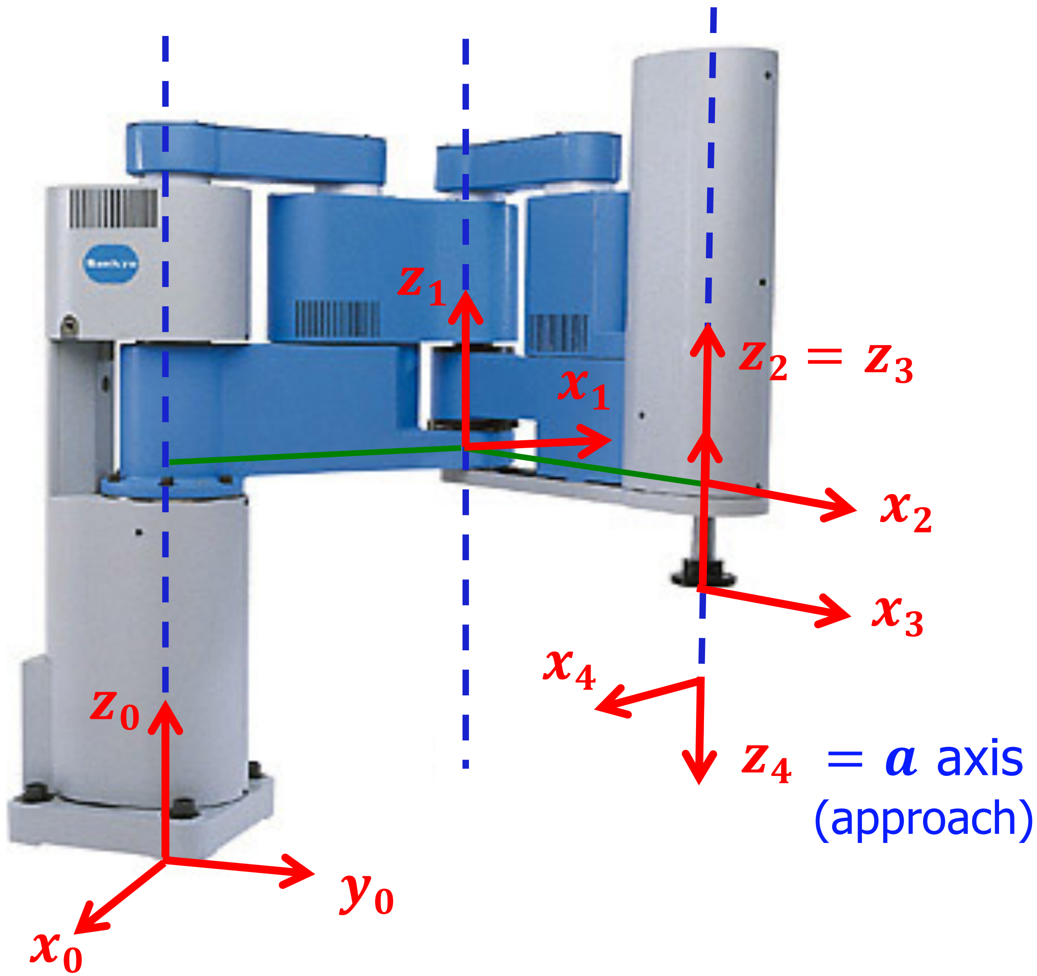


Step 2: link axes

the vertical 'heights'
of the link axes
are arbitrary
(for the time being)



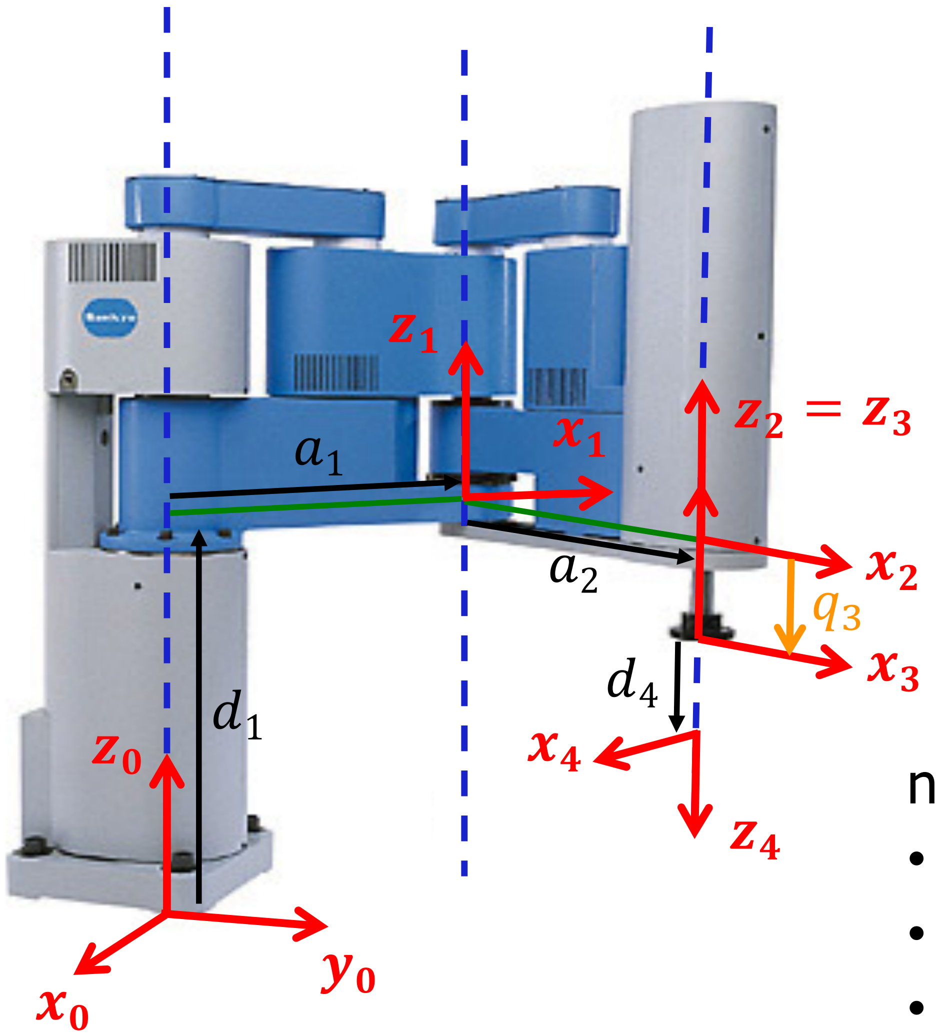
Step 3: frames



axes $\boldsymbol{y}_i$ for $i > 0$ are not shown

(nor needed; they form right-handed frames)

Step 4: DH table of parameters



i	α_i	a_i	d_i	θ_i
1	0	a_1	d_1	q_1
2	0	a_2	0	q_2
3	0	0	q_3	0
4	π	0	d_4	q_4

note that

- d_1 and d_4 could be set $= 0$
- $d_4 < 0$ (opposite to $\boldsymbol{z}_3$)
- $q_3 < 0$ in this configuration
- similarly, here $q_1 > 0, q_2 < 0, q_4 < 0$

Step 5: DH transformation matrices

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

$$^{0}A_1(q_1) = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & a_1 c\theta_1 \\ s\theta_1 & c\theta_1 & 0 & a_1 s\theta_1 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{1}A_2(q_2) = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & a_2 c\theta_2 \\ s\theta_2 & c\theta_2 & 0 & a_2 s\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{2}A_3(q_3) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$q = (q_1, q_2, q_3, q_4) = (\theta_1, \theta_2, d_3, \theta_4)$$

$$^{3}A_4(q_4) = \begin{bmatrix} c\theta_4 & s\theta_4 & 0 & 0 \\ s\theta_4 & -c\theta_4 & 0 & 0 \\ 0 & 0 & -1 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Step 6a: direct kinematics

homogeneous matrix ${}^{w}T_E$ as product of the ${}^{i-1}A_i(q_i)$'s

$$ {}^{0}A_2(q_1,q_2) = \begin{bmatrix} c_{12} & -s_{12} & 0 & a_1 c_1 + a_2 c_{12} \\ s_{12} & c_{12} & 0 & a_1 s_1 + a_2 s_{12} \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{0}A_3(q_1,q_2,q_3) = \begin{bmatrix} c_{12} & -s_{12} & 0 & a_1 c_1 + a_2 c_{12} \\ s_{12} & c_{12} & 0 & a_1 s_1 + a_2 s_{12} \\ 0 & 0 & 1 & d_1 + q_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$R(q_1,q_2,q_4) = [\, n \;\; s \;\; a \,]$ $\quad p = p(q_1,q_2,q_3)$

$$ {}^{w}T_E = {}^{0}A_4(q_1,q_2,q_3,q_4) = \begin{bmatrix} c_{124} & s_{124} & 0 & a_1 c_1 + a_2 c_{12} \\ s_{124} & -c_{124} & 0 & a_1 s_1 + a_2 s_{12} \\ 0 & 0 & -1 & d_1 + q_3 + d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$({}^{w}T_0 = {}^{4}T_E = I)$

Step 6b: direct kinematics

as task vector $r \in \mathbb{R}^m$

$$
{}^0A_4(q_1,q_2,q_3,q_4) = \begin{bmatrix} c_{124} & s_{124} & 0 & a_1c_1 + a_2c_{12} \\ s_{124} & -c_{124} & 0 & a_1s_1 + a_2s_{12} \\ 0 & 0 & -1 & d_1 + q_3 + d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

extract $\alpha_z \in \mathbb{R}$ from $R(q_1,q_2,q_4)$

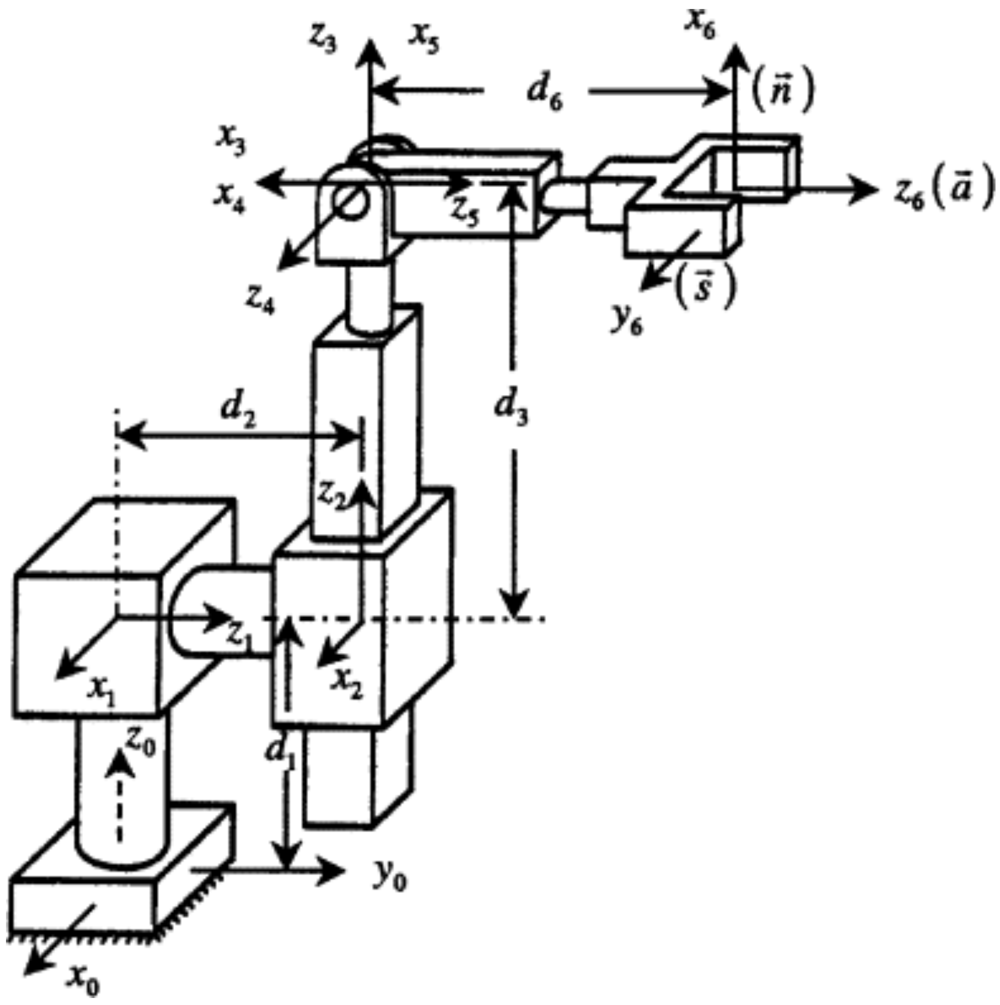
take $p \in \mathbb{R}^3$ as such from $p(q_1,q_2,q_3)$

$$
r = \begin{bmatrix} p_x \\ p_y \\ p_z \\ \alpha_z \end{bmatrix} = f_r(q) = \begin{bmatrix} a_1c_1 + a_2c_{12} \\ a_1s_1 + a_2s_{12} \\ d_1 + q_3 + d_4 \\ q_1 + q_2 + q_4 \end{bmatrix} \in \mathbb{R}^4
$$

MATLAB code available on web site: **dirkin_SCARA.m**

Stanford manipulator

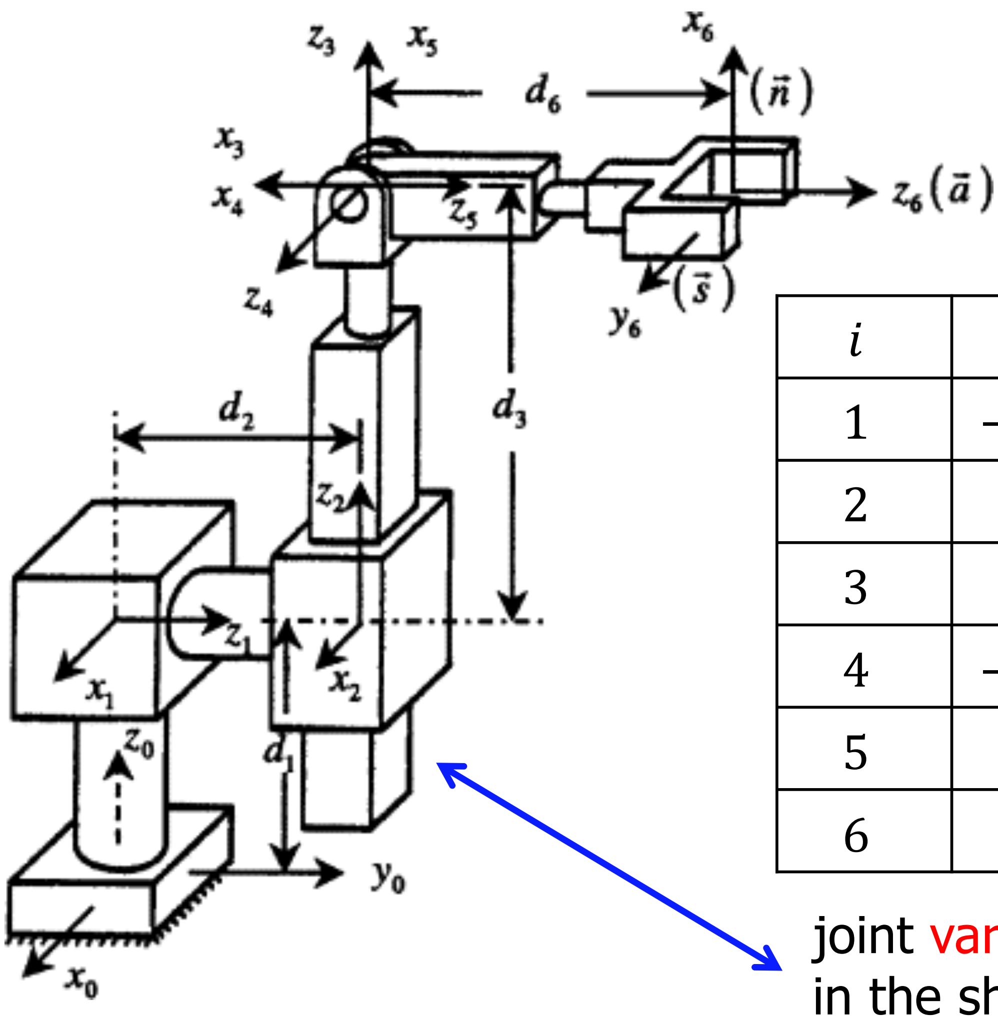
- 6-dof: 2R-1P-3R (spherical wrist)



- robot with shoulder offset
- 'one possible' DH assignment of frames is shown
- determine the associated
  - table of DH parameters
  - homogeneous transformation matrices
  - direct kinematics
- write a program for computing the direct kinematics
  - numerically (Matlab), given a $\boldsymbol{q}$
  - symbolically (Mathematica, Maple, Symbolic Manipulation Toolbox of Matlab, …)

DH table for Stanford manipulator

- 6-dof: 2R-1P-3R (spherical wrist)

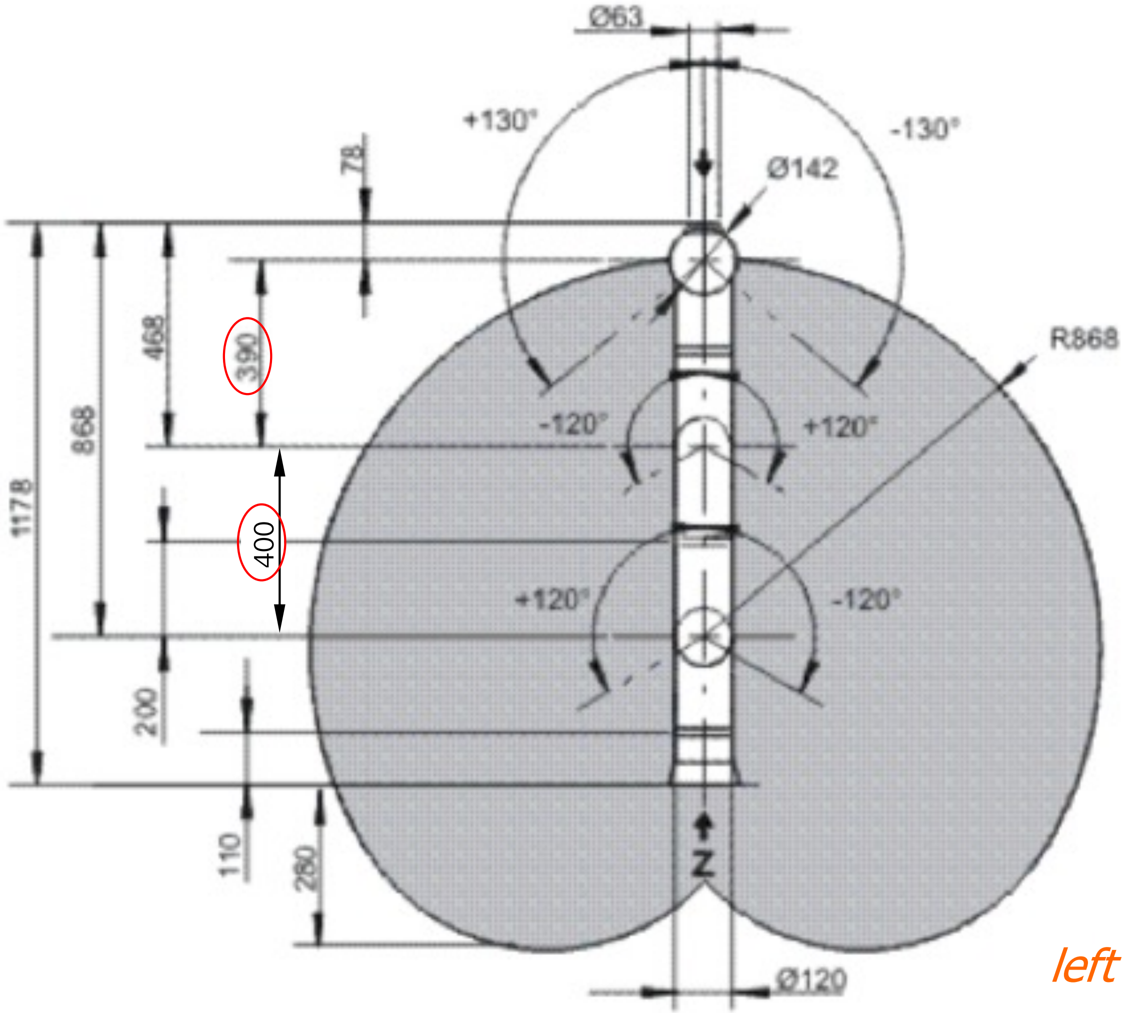


i	α_i	a_i	d_i	θ_i
1	$-\pi/2$	0	$d_1 > 0$	$q_1 = 0$
2	$\pi/2$	0	$d_2 > 0$	$q_2 = 0$
3	0	0	$q_3 > 0$	$-\pi/2$
4	$-\pi/2$	0	0	$q_4 = 0$
5	$\pi/2$	0	0	$q_5 = -\pi/2$
6	0	0	$d_6 > 0$	$q_6 = 0$

joint variables are in red, while their values in the shown robot configuration are in blue

KUKA LWR 4+

- 7R (no offsets, spherical shoulder and spherical wrist)



side view (from the left)

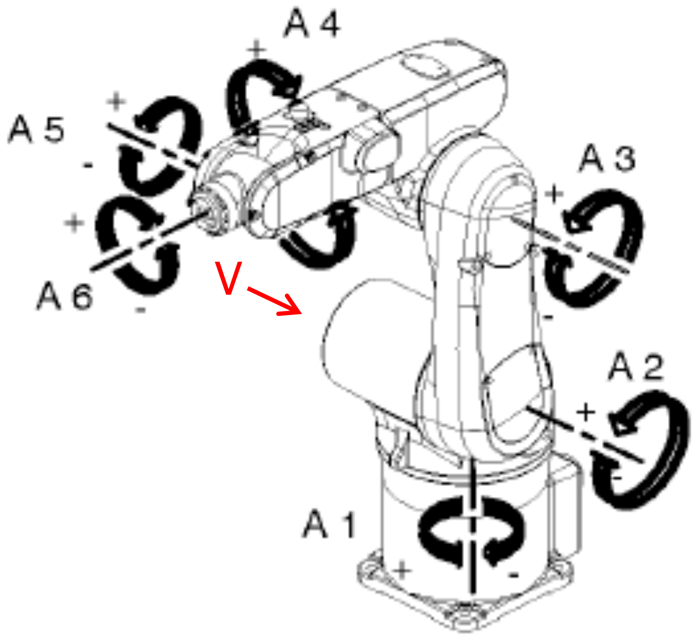
frontal view

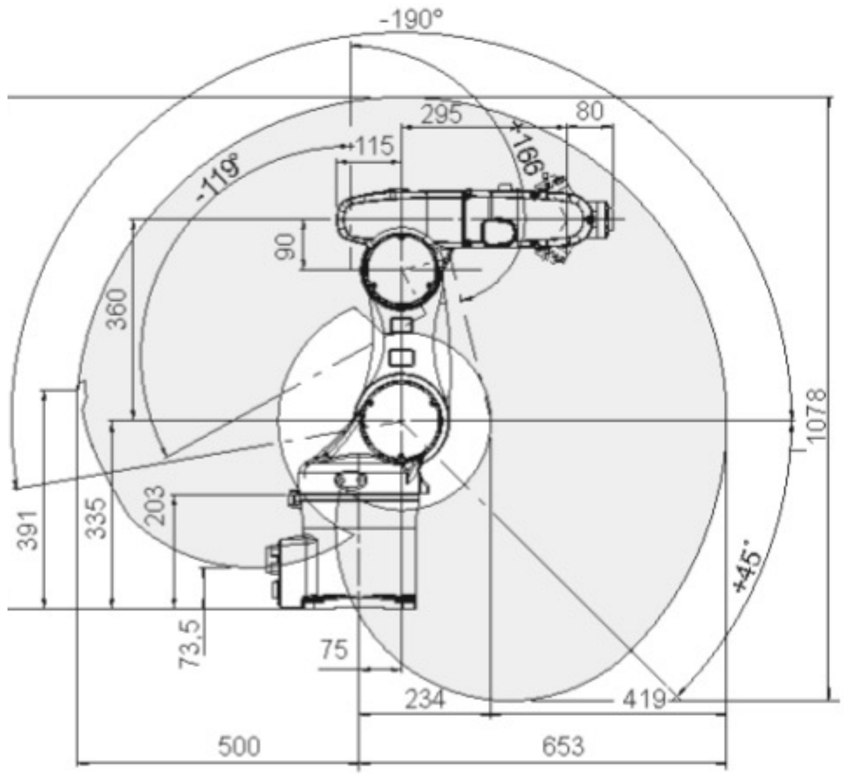
available at
DIAG Robotics Lab

- determine
  - frames and table of DH parameters
  - homogeneous transformation matrices
  - direct kinematics
  - d_1 and d_7 can be set $= 0$ or not (as needed)

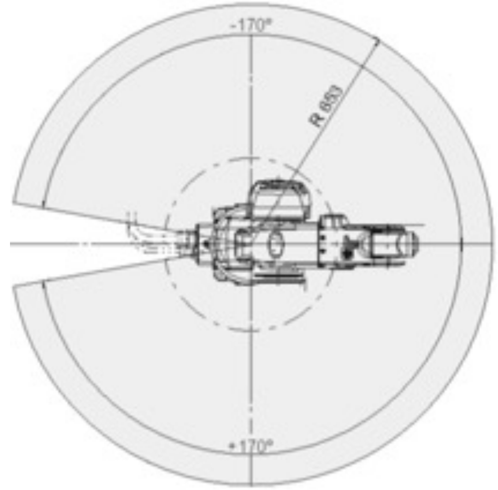
KUKA KR5 Sixx R650

- 6R (offsets at shoulder and elbow, spherical wrist)





side view (from observer in V)



top view

- determine
  - frames and table of DH parameters
  - homogeneous transformation matrices
  - direct kinematics (symbolic & numeric)

available at
DIAG Robotics Lab

Appendix: Modified DH convention

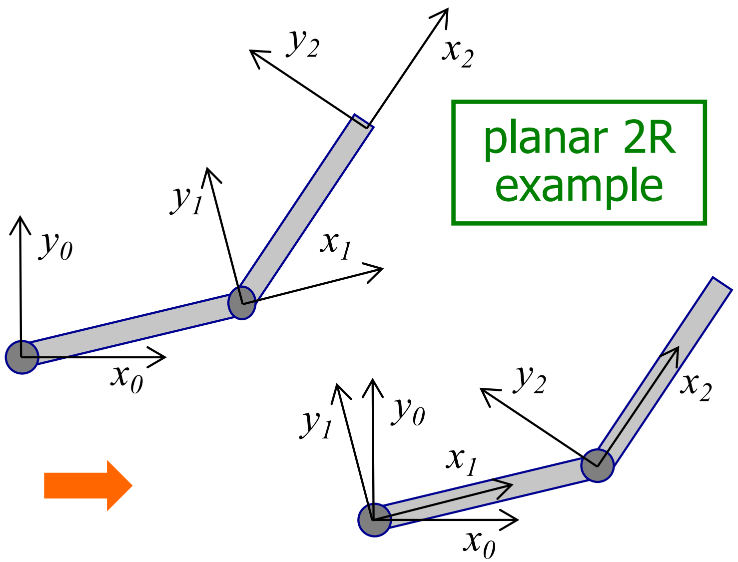
- a modified version introduced in J. Craig's book "Introduction to Robotics" (1986) and aligned for the indexing by Khalil and Kleinfinger (ICRA, 1986)
  - has z_i axis on joint i
  - a_i & α_i = distance & twist angle from z_{i-1} to z_i, measured along & about x_{i-1}
  - d_i & θ_i = distance & angle from x_{i-1} to x_i, measured along & about z_i
  - source of much confusion... if you are not aware of it (or don't mention it!)
  - convenient with link flexibility: a rigid frame at the base, another at the tip...

classical (or distal)

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

modified (or proximal)

$$
{}^{i-1}A_i^{\text{mod}} = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_i \\ c\alpha_i s\theta_i & c\alpha_i c\theta_i & -s\alpha_i & -d_i s\alpha_i \\ s\alpha_i s\theta_i & s\alpha_i c\theta_i & c\alpha_i & d_i c\alpha_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$



modified DH tends to place frames 'at the base' of each link