FOUNDATIONS OF ROBOTICS

Kinematics

B. Siciliano, L. Villani, G. Oriolo, A. De Luca

These slides are distributed as companion material to the textbook
Foundations of Robotics by B. Siciliano, L. Villani, G. Oriolo, A. De Luca
published by Springer Nature.
Modification is not allowed without permission

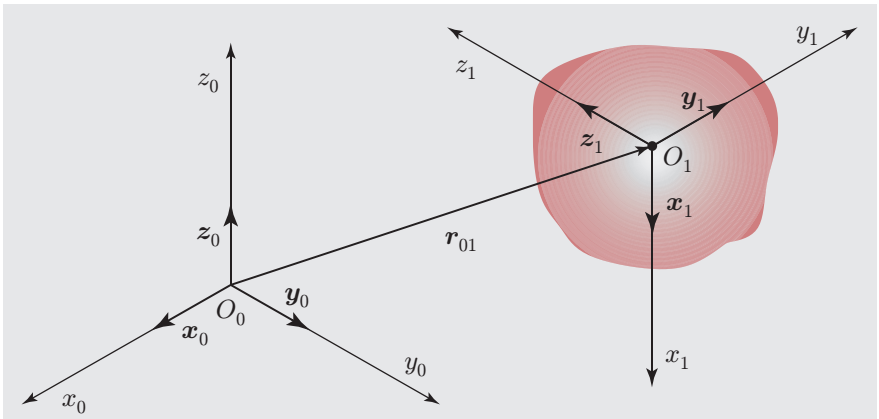
Outline

- Pose of a Rigid Body
- Rotation Matrix
- Other Representations of Orientation
- Homogeneous Transformations
- Configuration Space
- Task Space and Direct Kinematics
- Direct Kinematics of Manipulators
- Examples of Direct Kinematics
- Kinematic Calibration
- Reachable Workspace
- Inverse Kinematics

Pose of a Rigid Body

- A rigid body is described in space by
  - its position
  - its orientation
- Together, position and orientation define the pose of the rigid body
- The pose is described by attaching a reference frame to the body
  - all frames are orthogonal, right-handed O–xyz
  - unit vectors along the axes are denoted by $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$
- To distinguish multiple frames an index is added to frames and axes as a subscript

- Consider the reference frame O_0–$x_0y_0z_0$ with unit vectors $\boldsymbol{x}_0, \boldsymbol{y}_0, \boldsymbol{z}_0$
- Consider the frame O_1–$x_1y_1z_1$ attached to the rigid body with unit vectors $\boldsymbol{x}_1, \boldsymbol{y}_1, \boldsymbol{z}_1$



- The pose of the rigid body is described by
  - the position of O_1 with respect to O_0
  - the orientation of frame O_1–$x_1 y_1 z_1$ with respect to O_0–$x_0 y_0 z_0$
- The position of O_1 with respect to O_0 is the vector $\boldsymbol{r}_{01} \in I\!R^3$
- The vector is expressed in frame 0 by coordinates ${}^0r_x$, ${}^0r_y$, ${}^0r_z$ along x_0, y_0, z_0

$$ {}^0\boldsymbol{r}_{01} = {}^0r_x \boldsymbol{x}_0 + {}^0r_y \boldsymbol{y}_0 + {}^0r_z \boldsymbol{z}_0 = \begin{pmatrix} {}^0r_x \\ {}^0r_y \\ {}^0r_z \end{pmatrix} $$

where the left superscript indicates the frame in which the vector is represented

- Orientation is described by the unit vectors of frame 1 expressed in frame 0

$$
\begin{aligned}
{}^0\boldsymbol{x}_1 &= {}^0x_x\boldsymbol{x}_0 + {}^0x_y\boldsymbol{y}_0 + {}^0x_z\boldsymbol{z}_0 \\
{}^0\boldsymbol{y}_1 &= {}^0y_x\boldsymbol{x}_0 + {}^0y_y\boldsymbol{y}_0 + {}^0y_z\boldsymbol{z}_0 \\
{}^0\boldsymbol{z}_1 &= {}^0z_x\boldsymbol{x}_0 + {}^0z_y\boldsymbol{y}_0 + {}^0z_z\boldsymbol{z}_0
\end{aligned}
$$

- The coordinates are the direction cosines of the axes of frame O_1–$x_1y_1z_1$ with respect to frame O_0–$x_0y_0z_0$
- Each coordinate is a scalar product of two unit vectors, e.g. ${}^0y_x = \boldsymbol{x}_0^T\boldsymbol{y}_1$

- In the planar case each frame is represented by O–xy and unit vectors $\boldsymbol{x}$, $\boldsymbol{y}$
- The body moves in the $x_0 y_0$ plane
- Equations for position and orientation reduce to the 2D case

$$ {}^0\boldsymbol{r}_{01} = {}^0r_x \boldsymbol{x}_0 + {}^0r_y \boldsymbol{y}_0 = \begin{pmatrix} {}^0r_x \\ {}^0r_y \end{pmatrix} \quad \text{and} \quad \begin{aligned} {}^0\boldsymbol{x}_1 &= {}^0x_x \boldsymbol{x}_0 + {}^0x_y \boldsymbol{y}_0 \\ {}^0\boldsymbol{y}_1 &= {}^0y_x \boldsymbol{x}_0 + {}^0y_y \boldsymbol{y}_0 \end{aligned} $$

Rotation Matrix

- Orientation of a rigid body represented by a special 3×3 matrix $\boldsymbol{R}$
- This matrix collects the direction cosines of the body frame axes with respect to a reference frame
- Orientation of frame O_1–$x_1 y_1 z_1$ with respect to frame O_0–$x_0 y_0 z_0$ desribed by rotation matrix

$$
{}^0\boldsymbol{R}_1 = \begin{pmatrix} {}^0x_x & {}^0y_x & {}^0z_x \\ {}^0x_y & {}^0y_y & {}^0z_y \\ {}^0x_z & {}^0y_z & {}^0z_z \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}_0^T\boldsymbol{x}_1 & \boldsymbol{x}_0^T\boldsymbol{y}_1 & \boldsymbol{x}_0^T\boldsymbol{z}_1 \\ \boldsymbol{y}_0^T\boldsymbol{x}_1 & \boldsymbol{y}_0^T\boldsymbol{y}_1 & \boldsymbol{y}_0^T\boldsymbol{z}_1 \\ \boldsymbol{z}_0^T\boldsymbol{x}_1 & \boldsymbol{z}_0^T\boldsymbol{y}_1 & \boldsymbol{z}_0^T\boldsymbol{z}_1 \end{pmatrix} = \begin{pmatrix} {}^0\boldsymbol{x}_1 & {}^0\boldsymbol{y}_1 & {}^0\boldsymbol{z}_1 \end{pmatrix}
$$

- Columns of ${}^0\boldsymbol{R}_1$ are the unit vectors of the rotated frame expressed in the reference frame

- Column vectors $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{z}$ of any rotation matrix $\boldsymbol{R}$ are mutually orthogonal

$$\boldsymbol{x}^T\boldsymbol{y} = 0 \qquad \boldsymbol{x}^T\boldsymbol{z} = 0 \qquad \boldsymbol{y}^T\boldsymbol{z} = 0$$

- Each column has unit norm

$$\boldsymbol{x}^T\boldsymbol{x} = 1 \qquad \boldsymbol{y}^T\boldsymbol{y} = 1 \qquad \boldsymbol{z}^T\boldsymbol{z} = 1$$

- As a consequence $\boldsymbol{R}$ is orthonormal and satisfies

$$\boldsymbol{R}^T\boldsymbol{R} = \boldsymbol{I}$$

$$\boldsymbol{R}^{-1} = \boldsymbol{R}^T$$

$$\det \boldsymbol{R} = 1$$

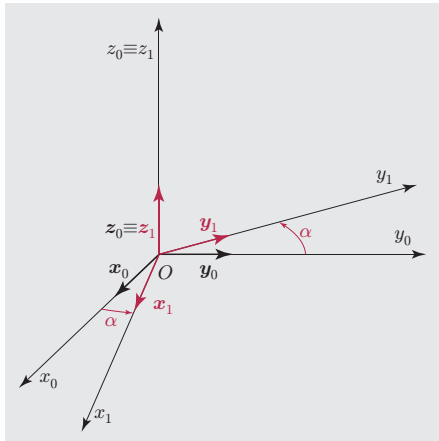
- Rotation matrices belong to the special orthonormal group $SO(m)$
- $SO(m)$ is the set of $m \times m$ orthonormal matrices with determinant equal to 1
  - in 3D space $m = 3$ and in the planar case $m = 2$
  - planar rotation matrix between frames 0 and 1 is

$$ {}^0\boldsymbol{R}_1 = \begin{pmatrix} \boldsymbol{x}_0^T \boldsymbol{x}_1 & \boldsymbol{x}_0^T \boldsymbol{y}_1 \\ \boldsymbol{y}_0^T \boldsymbol{x}_1 & \boldsymbol{y}_0^T \boldsymbol{y}_1 \end{pmatrix} = \begin{pmatrix} {}^0\boldsymbol{x}_1 & {}^0\boldsymbol{y}_1 \end{pmatrix} $$

Elementary Rotation Matrices

- Elementary rotation of frame O–xyz by an angle α about axis z

$$\boldsymbol{R}_z(\alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}$$



- Elementary rotation by angle β about axis x

$$\boldsymbol{R}_x(\beta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\beta & -\sin\beta \\ 0 & \sin\beta & \cos\beta \end{pmatrix}$$

- Elementary rotation by angle γ about axis y

$$\boldsymbol{R}_y(\gamma) = \begin{pmatrix} \cos\gamma & 0 & \sin\gamma \\ 0 & 1 & 0 \\ -\sin\gamma & 0 & \cos\gamma \end{pmatrix}$$

- For elementary rotations:

$$\boldsymbol{R}_k(-\vartheta) = \boldsymbol{R}_k^T(\vartheta) \qquad k = x, y, z$$

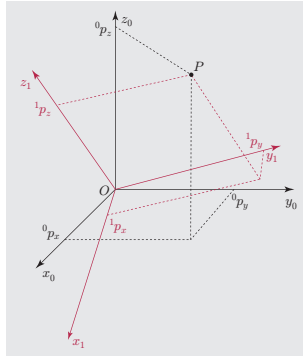
Rotation Matrix as a Coordinate Transformation

- Consider a point P described by vector $\boldsymbol{p}$ in frames O–$x_0y_0z_0$ and O–$x_1y_1z_1$

$$ {}^0\boldsymbol{p} = \begin{pmatrix} {}^0p_x & {}^0p_y & {}^0p_z \end{pmatrix}^T \qquad {}^1\boldsymbol{p} = \begin{pmatrix} {}^1p_x & {}^1p^1p_z \end{pmatrix}^T \qquad {}^0\boldsymbol{p} = {}^0\boldsymbol{R}_1\,{}^1\boldsymbol{p} $$

- ${}^0\boldsymbol{R}_1$ transforms coordinates of $\boldsymbol{p}$ from frame 1 to frame 0
- Using orthonormality of ${}^0\boldsymbol{R}_1$ the inverse coordinate transformation is

$$ {}^1\boldsymbol{p} = {}^1\boldsymbol{R}_0\,{}^0\boldsymbol{p} = {}^0\boldsymbol{R}_1^T\,{}^0\boldsymbol{p} $$



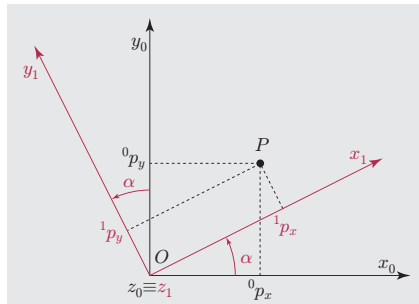
Example

- Two frames rotated by an angle α about axis z

$$
\begin{aligned}
{}^0p_x &= {}^1p_x \cos\alpha - {}^1p_y \sin\alpha \\
{}^0p_y &= {}^1p_x \sin\alpha + {}^1p_y \cos\alpha \\
{}^0p_z &= {}^1p_z
\end{aligned}
$$

$$
\Downarrow
$$

$$
{}^0\boldsymbol{p} = \boldsymbol{R}_z(\alpha)\,{}^1\boldsymbol{p}
$$



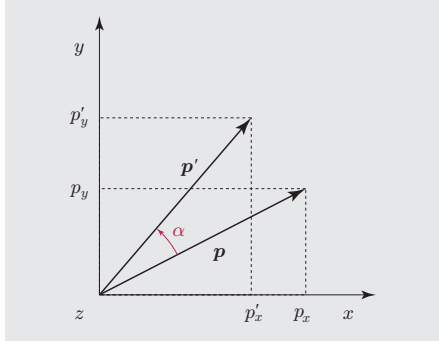
Rotation Matrix as an Operator

- A rotation matrix can be interpreted as an operator that rotates vectors in space
- Let $\boldsymbol{p}$ be a vector in frame O–xyz and let $\boldsymbol{p}' = \boldsymbol{R}\boldsymbol{p}$: vector $\boldsymbol{p}'$ has the same norm as $\boldsymbol{p}$ and is rotated according to $\boldsymbol{R}$
- Example: rotation of $\boldsymbol{p} = (p_x, p_y, p_z)$ in the plane xy by angle α about axis z

$$
\begin{aligned}
p'_x &= p_x \cos\alpha - p_y \sin\alpha \\
p'_y &= p_x \sin\alpha + p_y \cos\alpha \\
p'_z &= p_z
\end{aligned}
$$

$$\Downarrow$$

$$\boldsymbol{p}' = \boldsymbol{R}_z(\alpha)\boldsymbol{p}$$



Composition of Rotation Matrices

- Consider three frames O–$x_0y_0z_0$, O–$x_1y_1z_1$, O–$x_2y_2z_2$ with common origin
- Let ${}^j\boldsymbol{R}_i$ be the rotation matrix of frame i with respect to frame j

$$
{}^1\boldsymbol{p} = {}^1\boldsymbol{R}_2\,{}^2\boldsymbol{p}
$$

$$
{}^0\boldsymbol{p} = {}^0\boldsymbol{R}_1\,{}^1\boldsymbol{p} \qquad {}^0\boldsymbol{p} = {}^0\boldsymbol{R}_2\,{}^2\boldsymbol{p}
$$

$$
{}^0\boldsymbol{R}_2 = {}^0\boldsymbol{R}_1\,{}^1\boldsymbol{R}_2
$$

- Composition of successive rotations with respect to the current frame obtained by postmultiplication of rotation matrices
- Alternatively, composition of successive rotations with respect to the fixed frame obtained by premultiplication of rotation matrices

$$
{}^j\boldsymbol{R}_i = ({}^i\boldsymbol{R}_j)^{-1} = ({}^i\boldsymbol{R}_j)^T
$$

- Matrix products do not commute so the result depends on the order of rotations

Other Representations of Orientation

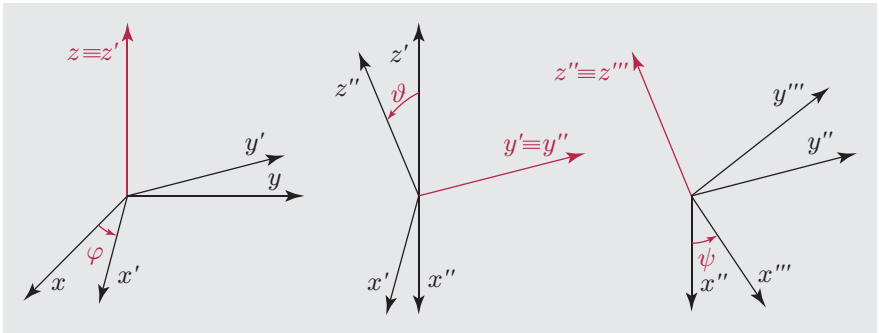
- Rotation matrices give a globally valid but redundant description of orientation
- They have nine elements constrained by six orthonormality conditions
- A minimal representation uses three independent parameters to describe a rigid body orientation in space
- A minimal representation of the special orthonormal group $SO(m)$ requires $m(m-1)/2$ parameters
  - thus three parameters are needed to parametrize $SO(3)$ for spatial orientation and one parameter for $SO(2)$ in the planar case

Euler Angles

- A minimal representation can be obtained using an ordered set of three Euler angles
- The three angles are organized in the array $\boldsymbol{\phi} = (\varphi, \vartheta, \psi)$
- A generic orientation is obtained by a sequence of three elementary rotations about coordinate axes
- Two successive rotations must not occur about the same axis to ensure a valid triplet of Euler angles $\Rightarrow$ out of 27 possible ordered triplets only 12 distinct Euler angles sequences are allowed

ZYZ Euler Angles

- ZYZ Euler angles are defined by three rotations about moving axes
  - first rotate the reference frame by angle φ about axis z using $\boldsymbol{R}_z(\varphi)$
  - then rotate the current frame by angle ϑ about axis y' using $\boldsymbol{R}_{y'}(\vartheta)$
  - finally rotate the current frame by angle ψ about axis z'' using $\boldsymbol{R}_{z''}(\psi)$



- The rotation matrix associated with ZYZ Euler angles is obtained by composing the three elementary rotations

$$\begin{aligned}\boldsymbol{R}_{zyz}(\boldsymbol{\phi}) &= \boldsymbol{R}_z(\varphi)\boldsymbol{R}_{y'}(\vartheta)\boldsymbol{R}_{z''}(\psi) \\ &= \begin{pmatrix} c_\varphi c_\vartheta c_\psi - s_\varphi s_\psi & -c_\varphi c_\vartheta s_\psi - s_\varphi c_\psi & c_\varphi s_\vartheta \\ s_\varphi c_\vartheta c_\psi + c_\varphi s_\psi & -s_\varphi c_\vartheta s_\psi + c_\varphi c_\psi & s_\varphi s_\vartheta \\ -s_\vartheta c_\psi & s_\vartheta s_\psi & c_\vartheta \end{pmatrix}\end{aligned}$$

- Explicit expressions of the elements of $\boldsymbol{R}_{zyz}(\boldsymbol{\phi})$ are used to solve the inverse problem for Euler angles

- Given a rotation matrix $\boldsymbol{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$
  - when $s_\vartheta > 0$

$$\begin{aligned} \varphi &= \operatorname{atan2}(r_{23}, r_{13}) \\ \vartheta &= \operatorname{atan2}(\sqrt{r_{13}^2 + r_{23}^2}, r_{33}) \\ \psi &= \operatorname{atan2}(r_{32}, -r_{31}) \end{aligned}$$

  - when $s_\vartheta < 0$

$$\begin{aligned} \varphi &= \operatorname{atan2}(-r_{23}, -r_{13}) \\ \vartheta &= \operatorname{atan2}(-\sqrt{r_{13}^2 + r_{23}^2}, r_{33}) \\ \psi &= \operatorname{atan2}(-r_{32}, r_{31}) \end{aligned}$$

- When $s_\vartheta = 0$ representation becomes singular and only $\varphi \pm \psi$ can be determined
  - these situations define representation singularities of the Euler angles

Roll–Pitch–Yaw Angles

- Another class of minimal orientation representations uses roll–pitch–yaw (RPY) angles
- RPY angles are defined by a sequence of three rotations about fixed axes of the original reference frame
- As for Euler angles there are 12 admissible RPY sequences satisfying the condition on successive axes
- RPY angles originated in aeronautics and are widely used in robotics as well

XYZ RPY Angles

- Each RPY sequence defined around fixed axes can be associated with an Euler angles sequence
- Premultiplication of rotation matrices is used when rotations are performed about fixed axes
- This corresponds to reversing the order of rotations in the product that defines the orientation
- A common choice is the XYZ RPY sequence around fixed axes which is equivalent to the ZYX Euler sequence

$$\begin{aligned}\boldsymbol{R}_{zyx}(\boldsymbol{\phi}) &= \boldsymbol{R}_z(\varphi)\boldsymbol{R}_{y'}(\vartheta)\boldsymbol{R}_{x''}(\psi) \\ &= \begin{pmatrix} c_\varphi c_\vartheta & c_\varphi s_\vartheta s_\psi - s_\varphi c_\psi & c_\varphi s_\vartheta c_\psi + s_\varphi s_\psi \\ s_\varphi c_\vartheta & s_\varphi s_\vartheta s_\psi + c_\varphi c_\psi & s_\varphi s_\vartheta c_\psi - c_\varphi s_\psi \\ -s_\vartheta & c_\vartheta s_\psi & c_\vartheta c_\psi \end{pmatrix}\end{aligned}$$

- Given a rotation matrix $\boldsymbol{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$
  - when $\vartheta \in (-\pi/2, \pi/2)$

$$\begin{aligned} \varphi &= \operatorname{atan2}(r_{21}, r_{11}) \\ \vartheta &= \operatorname{atan2}(-r_{31}, \sqrt{r_{32}^2 + r_{33}^2}) \\ \psi &= \operatorname{atan2}(r_{32}, r_{33}) \end{aligned}$$

  - when $\vartheta \in (\pi/2, \pi]$ or $\vartheta \in (-\pi, -\pi/2)$

$$\begin{aligned} \varphi &= \operatorname{atan2}(-r_{21}, -r_{11}) \\ \vartheta &= \operatorname{atan2}(-r_{31}, -\sqrt{r_{32}^2 + r_{33}^2}) \\ \psi &= \operatorname{atan2}(-r_{32}, -r_{33}) \end{aligned}$$

- When $\vartheta = \pm\pi/2$ representation becomes singular and only $\varphi \pm \psi$ can be determined
  - these situations define representation singularities of the RPY XYZ angles

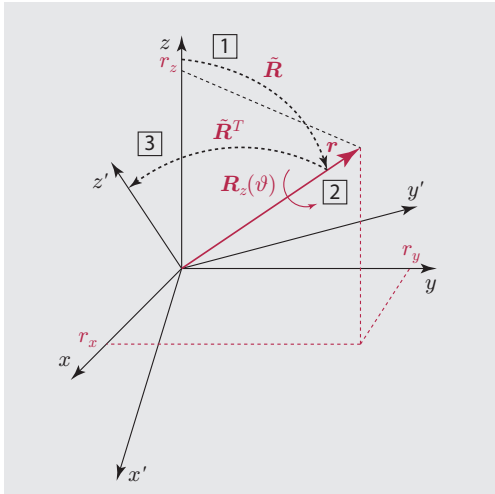
Axis–Angle

- A nonminimal representation uses four parameters given by a rotation angle and a unit axis in space
- Let $\boldsymbol{r} = (r_x, r_y, r_z)$ be the unit vector of the rotation axis with respect to frame O–xyz
- The orientation is described by the pair $(\boldsymbol{r}, \vartheta)$ where ϑ is the rotation angle about axis $\boldsymbol{r}$
- This representation is useful for trajectory planning of end-effector orientation

- Let

$$\tilde{\boldsymbol{R}} = \begin{pmatrix} \boldsymbol{n} & \boldsymbol{s} & \boldsymbol{r} \end{pmatrix}$$

be the rotation matrix that aligns coordinate axis $\boldsymbol{z}$ with unit axis $\boldsymbol{r}$



- The rotation matrix associated with axis–angle parameters $(\boldsymbol{r}, \vartheta)$ can be written as

$$\begin{aligned}\boldsymbol{R}(\boldsymbol{r}, \vartheta) &= \tilde{\boldsymbol{R}}\, \boldsymbol{R}_z(\vartheta)\, \tilde{\boldsymbol{R}}^T = \boldsymbol{r}\boldsymbol{r}^T + \left(\boldsymbol{n}\boldsymbol{n}^T + \boldsymbol{s}\boldsymbol{s}^T\right)\cos\vartheta + \left(\boldsymbol{s}\boldsymbol{n}^T - \boldsymbol{n}\boldsymbol{s}^T\right)\sin\vartheta \\ &= \boldsymbol{r}\boldsymbol{r}^T + (\boldsymbol{I} - \boldsymbol{r}\boldsymbol{r}^T)\cos\vartheta + \boldsymbol{S}(\boldsymbol{r})\sin\vartheta \\ &= \begin{pmatrix} r_x^2(1-c_\vartheta) + c_\vartheta & r_x r_y(1-c_\vartheta) - r_z s_\vartheta & r_x r_z(1-c_\vartheta) + r_y s_\vartheta \\ r_x r_y(1-c_\vartheta) + r_z s_\vartheta & r_y^2(1-c_\vartheta) + c_\vartheta & r_y r_z(1-c_\vartheta) - r_x s_\vartheta \\ r_x r_z(1-c_\vartheta) - r_y s_\vartheta & r_y r_z(1-c_\vartheta) + r_x s_\vartheta & r_z^2(1-c_\vartheta) + c_\vartheta \end{pmatrix}\end{aligned}$$

- $\boldsymbol{S}(\boldsymbol{r})$ is the skew-symmetric matrix built from the components of $\boldsymbol{r}$
  - this expression eliminates the need to specify auxiliary vectors orthogonal to $\boldsymbol{r}$

- Given a rotation matrix $\boldsymbol{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$
- When

$$\sin\vartheta = \pm\frac{1}{2}\sqrt{(r_{12}-r_{21})^2 + (r_{13}-r_{31})^2 + (r_{23}-r_{32})^2} \neq 0$$

two solutions are obtained ($r_x^2 + r_y^2 + r_z^2 = 1$)

$$\vartheta = \operatorname{atan2}\left(\sin\vartheta, \frac{r_{11}+r_{22}+r_{33}-1}{2}\right)$$

$$\boldsymbol{r} = \frac{1}{2\sin\vartheta}\begin{pmatrix} r_{32}-r_{23} \\ r_{13}-r_{31} \\ r_{21}-r_{12} \end{pmatrix}$$

- When $s_\vartheta = 0$ the representation becomes singular and it is necessary to refer to the particular expressions attained by $\boldsymbol{R}$ and find the solving formulas for $\vartheta = 0$ and $\vartheta = \pi$

Unit Quaternion

- Drawbacks of the axis–angle representation can be overcome by the unit quaternion representation
- A unit quaternion is defined as $\mathcal{Q} = \{\eta, \boldsymbol{\epsilon}\}$
  - the scalar part is
$$\eta = \cos\frac{\vartheta}{2}$$
  - the vector part is
$$\boldsymbol{\epsilon} = \sin\frac{\vartheta}{2}\boldsymbol{r}$$
  - η and $\boldsymbol{\epsilon}$ are subject to constraint
$$\eta^2 + \epsilon_x^2 + \epsilon_y^2 + \epsilon_z^2 = 1$$

- The pair $(\eta, \boldsymbol{\epsilon})$ encodes a rotation by angle ϑ about axis $\boldsymbol{r}$
- A rotation about $-\boldsymbol{r}$ by $-\vartheta$ gives the same quaternion as that associated with a rotation about $\boldsymbol{r}$ by ϑ
- The rotation matrix associated with unit quaternion parameters $(\eta, \boldsymbol{\epsilon})$ can be written as

$$\boldsymbol{R}(\eta, \boldsymbol{\epsilon}) = \begin{pmatrix} 2(\eta^2 + \epsilon_x^2) - 1 & 2(\epsilon_x\epsilon_y - \eta\epsilon_z) & 2(\epsilon_x\epsilon_z + \eta\epsilon_y) \\ 2(\epsilon_x\epsilon_y + \eta\epsilon_z) & 2(\eta^2 + \epsilon_y^2) - 1 & 2(\epsilon_y\epsilon_z - \eta\epsilon_x) \\ 2(\epsilon_x\epsilon_z - \eta\epsilon_y) & 2(\epsilon_y\epsilon_z + \eta\epsilon_x) & 2(\eta^2 + \epsilon_z^2) - 1 \end{pmatrix}$$

- Given a rotation matrix

$$\boldsymbol{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$$

  one possible solution for the corresponding unit quaternion is

$$\eta = \frac{1}{2}\sqrt{r_{11} + r_{22} + r_{33} + 1}$$

$$\boldsymbol{\epsilon} = \frac{1}{2}\begin{pmatrix} \operatorname{sign}(r_{32} - r_{23})\sqrt{r_{11} - r_{22} - r_{33} + 1} \\ \operatorname{sign}(r_{13} - r_{31})\sqrt{r_{22} - r_{33} - r_{11} + 1} \\ \operatorname{sign}(r_{21} - r_{12})\sqrt{r_{33} - r_{11} - r_{22} + 1} \end{pmatrix}$$

- These expressions avoid the singularities that appear in the axis–angle inverse solution

- The unit quaternion associated with $\boldsymbol{R}^{-1} = \boldsymbol{R}^T$ is

$$\mathcal{Q}^{-1} = \{\eta, -\boldsymbol{\epsilon}\}$$

- Let $\mathcal{Q}_1 = \eta_1, \boldsymbol{\epsilon}_1$ and $\mathcal{Q}_2 = \eta_2, \boldsymbol{\epsilon}_2$ correspond to $\boldsymbol{R}_1$ and $\boldsymbol{R}_2$
- The unit quaternion corresponding to the product $\boldsymbol{R}_1\boldsymbol{R}_2$ is given by

$$\mathcal{Q}_1 * \mathcal{Q}_2 = \{\eta_1\eta_2 - \boldsymbol{\epsilon}_1^T\boldsymbol{\epsilon}_2, \eta_1\boldsymbol{\epsilon}_2 + \eta_2\boldsymbol{\epsilon}_1 + \boldsymbol{\epsilon}_1 \times \boldsymbol{\epsilon}_2\}$$

- The quaternion product operator $*$ has identity element $\{1, \mathbf{0}\}$

Exponential Coordinates

- Exponential coordinates provide another equivalent expression of a rotation matrix in $SO(3)$
- This representation combines the axis–angle form with the exponential of a matrix
- For a generic square matrix $\boldsymbol{A}$ the exponential is defined by the power series

$$\exp \boldsymbol{A} = \boldsymbol{I} + \boldsymbol{A} + \frac{\boldsymbol{A}^2}{2!} + \frac{\boldsymbol{A}^3}{3!} + \cdots$$

- Let $\boldsymbol{S}(\boldsymbol{r})$ be the skew-symmetric matrix built from a unit vector $\boldsymbol{r}$
- One has the properties

$$\boldsymbol{S}^2(\boldsymbol{r}) = \boldsymbol{S}(\boldsymbol{r}) \cdot \boldsymbol{S}(\boldsymbol{r}) = \begin{pmatrix} -(r_y^2 + r_z^2) & r_x r_y & r_x r_z \\ r_y r_x & -(r_z^2 + r_x^2) & r_y r_z \\ r_z r_x & r_z r_y & -(r_x^2 + r_y^2) \end{pmatrix}$$

$$= \boldsymbol{r}\boldsymbol{r}^T - \boldsymbol{I}$$

$$\boldsymbol{S}^3(\boldsymbol{r}) = \boldsymbol{S}(\boldsymbol{r}) \cdot \boldsymbol{S}^2(\boldsymbol{r}) = \boldsymbol{S}(\boldsymbol{r}) \left(\boldsymbol{r}\boldsymbol{r}^T - \boldsymbol{I} \right) = -\boldsymbol{S}(\boldsymbol{r})$$

- These relations allow a compact expression of the exponential map of $\boldsymbol{S}(\vartheta \boldsymbol{r})$

- Consider the rotation matrix corresponding to the axis–angle representation

$$\begin{aligned}\boldsymbol{R}(\boldsymbol{r},\vartheta) &= \boldsymbol{I} - \boldsymbol{I} + \boldsymbol{r}\boldsymbol{r}^T + \left(\boldsymbol{I} - \boldsymbol{r}\boldsymbol{r}^T\right)\cos\vartheta + \boldsymbol{S}(\boldsymbol{r})\sin\vartheta \\ &= \boldsymbol{I} + \boldsymbol{S}(\boldsymbol{r})\sin\vartheta + \left(\boldsymbol{r}\boldsymbol{r}^T - \boldsymbol{I}\right) - \left(\boldsymbol{r}\boldsymbol{r}^T - \boldsymbol{I}\right)\cos\vartheta \\ &= \boldsymbol{I} + \boldsymbol{S}(\boldsymbol{r})\sin\vartheta + \boldsymbol{S}^2(\boldsymbol{r})\left(1 - \cos\vartheta\right)\end{aligned}$$

- Using the series of $\exp\boldsymbol{A}$ and the properties of $\boldsymbol{S}(\boldsymbol{r})$ the series simplifies to a combination of $\boldsymbol{I}$, $\boldsymbol{S}(\boldsymbol{r})$ and $\boldsymbol{S}^2(\boldsymbol{r})$

$$\begin{aligned}\exp\boldsymbol{S}(\vartheta\boldsymbol{r}) &= \boldsymbol{I} + \boldsymbol{S}(\boldsymbol{r})\vartheta + \boldsymbol{S}^2(\boldsymbol{r})\frac{\vartheta^2}{2!} + \boldsymbol{S}^3(\boldsymbol{r})\frac{\vartheta^3}{3!} + \cdots \\ &= \boldsymbol{I} + \left(\vartheta - \frac{\vartheta^3}{3!} + \frac{\vartheta^5}{5!} - \cdots\right)\boldsymbol{S}(\boldsymbol{r}) + \left(\frac{\vartheta^2}{2!} - \frac{\vartheta^4}{4!} + \frac{\vartheta^6}{6!} - \cdots\right)\boldsymbol{S}^2(\boldsymbol{r}) \\ &= \boldsymbol{I} + \sin\vartheta\,\boldsymbol{S}(\boldsymbol{r}) + \left(1 - \cos\vartheta\right)\boldsymbol{S}^2(\boldsymbol{r}) = \boldsymbol{R}(\boldsymbol{r},\vartheta)\end{aligned}$$

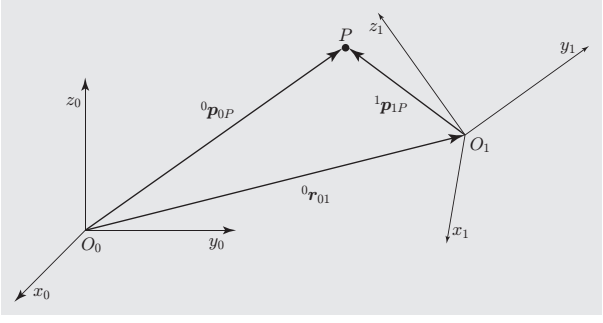
- This shows the equivalence with the axis–angle rotation matrix

Homogeneous Transformations

- The pose of a rigid body is described by a position vector for translation and a rotation matrix for orientation with respect to a reference frame
- It is convenient to combine translation and rotation into a single matrix operator
- This leads to the notion of homogeneous representation and homogeneous transformation matrix

- Consider frames O_0–$x_0y_0z_0$ and O_1–$x_1y_1z_1$
- Let ${}^0\boldsymbol{p}_{0P}$ be the position of P expressed in frame 0
- Let ${}^1\boldsymbol{p}_{1P}$ be the position of P expressed in frame 1
- Let ${}^0\boldsymbol{r}_{01}$ be the position of O_1 expressed in frame 0 and ${}^0\boldsymbol{R}_1$ the rotation matrix of frame 1 with respect to frame 0

$$ {}^0\boldsymbol{p}_{0P} = {}^0\boldsymbol{r}_{01} + {}^0\boldsymbol{R}_1 {}^1\boldsymbol{p}_{1P} $$



- Premultiplying by ${}^0\boldsymbol{R}_1^T$ and using orthonormality gives the inverse relation in terms of frame 1

$$ {}^1\boldsymbol{p}_{1P} = -{}^0\boldsymbol{R}_1^T {}^0\boldsymbol{r}_{01} + {}^0\boldsymbol{R}_1^T {}^0\boldsymbol{p}_{0P} $$

- Using ${}^1\boldsymbol{R}_0 = {}^0\boldsymbol{R}_1^T$ the same relation can be written as

$$ {}^1\boldsymbol{p}_{1P} = -{}^1\boldsymbol{R}_0 {}^0\boldsymbol{r}_{01} + {}^1\boldsymbol{R}_0 {}^0\boldsymbol{p}_{0P} $$

- For a vector ${}^i\boldsymbol{p}_{iP}$ with initial point in the origin of frame i define its homogeneous representation

$$ {}^i\tilde{\boldsymbol{p}}_{iP} = \begin{pmatrix} {}^i\boldsymbol{p}_{iP} \\ 1 \end{pmatrix} $$

- Homogeneous coordinates allow rigid motions to be represented by linear transformations in $\mathbb{R}^4$

- With homogeneous coordinates the coordinate transformation between frames 0 and 1 is described by a 4×4 matrix

$$ {}^0\boldsymbol{A}_1 = \begin{pmatrix} {}^0\boldsymbol{R}_1 & {}^0\boldsymbol{r}_{01} \\ \boldsymbol{0}^T & 1 \end{pmatrix} $$

- Since ${}^0\boldsymbol{r}_{01} \in I\!R^3$ and ${}^0\boldsymbol{R}_1 \in SO(3)$ one has ${}^0\boldsymbol{A}_1 \in SE(3) = I\!R^3 \times SO(3)$ where $SE(3)$ denotes the special Euclidean group
- Using homogeneous coordinates the coordinate transformation becomes linear

$$ {}^0\tilde{\boldsymbol{p}}_{0P} = {}^0\boldsymbol{A}_1 {}^1\tilde{\boldsymbol{p}}_{1P} $$

- ${}^0\boldsymbol{A}_1$ is called homogeneous transformation matrix and it compactly represents translation and rotation between two frames

- The inverse homogeneous transformation matrix ${}^1\boldsymbol{A}_0$ satisfies

$$ {}^1\tilde{\boldsymbol{p}}_{1P} = {}^1\boldsymbol{A}_0 {}^0\tilde{\boldsymbol{p}}_{0P} = \left({}^0\boldsymbol{A}_1\right)^{-1} {}^0\tilde{\boldsymbol{p}}_{0P} $$

- In block form the inverse matrix is

$$ {}^1\boldsymbol{A}_0 = \begin{pmatrix} {}^0\boldsymbol{R}_1^T & -{}^0\boldsymbol{R}_1^T {}^0\boldsymbol{r}_{01} \\ \boldsymbol{0}^T & 1 \end{pmatrix} = \begin{pmatrix} {}^1\boldsymbol{R}_0 & {}^1\boldsymbol{r}_{10} \\ \boldsymbol{0}^T & 1 \end{pmatrix} $$

- Homogeneous transformation matrices are not orthonormal
- In general the inverse is not equal to the transpose

$$ \boldsymbol{A}^{-1} \neq \boldsymbol{A}^T $$

- A sequence of coordinate transformations is obtained by composing homogeneous transformations

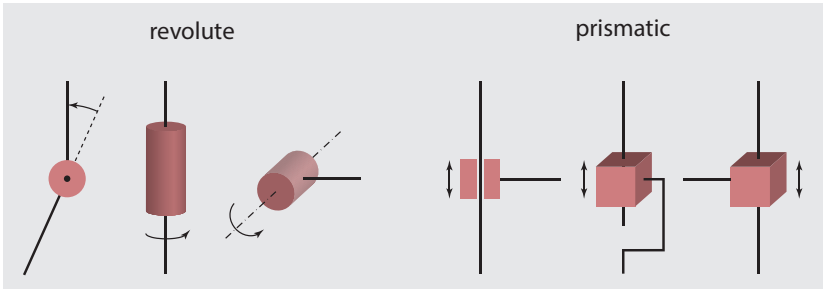
$$ {}^0\tilde{\boldsymbol{p}}_{0P} = {}^0\boldsymbol{A}_1 {}^1\boldsymbol{A}_2 \ldots {}^{n-1}\boldsymbol{A}_n, {}^n\tilde{\boldsymbol{p}}_{nP} $$

where ${}^{i-1}\boldsymbol{A}_i$ maps homogeneous coordinates of P from frame $i-1$ to frame i

Configuration Space

- Tools for describing the pose of a single rigid body are used to characterize the posture of a robot
- From a mechanical viewpoint a robot is a collection of rigid bodies connected by kinematic pairs called joints
- Robot joints are of two elementary types: revolute and prismatic, and their displacement is described by an angular or linear joint variable

- A compact notation uses R for revolute and P for prismatic so that a string such as RPR or 3R identifies the sequence and type of joints in the kinematic structure



- The above mechanical characterization applies to both fixed-base manipulators and mobile robots
- In manipulators rigid bodies called links form a kinematic chain with one end connected to a fixed base and the other to an end-effector

- The end-effector is a gripper or another tool used to manipulate objects in the workspace $\mathcal{W}$
- In mobile robots the base is itself a body that moves in the workspace and additional bodies may be connected via joints as links of mobile manipulators, trailers or limbs
- A robot moves in a workspace $\mathcal{W}$ which can be either $\mathbb{R}^2$ or $\mathbb{R}^3$ corresponding to planar or spatial robots
- The posture of a robot is the position of all its points in the workspace
- The posture is uniquely determined by a set of generalized coordinates
- Generalized coordinates for robots are essentially of two types: Cartesian coordinates describing positions in $\mathbb{R}^m$ and orientation coordinates describing spatial orientations
- The dimension of $SO(m)$ is $m(m-1)/2$

- The values of the generalized coordinates associated with a posture define a configuration of the robot
- The configuration space $\mathcal{C}$ is the set of all admissible configurations of the robot
- In general $\mathcal{C}$ is a Cartesian product of Euclidean and orientation spaces
- Configuration space plays a central role in modelling, planning and control of robotic systems

- Some examples of configuration spaces
  - a single-body polygonal robot in $\mathcal{W} = I\!R^2$ has configuration space $\mathcal{C} = I\!R^2 \times SO(2)$ whose dimension is $n = 3$
  - a single-body polyhedral robot in $\mathcal{W} = I\!R^3$ has configuration space $\mathcal{C} = I\!R^3 \times SO(3)$ whose dimension is $n = 6$
  - a fixed-base planar manipulator with n revolute joints has configuration space $\mathcal{C}$ which is a subset of $(I\!R^2 \times SO(2))^n$ and is fully described by the n joint variables
  - a fixed-base spatial manipulator with n revolute joints has configuration space $\mathcal{C}$ which is a subset of $(I\!R^3 \times SO(3))^n$ with dimension $6n - 5n = n$ and is fully described by n relative or absolute joint variables
  - a fixed-base spatial manipulator with n_1 revolute and n_2 prismatic joints has configuration space $\mathcal{C}$ which is a subset of $(I\!R^3 \times SO(3))^n$ with dimension $6n - 5n = n$ and is fully described by n joint variables, where n_1 variables are angular and n_2 variables are linear
  - a differential-drive tractor with a trailer moving in $\mathcal{W} = I\!R^2$ has configuration space $\mathcal{C}$ which is a subset of $(I\!R^2 \times SO(2))^2$. If the trailer is connected to the tractor by a revolute joint, the configuration of the robot can be fully described by the pose of the tractor and the orientation of the trailer, and thus the dimension of $\mathcal{C}$ is $n = 4$

- If n is the dimension of $\mathcal{C}$, the configuration of the robot can be described by the vector of generalized coordinates

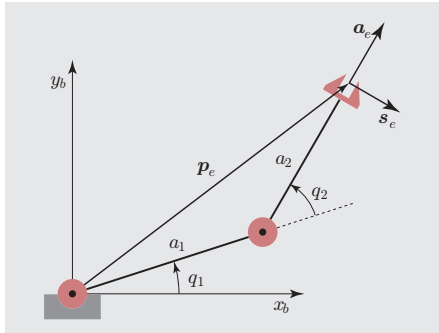
$$\boldsymbol{q} = \begin{pmatrix} q_1 \\ \vdots \\ q_n \end{pmatrix} \in I\!R^n$$

- The number of degrees of freedom (DoFs) of the robot is $\eta = n - k$
  - here k is the number of kinematic constraints to which the robot is subject
  - if $k = 0$ the robot is free to move instantaneously in any direction of $\mathcal{C}$ and η coincides with n

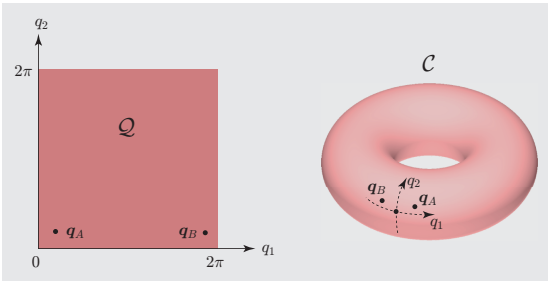
- Consider a planar arm with two links and two revolute joints
  - its configuration is $\boldsymbol{q} = (q_1, q_2)$ where q_1 and q_2 are the relative joint coordinates
  - assuming no joint limits the configuration space is

$$\mathcal{Q} = \boldsymbol{q} = (q_1, q_2) : q_1 \in [0, 2\pi), q_2 \in [0, 2\pi)$$

  - this representation is injective but not topologically correct because opposite edges of the square correspond to the same configurations



- Topology of the 2R configuration space



- Whenever the configuration includes angular coordinates the configuration space $\mathcal{C}$ is not a Euclidean space
- In general $\mathcal{C}$ is an n-dimensional manifold, namely a space that locally resembles $\mathbb{R}^n$ but has a different global topology
- A manifold can be covered by charts each defined by a locally invertible continuous mapping with a continuous inverse

Distance

- Many planning and control methods require a notion of distance in configuration space
- In principle the distance between two points in $\mathcal{C}$ could be defined as the length of the shortest path between them: the geodesic
- Computing geodesics over generic manifolds is difficult and different metrics are typically used in robotics

- Given a configuration $\boldsymbol{q}$ let $\mathcal{B}(\boldsymbol{q})$ be the subset of the workspace $\mathcal{W}$ occupied by the robot and let $p(\boldsymbol{q})$ be the position in $\mathcal{W}$ of a point p on $\mathcal{B}(\boldsymbol{q})$
- Intuitively configurations are close if the regions $\mathcal{B}(\boldsymbol{q}_A)$ and $\mathcal{B}(\boldsymbol{q}_B)$ are close to each other
- A distance function with this property is

$$d_1(\boldsymbol{q}_A, \boldsymbol{q}_B) = \max_{p \in \mathcal{B}} \|p(\boldsymbol{q}_A) - p(\boldsymbol{q}_B)\|$$

- This displacement metric measures the maximum displacement in $\mathcal{W}$ induced by the two configurations on a point moving over the robot

- Direct use of d_1 may be cumbersome because it requires characterizing the volume occupied by the robot and computing a maximum over all points
- A practical alternative is to use a finite set of control points $p_1, \ldots, p_N$ distributed along the robot bodies and define

$$d_2(\boldsymbol{q}_A, \boldsymbol{q}_B) = \sum_{i=1}^{N} \|p_i(\boldsymbol{q}_A) - p_i(\boldsymbol{q}_B)\|$$

- Sometimes the simple Euclidean norm of the difference of configurations is used

$$d_3(\boldsymbol{q}_A, \boldsymbol{q}_B) = \|\boldsymbol{q}_A - \boldsymbol{q}_B\|$$

- One must keep in mind that this last definition is only approximate when generalized coordinates include angular variables

Task Space and Direct Kinematics

- Robots are used to perform physical operations called tasks in their workspace
- In industrial settings a manipulator is often required to execute tasks that concern only the position of the end-effector or its full pose
- Single-body wheeled robots in service applications have tasks typically defined by the planar position of a point on the robot body and its orientation or by the position of the tractor and the relative orientation between consecutive bodies
- In legged robots tasks are usually of composite nature involving for instance the pose of a frame attached to the trunk and another attached to the foot that is swinging during the step
- In visual servoing tasks may be specified in terms of quantities defined on the image plane of a camera

- In general a task is described in terms of a set of m suitable task coordinates collected in the vector $\boldsymbol{y}$
- The set of all possible values of $\boldsymbol{y}$ defines the task space $\mathcal{T}$ (also called operational space)
- When some task coordinates are angular $\mathcal{T}$ is a manifold rather than a Euclidean space
- Task coordinates depend on the configuration of the robot and are defined so as to describe the operations the robot should perform
- Task coordinates can be expressed as a function of the generalized coordinates $\boldsymbol{q}$ via an algebraic relationship called direct kinematics

$$\boldsymbol{y} = \boldsymbol{k}(\boldsymbol{q})$$

where $\boldsymbol{k}(\cdot)$ is referred to as direct kinematics map or task function

- A necessary condition for task realizability is $m \leq n$ where n is the dimension of the configuration space

Kinematic Redundancy

- A robot is said to be kinematically redundant if $m < n$ with respect to a given task
- The difference between n and m is called degree of redundancy
- In the presence of redundancy any given value of the task vector $\boldsymbol{y}$ can be realized by infinitely many configurations $\boldsymbol{q}$
- Redundancy is always defined with respect to a specific task so that the same robot may be redundant for one task and nonredundant for another
- For manipulators redundancy occurs whenever there are more joint variables than task coordinates
- A six-DoF manipulator that has to control only the position of the end-effector is redundant with respect to that task
- A manipulator may be nonredundant when the task requires full pose control of the end-effector and redundant when the task requires only a subset of pose variables, for instance when the approach direction is inconsequential for the task

- One may wonder why use a robot that is kinematically redundant for a given task
- Redundancy significantly increases the flexibility of the robot in performing tasks
- A fixed-base manipulator with seven DoFs arranged as in the human arm,i.e., three in the shoulder, one in the elbow, and three in the wrist (not considering the hand DoFs) is kinematically redundant with respect to the pose of the hand
  - in fact, if the hand is fixed at a certain position and orientation in space, it is still possible to move the elbow along a circle thanks to the single degree of redundancy
- Redundancy can be exploited to satisfy additional requirements such as avoiding collisions or singularities, keeping joints within limits or optimizing performance indices while executing the primary task
- A more in-depth treatment of kinematic redundancy is deferred to the study of differential kinematics

Direct Kinematics of Manipulators

- Direct kinematics $\boldsymbol{y} = \boldsymbol{k}(\boldsymbol{q})$ relates configuration variables to task coordinates
- For a fixed-base manipulator and tasks on the end-effector, the relevant map is the end-effector direct kinematics
- Joint variables collected in $\boldsymbol{q}$ fully describe the manipulator configuration
- The task vector for end-effector pose is denoted by $\boldsymbol{y}_e$

End-Effector Tasks

- The pose of the end-effector is described by position $\boldsymbol{p}_e$ and orientation $\boldsymbol{\phi}_e$

$$\boldsymbol{y}_e = \begin{pmatrix} \boldsymbol{p}_e \\ \boldsymbol{\phi}_e \end{pmatrix}$$

  - $\boldsymbol{p}_e \in I\!R^3$ is the position of the end-effector reference point with respect to the base frame
  - $\boldsymbol{\phi}_e$ is a suitable three-dimensional parametrization of its orientation
- The task space for end-effector pose is $\mathcal{T} = I\!R^3 \times SO(3)$ with dimension $m = 6$
- Expressing $\boldsymbol{p}_e$ and $\boldsymbol{\phi}_e$ as functions of the joint variables $\boldsymbol{q}$ defines the end-effector direct kinematics map

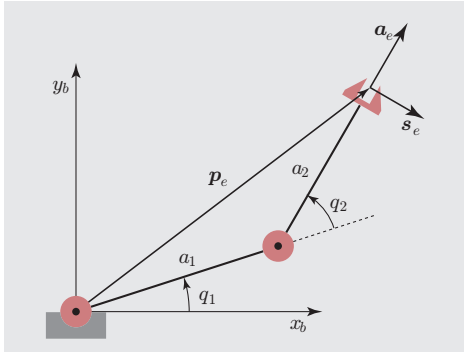
$$\boldsymbol{y}_e = \begin{pmatrix} \boldsymbol{p}_e(\boldsymbol{q}) \\ \boldsymbol{\phi}_e(\boldsymbol{q}) \end{pmatrix} = \boldsymbol{k}_e(\boldsymbol{q})$$

- Depending on the task only a subset of pose variables may be included in $\boldsymbol{y}_e$ such as pure positioning or pure orientation

- Pose task for 2R planar arm

$$\boldsymbol{y}_e = \begin{pmatrix} \boldsymbol{p}_e \\ \boldsymbol{\phi}_e \end{pmatrix} = \begin{pmatrix} p_x \\ p_y \\ \phi \end{pmatrix}$$

  - (p_x, p_y) are the Cartesian coordinates of the representative point of the end-effector
  - ϕ is its planar orientation with respect to the x axis



- The expression of the direct kinematics map is

$$\begin{pmatrix} p_x \\ p_y \\ \phi \end{pmatrix} = \boldsymbol{k}_e(\boldsymbol{q}) = \begin{pmatrix} a_1 c_1 + a_2 c_{12} \\ a_1 s_1 + a_2 s_{12} \\ q_1 + q_2 \end{pmatrix}$$

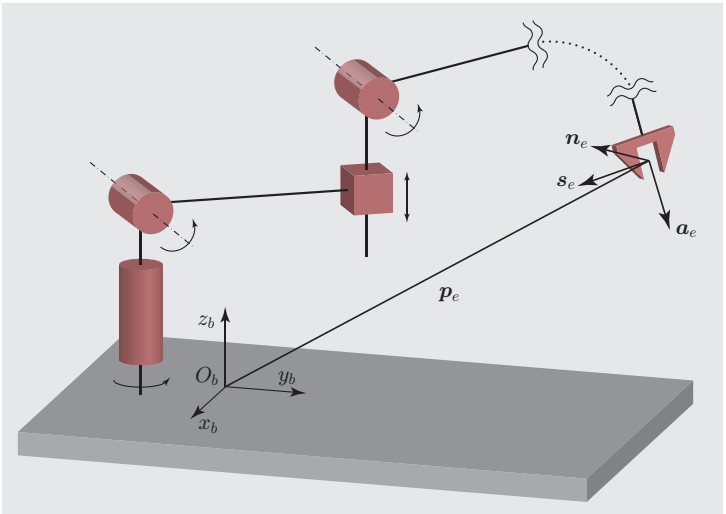
  - the manipulator has only two DoFs, so it is not possible to independently control position and orientation of the end-effector

- An alternative description uses an end-effector frame $(\boldsymbol{n}_e, \boldsymbol{s}_e, \boldsymbol{a}_e)$ attached to the tool with reference to a gripper
  - $\boldsymbol{a}_e$ is a unit vector along the approach direction
  - $\boldsymbol{s}_e$ is a unit vector normal to $\boldsymbol{a}_e$ in the sliding plane of the jaws
  - $\boldsymbol{n}_e$ is a unit vector normal so that $(\boldsymbol{n}_e, \boldsymbol{s}_e, \boldsymbol{a}_e)$ is right-handed

$$
{}^b\boldsymbol{T}_e = \begin{pmatrix} {}^b\boldsymbol{n}_e & {}^b\boldsymbol{s}_e & {}^b\boldsymbol{a}_e & {}^b\boldsymbol{p}_e \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} {}^b\boldsymbol{R}_e & {}^b\boldsymbol{p}_e \\ \boldsymbol{0}^T & 1 \end{pmatrix}
$$

- All vectors are expressed with respect to the base frame and ${}^b\boldsymbol{R}_e$ is the rotation matrix of the end-effector frame

- End-effector frame and base frame



- For the 2R planar arm with appropriate base and end-effector frames one obtains

$$
{}^b\boldsymbol{T}_e(\boldsymbol{q}) = \begin{pmatrix} {}^b\boldsymbol{n}_e(\boldsymbol{q}) & {}^b\boldsymbol{s}_e(\boldsymbol{q}) & {}^b\boldsymbol{a}_e(\boldsymbol{q}) & {}^b\boldsymbol{p}_e(\boldsymbol{q}) \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & s_{12} & c_{12} & a_1c_1 + a_2c_{12} \\ 0 & -c_{12} & s_{12} & a_1s_1 + a_2s_{12} \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

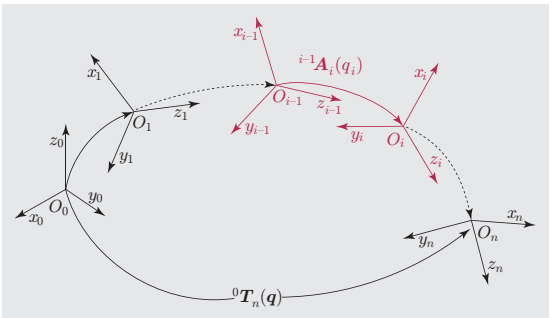
- Position and orientation of the end-effector can be directly extracted from ${}^b\boldsymbol{T}_e(\boldsymbol{q})$

Open-Chain Manipulators

- A kinematic chain is open if there exists only one path of rigid bodies and joints connecting the two ends of the chain
- A kinematic chain is closed if the links form at least one loop
- Computing direct kinematics for open chains is simpler

- Consider an open-chain manipulator with $n+1$ links and n joints and attach a frame to each link
- Let ${}^{i-1}\boldsymbol{A}_i(q_i)$ be the homogeneous transformation from frame i to frame $i-1$
- The pose of frame n with respect to frame 0 is obtained by composition as

$$ {}^0\boldsymbol{T}_n(\boldsymbol{q}) = {}^0\boldsymbol{A}_1(q_1){}^1\boldsymbol{A}_2(q_2)\dots{}^{n-1}\boldsymbol{A}_n(q_n) = \prod_{i=1}^{n} {}^{i-1}\boldsymbol{A}_i(q_i) $$



- Let ${}^{b}\boldsymbol{T}_{0}$ be the homogeneous transformation from the base frame to frame 0
- Let ${}^{n}\boldsymbol{T}_{e}$ be the homogeneous transformation from frame n to the end-effector frame
- The pose of the end-effector frame with respect to the base frame is

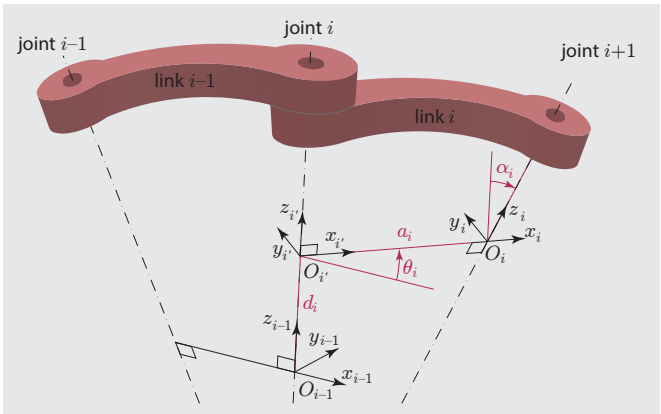
$$
{}^{b}\boldsymbol{T}_{e}(\boldsymbol{q}) = \begin{pmatrix} {}^{b}\boldsymbol{n}_{e}(\boldsymbol{q}) & {}^{b}\boldsymbol{s}_{e}(\boldsymbol{q}) & {}^{b}\boldsymbol{a}_{e}(\boldsymbol{q}) & {}^{b}\boldsymbol{p}_{e}(\boldsymbol{q}) \\ 0 & 0 & 0 & 1 \end{pmatrix} = {}^{b}\boldsymbol{T}_{0} \left(\prod_{i=1}^{n} {}^{i-1}\boldsymbol{A}_{i}(q_{i}) \right) {}^{n}\boldsymbol{T}_{e}
$$

- This expression provides an effective way to compute the direct kinematics of an open-chain manipulator iteratively

Denavit–Hartenberg Convention

- For systematic computation of direct kinematics it is useful to adopt a convention for attaching frames to the links
- The Denavit–Hartenberg (DH) convention associates one frame to each link using four parameters per link
- The DH convention exploits the serial structure of open-chain manipulators

- For each joint i the DH parameters are $\alpha_i, a_i, d_i, \theta_i$
  - d_i and θ_i describe the relative displacement along and about the joint axis
  - a_i and α_i describe the link geometry between joint i and joint $i+1$



- Choose axis z_i along the axis of joint $i+1$
- Locate the origin O_i at the intersection of axis z_i with the common normal to axes z_{i-1} and z_i
- Choose axis x_i along the common normal to axes z_{i-1} and z_i with positive direction from joint i to joint $i+1$
- Choose axis y_i so as to obtain a right-handed frame
- For frame 0, the direction of z_0 is specified while O_0 and x_0 can be chosen conveniently, for example, coincident with the base frame
- For frame n, axis x_n intersects and is orthogonal to z_{n-1}. and axis z_n is aligned with z_{n-1} for a revolute joint or chosen orthogonal to x_n for a prismatic joint

- Once the frames are assigned, the DH parameters $\alpha_i, a_i, d_i, \theta_i$ are collected in a table
  - θ_i angle between axes x_{i-1} and x_i about the axis z_{i-1}, taken positive if the rotation is counter-clockwise;
  - d_i distance between O_{i-1} and $O_{i'}$, evaluated along z_{i-1};
  - a_i distance between $O_{i'}$ and O_i, evaluated along x_i;
  - α_i angle between axes z_{i-1} and z_i about the axis x_i, taken positive if the rotation is counter-clockwise.

- Computing the transformation that superposes frame $i-1$ to frame i according to the following steps
  - rotate and translate frame $i-1$ about and along axis z_{i-1} to obtain frame i' using d_i and θ_i

$$ {}^{i-1}\boldsymbol{A}_{i'} = \begin{pmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

  - translate frame i' by a_i along axis $x_i = x_{i'}$ and rotate it by α_i about x_i to obtain frame i

$$ {}^{i'}\boldsymbol{A}_{i} = \begin{pmatrix} 1 & 0 & 0 & a_i \\ 0 & \cos\alpha_i & -\sin\alpha_i & 0 \\ 0 & \sin\alpha_i & \cos\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

- The homogeneous transformation from frame i to frame $i-1$ is obtained by composing the two elementary transformations

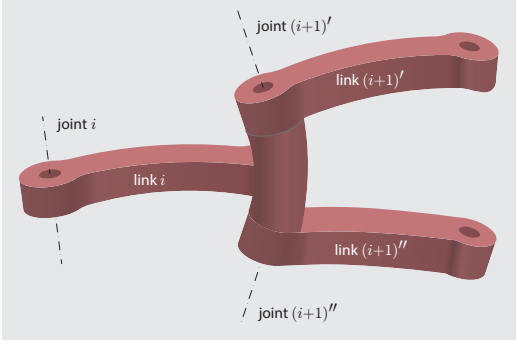
$$
{}^{i-1}\boldsymbol{A}_i(q_i) = \begin{pmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

- For a revolute joint θ_i is the joint variable q_i and d_i is constant
- For a prismatic joint d_i is the joint variable q_i and θ_i is constant

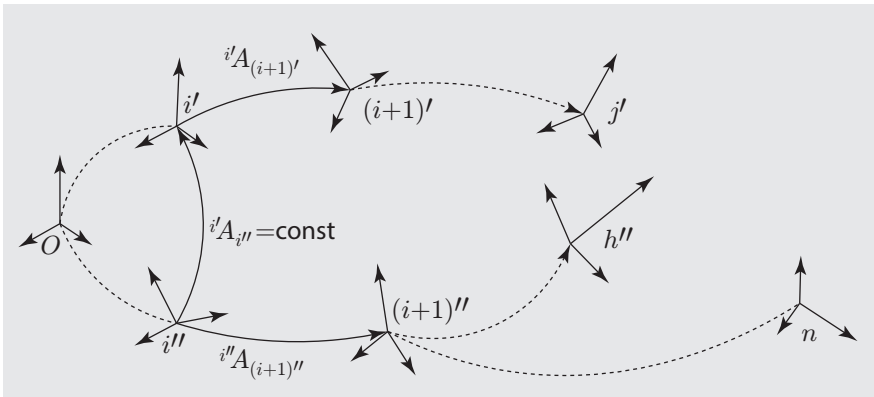
- Procedure for direct kinematics of open chains with DH convention
  - attach frames to all links according to the DH rules
  - write the table of DH parameters for the manipulator
  - express each homogeneous transformation ${}^{i-1}\boldsymbol{A}_i(q_i)$ in terms of the corresponding joint variable
  - compose the transformations from the base frame to the end-effector frame to obtain ${}^{b}\boldsymbol{T}_e(\boldsymbol{q})$

Extension to Closed-Chain Manipulators

- In a closed-chain manipulator links form at least one loop
- The number of joint variables generally exceeds the number of DoFs of the structure
- Joint variables are not all independent and are related by loop closure constraints



- Consider a closed-chain manipulator with $n+1$ links and a set of n joints including actuated and unactuated joints
- Choose one joint c to open the loop and obtain an equivalent open chain



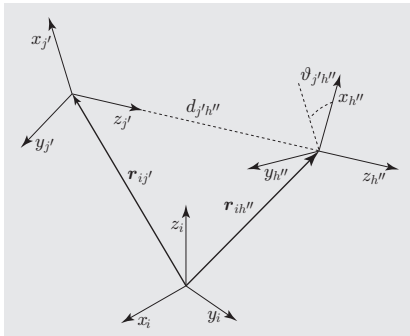
- The joint variables of the open chain are partitioned into two groups on the two sides of the cut joint
- Let frames i' and j' denote the two sides of the cut joint on one branch and frames i'' and h'' on the other branch
- Let $\boldsymbol{q}' = (q_{(i+1)'}, \ldots, q_{j'})$ and $\boldsymbol{q}'' = (q_{(i+1)''}, \ldots, q_{h''})$ be the joint variables on the two paths

$$ {}^{i'}\boldsymbol{A}_{j'}(\boldsymbol{q}') = {}^{i'}\boldsymbol{A}_{(i+1)'}(q_{(i+1)'}) \ldots {}^{(j-1)'}\boldsymbol{A}_{j'}(q_{j'}) $$

$$ {}^{i''}\boldsymbol{A}_{h''}(\boldsymbol{q}'') = {}^{i''}\boldsymbol{A}_{(i+1)''}(q_{(i+1)''}) \ldots {}^{(h-1)''}\boldsymbol{A}_{h''}(q_{h''}) $$

- The two transformations from frame i to frame j' and from frame i to frame h'' must be consistent to close the loop

- Let ${}^{i}\boldsymbol{z}_{j'}(\boldsymbol{q}')$ and ${}^{i}\boldsymbol{z}_{h''}(\boldsymbol{q}'')$ be the unit vectors of axes $z_{j'}$ and $z_{h''}$ expressed in frame i
- Let ${}^{i}\boldsymbol{r}_{ij'}(\boldsymbol{q}')$ and ${}^{i}\boldsymbol{r}_{ih''}(\boldsymbol{q}'')$ be the position vectors of the origins $O_{j'}$ and $O_{h''}$ expressed in frame i



$$ {}^{i}\boldsymbol{z}_{j'}(\boldsymbol{q}') = {}^{i}\boldsymbol{z}_{h''}(\boldsymbol{q}'') $$

$$ {}^{i}\boldsymbol{x}_{j'}^{T}(\boldsymbol{q}')\,{}^{i}\boldsymbol{x}_{h''}(\boldsymbol{q}'') = \cos\theta_{j'h''} $$

$$ {}^{i}\boldsymbol{R}_{j'}^{T}(\boldsymbol{q}')\left({}^{i}\boldsymbol{r}_{ij'}(\boldsymbol{q}') - {}^{i}\boldsymbol{r}_{ih''}(\boldsymbol{q}'')\right) = \begin{pmatrix} 0 & 0 & d_{j'h''} \end{pmatrix}^{T} $$

$$ \begin{pmatrix} {}^{i}\boldsymbol{x}_{j'}^{T}(\boldsymbol{q}') \\ {}^{i}\boldsymbol{y}_{j'}^{T}(\boldsymbol{q}') \end{pmatrix} \left({}^{i}\boldsymbol{r}_{ij'}(\boldsymbol{q}') - {}^{i}\boldsymbol{r}_{ih''}(\boldsymbol{q}'')\right) = \begin{pmatrix} 0 \\ 0 \end{pmatrix} $$

- These equalities enforce position and orientation consistency at the cut joint

- If the cut joint c is revolute the angle $\theta_{j'h''}$ varies while $d_{j'h''}$ is constant

$$\begin{cases} {}^i\boldsymbol{z}_{j'}(\boldsymbol{q}') = {}^i\boldsymbol{z}_{h''}(\boldsymbol{q}'') \\ {}^i\boldsymbol{R}_{j'}^T(\boldsymbol{q}') \left({}^i\boldsymbol{r}_{ij'}(\boldsymbol{q}') - {}^i\boldsymbol{r}_{ih''}(\boldsymbol{q}'') \right) = \begin{pmatrix} 0 & 0 & d_{j'h''} \end{pmatrix}^T \end{cases}$$

- If the cut joint c is prismatic both $\theta_{j'h''}$ and $d_{j'h''}$ are constant and the constraints are

$$\begin{cases} {}^i\boldsymbol{z}_{j'}(\boldsymbol{q}') = {}^i\boldsymbol{z}_{h''}(\boldsymbol{q}'') \\ {}^i\boldsymbol{x}_{j'}^T(\boldsymbol{q}') {}^i\boldsymbol{x}_{h''}(\boldsymbol{q}'') = \cos\theta_{j'h''} \\ \begin{pmatrix} {}^i\boldsymbol{x}_{j'}^T(\boldsymbol{q}') \\ {}^i\boldsymbol{y}_{j'}^T(\boldsymbol{q}') \end{pmatrix} \left({}^i\boldsymbol{r}_{ij'}(\boldsymbol{q}') - {}^i\boldsymbol{r}_{ih''}(\boldsymbol{q}'') \right) = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \end{cases}$$

- In both cases six scalar loop-closure constraints are obtained

- Collect all joint variables of the open chain in the vector

$$\boldsymbol{q}_o = \begin{pmatrix} \boldsymbol{q} \\ \boldsymbol{\varrho} \end{pmatrix}$$

- Loop-closure constraints allow solving for the dependent joint variables as functions of the independent ones

$$\boldsymbol{\varrho} = \boldsymbol{\varrho}(\boldsymbol{q})$$

- The direct kinematics of the closed-chain manipulator is obtained by applying the open-chain formula to $\boldsymbol{q}_o$ and then substituting $\boldsymbol{\varrho}(\boldsymbol{q})$

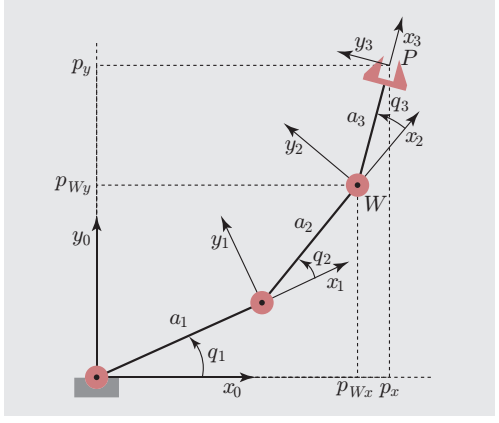
- Procedure for direct kinematics of closed chains with DH convention
  - in the closed chain, select one joint that is not actuated. Assume that the joint is cut open so as to obtain an open chain in a tree structure
  - compute the homogeneous transformations according to DH convention
  - find the equality constraints for the two frames connected by the cut joint
  - solve the constraints for a reduced number of joint variables
  - express the homogeneous transformations in terms of the above joint variables and compute the direct kinematics function by composing the various transformations along a direct path from the base frame to the end-effector frame

Examples of Direct Kinematics

- Direct kinematics computation for typical industrial manipulators
- Use of DH frames and parameters to obtain homogeneous transformations
- Examples include planar arms, anthropomorphic arms, SCARA manipulators and wrists
- Kinematic parameters for DH table may differ from manufacturer data sheet
- Encoder readings may require pre-processing before using DH kinematics

3R Planar Arm

- Three revolute joints with parallel axes normal to the plane (x_0, y_0)
- Link frames attached according to the DH convention



- DH parameters collected in the table
- All joint axes parallel and all d_i equal to zero

Link	α_i	a_i	d_i	θ_i
1	0	a_1	0	q_1
2	0	a_2	0	q_2
3	0	a_3	0	q_3

- For revolute joints the homogeneous transformation ${}^{i-1}\boldsymbol{A}_i(q_i)$ has the same structure for all links

$$ {}^{i-1}\boldsymbol{A}_i(q_i) = \begin{pmatrix} c_i & -s_i & 0 & a_i c_i \\ s_i & c_i & 0 & a_i s_i \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad i = 1, 2, 3 $$

- End-effector frame: homogeneous transformation from the base frame to frame 3

$$ {}^{0}\boldsymbol{T}_3(\boldsymbol{q}) = {}^{0}\boldsymbol{A}_1 {}^{1}\boldsymbol{A}_2 {}^{2!}\boldsymbol{A}_3 = \begin{pmatrix} c_{123} & -s_{123} & 0 & a_1 c_1 + a_2 c_{12} + a_3 c_{123} \\ s_{123} & c_{123} & 0 & a_1 s_1 + a_2 s_{12} + a_3 s_{123} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

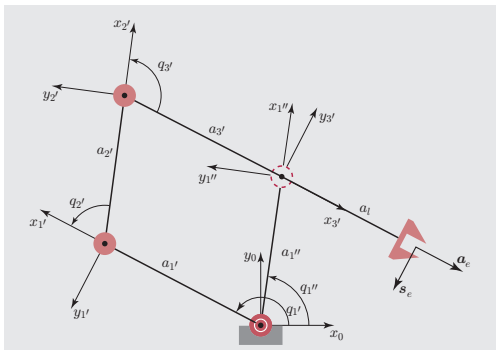
- Task vector includes planar position and orientation of the end-effector

$$\begin{pmatrix} p_x \\ p_y \\ \phi \end{pmatrix} = \boldsymbol{k}_e(\boldsymbol{q}) = \begin{pmatrix} a_1 c_1 + a_2 c_{12} + a_3 c_{123} \\ a_1 s_1 + a_2 s_{12} + a_3 s_{123} \\ q_1 + q_2 + q_3 \end{pmatrix}$$

- If only Cartesian position (p_x, p_y) is required the arm has one redundant DoF with respect to the task

Parallelogram Arm

- Four links arranged in a parallelogram structure
- Two branches starting from the base links $1'$ and $1''$
- End-effector attached to link $3'$ through a fixed offset



- Revolute joints with parallel axes and equal link lengths on opposite sides
- $a_{1'} = a_{3'} = a_w$ and $a_{2'} = a_{1''} = a_h$ due to the parallelogram structure

Link	α_i	a_i	d_i	θ_i
$1'$	0	a_w	0	$q_{1'}$
$2'$	0	a_h	0	$q_{2'}$
$3'$	0	a_w	0	$q_{3'}$
$1''$	0	a_h	0	$q_{1''}$

- Transformation along the branch $0 \to 1' \to 2' \to 3'$

$$
{}^0\boldsymbol{A}_{3'}(\boldsymbol{q}') = {}^0\boldsymbol{A}_{1'}{}^{1'}\boldsymbol{A}_{2'}{}^{2'}\boldsymbol{A}_{3'} = \begin{pmatrix} c_{1'2'3'} & -s_{1'2'3'} & 0 & a_w c_{1'} + a_h c_{1'2'} + a_w c_{1'2'3'} \\ s_{1'2'3'} & c_{1'2'3'} & 0 & a_w s_{1'} + !a_h s_{1'2'} + a_w s_{1'2'3'} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

where $\boldsymbol{q}' = (q_{1'}, q_{2'}, q_{3'})$

- Transformation along the branch $0 \to 1''$

$$
{}^0\boldsymbol{A}_{1''}(q_{1''}) = \begin{pmatrix} c_{1''} & -s_{1''} & 0 & a_h c_{1''} \\ s_h & c_{1''} & 0 & a_h s_{1''} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

- End-effector frame has origin at distance a_l from $O_{3'}$ along $x_{3'}$ and unit vector $\boldsymbol{n}_e$ aligned with $\boldsymbol{z}_0$

$$ {}^{3'}\boldsymbol{T}_e = \begin{pmatrix} 0 & 0 & 1 & a_l \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

- Position of the wrist center obtained by equating the two branches in the base frame

$$ {}^{3'}\boldsymbol{R}_0(\boldsymbol{q}') \left({}^{0}\boldsymbol{r}_{03'}(\boldsymbol{q}') - {}^{0}\boldsymbol{r}_{01''}(q_{1''}) \right) = \boldsymbol{0} $$

- This leads to the scalar constraints

$$ \begin{aligned} a_w(c_{1'} + c_{1'2'3'}) + a_h(c_{1'2'} - c_{1''}) &= 0 \\ a_w(s_{1'} + s_{1'2'3'}) + a_h(s_{1'2'} - s_{1''}) &= 0 \end{aligned} $$

- Geometric constraints imply simple relations among the joint variables

$$q_{2'} + q_{3'} = \pm\pi$$
$$q_{1'} + q_{2'} = q_{1''}$$

$$\Downarrow$$

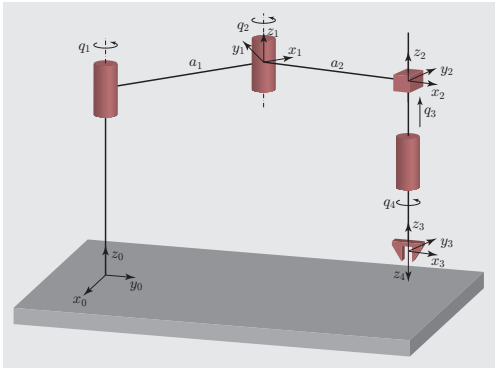
$$q_{2'} = q_{1''} - q_{1'}$$
$$q_{3'} = -q_{2'} \pm \pi = q_{1'} - q_{1''} \pm \pi$$

- Independent joint variables are $\boldsymbol{q} = (q_{1'}, q_{1''})$
- Substituting the expressions of $q_{2'}$ and $q_{3'}$ into the homogeneous transformation ${}^0\boldsymbol{A}_{3'}$ yields the direct kinematics function

$${}^0\boldsymbol{T}_e(\boldsymbol{q}) = {}^0\boldsymbol{A}_{3'}(\boldsymbol{q})\,{}^{3'}\boldsymbol{T}_e = \begin{pmatrix} 0 & -s_{1'} & -c_{1'} & a_h c_{1''} - a_l c_{1'} \\ 0 & c_{1'} & -s_{1'} & a_h s_{1''} - a_l s_{1'} \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

SCARA Manipulator

- Four-DoF RRPR structure with all joint axes vertical and parallel
- Mobility of end-effector corresponds to that of a 3R planar arm plus vertical translation



- Link frames chosen according to DH convention with RRPR sequence

Link	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	0	q_4

- DH homogeneous transformations ${}^0\boldsymbol{A}_1$ and ${}^1\boldsymbol{A}_2$

$$ {}^0\boldsymbol{A}_1(q_1) = \begin{pmatrix} c_1 & -s_1 & 0 & a_1 c_1 \\ s_1 & c_1 & 0 & a_1 s_1 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^1\boldsymbol{A}_2(q_2) = \begin{pmatrix} c_2 & -s_2 & 0 & a_2 c_2 \\ s_2 & c_2 & 0 & a_2 s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

- Prismatic joint and final revolute joint

$$ {}^2\boldsymbol{A}_3(q_3) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^3\boldsymbol{A}_4(q_4) = \begin{pmatrix} c_4 & s_4 & 0 & 0 \\ s_4 & -c_4 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

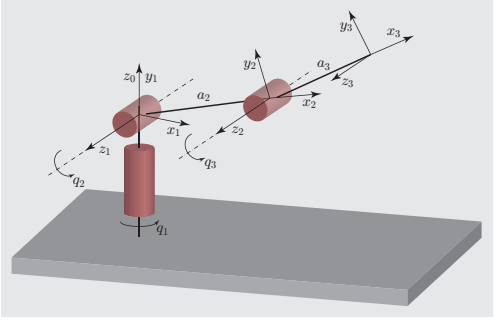
- Direct kinematics obtained as product of the four link transformations

$$ {}^0\boldsymbol{T}_4(\boldsymbol{q}) = {}^0\boldsymbol{A}_1 {}^1\boldsymbol{A}_2 {}^2\boldsymbol{A}_3 {}^3\boldsymbol{A}_4 = \begin{pmatrix} c_{124} & s_{124} & 0 & a_1 c_1 + a_2 c_{12} \\ s_{124} & -c_{124} & 0 & a_1 s_1 + a_2 s_{12} \\ 0 & 0 & -1 & d_1 + q_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

- Task vector can include planar position orientation and vertical displacement of the end-effector

Anthropomorphic Arm

- Three revolute joints forming a shoulder type spherical structure
- Link frames chosen so that z_1 and z_2 are parallel
- Base frame origin at the intersection of z_0 and z_1 with $d_1 = 0$



- DH parameters for the three link frames

Link	α_i	a_i	d_i	θ_i
1	$\pi/2$	0	0	q_1
2	0	a_2	0	q_2
3	0	a_3	0	q_3

- DH homogeneous transformations for the first joint

$$ {}^{0}\boldsymbol{A}_1(q_1) = \begin{pmatrix} c_1 & 0 & s_1 & 0 \\ s_1 & 0 & -c_1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

- For $i = 2, 3$ the transformations have planar structure

$$ {}^{i-1}\boldsymbol{A}_i(q_i) = \begin{pmatrix} c_i & -s_i & 0 & a_i c_i \\ s_i & c_i & 0 & a_i s_i \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

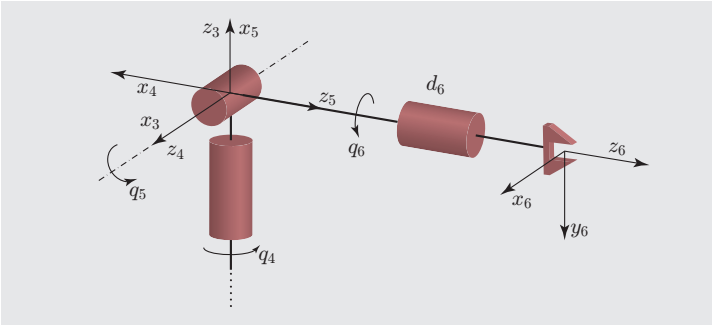
- Direct kinematics to frame 3 obtained as

$$ {}^0\boldsymbol{T}_3(\boldsymbol{q}) = {}^0\boldsymbol{A}_1 {}^1\boldsymbol{A}_2 {}^2\boldsymbol{A}_3 = \begin{pmatrix} c_1c_{23} & -c_1s_{23} & s_1 & c_1(a_2c_2 + a_3c_{23}) \\ s_1c_{23} & -s_1s_{23} & -c_1 & s_1(a_2c_2 + a_3c_{23}) \\ s_{23} & c_{23} & 0 & a_2s_2 + a_3s_{23} \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

- Frame 3 does not generally coincide with the end-effector frame so an additional constant transform ${}^3\boldsymbol{T}_e$ may be needed

Spherical Wrist

- Three revolute joints with concurrent axes at the wrist center
- Joint variables numbered q_4, q_5, q_6 when mounted on a 3R arm
- Axes z_3, z_4, z_5 intersect at a single point yielding spherical motion



- DH parameters for the three link frames

Link	α_i	a_i	d_i	θ_i
4	$-\pi/2$	0	0	q_4
5	$\pi/2$	0	0	q_5
6	0	0	d_6	q_6

- DH homogeneous transformations for the three wrist joints

$$
{}^{3}\boldsymbol{A}_{4}(q_4) = \begin{pmatrix} c_4 & 0 & -s_4 & 0 \\ s_4 & 0 & c_4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{4}\boldsymbol{A}_{5}(q_5) = \begin{pmatrix} c_5 & 0 & s_5 & 0 \\ s_5 & 0 & -c_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

$$
{}^{5}\boldsymbol{A}_{6}(q_6) = \begin{pmatrix} c_6 & -s_6 & 0 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

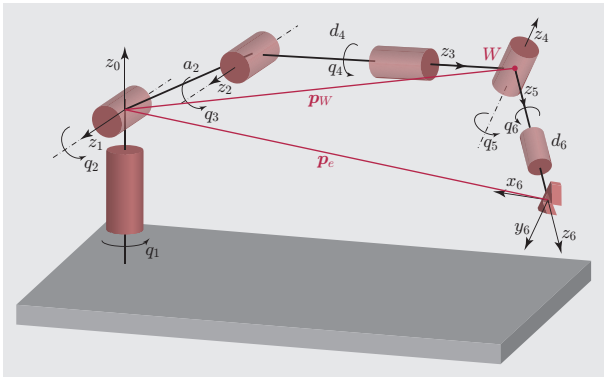
- Composition of the three transformations gives

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

- Rotation block ${}^3\boldsymbol{R}_6$ coincides with the $\boldsymbol{R}_{zyz}$ matrix of Euler angles
- Unit vectors of frame 6 coincide with those of a standard end-effector frame

6R Manipulator with Spherical Wrist

- Combination of anthropomorphic arm and spherical wrist
- Six revolute joints yielding full pose control of the end-effector



- DH parameters for the complete 6R structure

Link	α_i	a_i	d_i	θ_i
1	$\pi/2$	0	0	q_1
2	0	a_2	0	q_2
3	$\pi/2$	0	0	q_3
4	$-\pi/2$	0	d_4	q_4
5	$\pi/2$	0	0	q_5
6	0	0	d_6	q_6

- Rows 3 and 4 of the DH table differ from the separate arm and wrist structures
- Corresponding homogeneous transforms ${}^2\boldsymbol{A}_3$ and ${}^3\boldsymbol{A}_4$ are modified

$$ {}^2\boldsymbol{A}_3(q_3) = \begin{pmatrix} c_3 & 0 & s_3 & 0 \\ s_3 & 0 & -c_3 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^3\boldsymbol{A}_4(q_4) = \begin{pmatrix} c_4 & 0 & -s_4 & 0 \\ s_4 & 0 & c_4 & 0 \\ 0 & -1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

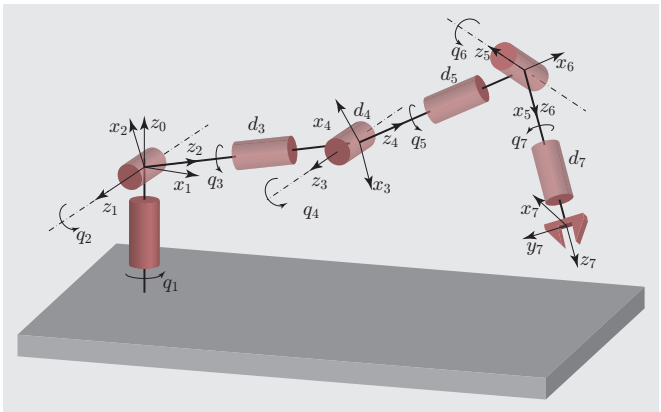
- Assuming the end-effector frame coincides with frame 6 the position vector is

$$ \boldsymbol{p}_e = \begin{pmatrix} a_2c_1c_2 + d_4c_1s_{23} + d_6\left(c_1(c_{23}c_4s_5 + s_{23}c_5) + s_1s_4s_5\right) \\ a_2s_1c_2 + d_4s_1s_{23} + d_6\left(s_1(c_{23}c_4s_5 + s_{23}c_5) - c_1s_4s_5\right) \\ a_2s_2 - d_4c_{23} + d_6(s_{23}c_4s_5 - c_{23}c_5) \end{pmatrix} $$

- Orientation expression is obtained from the rotation part of the full homogeneous transform

DLR Manipulator

- Seven-DoF DLR arm with a kinematic structure similar to a 6R arm plus an extra joint
- Inherently redundant for tasks involving end-effector position and orientation



- DH parameters for the 7R DLR manipulator

Link	α_i	a_i	d_i	θ_i
1	$\pi/2$	0	0	q_1
2	$\pi/2$	0	0	q_2
3	$\pi/2$	0	d_3	q_3
4	$\pi/2$	0	0	q_4
5	$\pi/2$	0	d_5	q_5
6	$\pi/2$	0	0	q_6
7	0	0	d_7	q_7

- Direct kinematics can be computed using the standard DH procedure as for previous manipulators
- A different tool may be mounted so that the last joint axis is orthogonal to the previous one with $d_7 = 0$ and $a_7 \neq 0$
- In that case the approach vector of the end-effector is aligned with x_7 and a constant transformation ${}^7\boldsymbol{T}_e$ is needed to obtain a standard end-effector frame

Kinematic Calibration

- For a given configuration the actual task space variables may deviate from those computed via direct kinematics
  - deviations arise because real link dimensions differ from the DH parameters due to construction and assembly tolerances
- The difference between commanded and reached end-effector pose characterizes the manipulator accuracy, which depends on the end-effector position in the reachable workspace and is critical in programming oriented environments
  - industrial robots typically have positional accuracy of the order of a few millimeters but calibration can improve this significantly
- The DH parameters used in the direct kinematics must be as precise as possible to improve manipulator accuracy
- Kinematic calibration estimates DH parameters from a series of measurements of the end-effector pose and thus direct measurement of the geometric parameters of the structure is avoided

- Calibration is a general issue for any task function $\boldsymbol{y} = \boldsymbol{k}(\boldsymbol{q})$ whose parameters are only approximately known
- Here attention is focused on the end-effector pose in 3D with position $\boldsymbol{p}_e \in I\!R^3$ and orientation $\boldsymbol{\phi}_e \in I\!R^3$
- The direct kinematics is rewritten by emphasizing the dependence on the DH parameters

$$\boldsymbol{y}_e = \boldsymbol{k}_e(\boldsymbol{\alpha}, \boldsymbol{a}, \boldsymbol{d}, \boldsymbol{\theta})$$

  where $\boldsymbol{y}_e \in I\!R^6$ collects position and orientation of the end-effector
- Let $\boldsymbol{y}_m$ be the measured pose and $\boldsymbol{y}_n$ the nominal pose computed from the nominal parameters, and the deviation is $\Delta\boldsymbol{y} = \boldsymbol{y}_m - \boldsymbol{y}_n$

- Assume small perturbations of the nominal DH parameters
- The deviation $\Delta \boldsymbol{y}$ is approximated to first order as

$$\Delta \boldsymbol{y} = \frac{\partial \boldsymbol{k}_e}{\partial \boldsymbol{\alpha}} \Delta \boldsymbol{\alpha} + \frac{\partial \boldsymbol{k}_e}{\partial \boldsymbol{a}} \Delta \boldsymbol{a} + \frac{\partial \boldsymbol{k}_e}{\partial \boldsymbol{d}} \Delta \boldsymbol{d} + \frac{\partial \boldsymbol{k}_e}{\partial \boldsymbol{\theta}} \Delta \boldsymbol{\theta}$$

  where $\Delta \boldsymbol{\alpha}$, $\Delta \boldsymbol{a}$, $\Delta \boldsymbol{d}$, $\Delta \boldsymbol{\theta}$ are constant deviations between real and nominal DH parameters
- Variations on joint variables are due to constant biases on position sensor data (encoders)
- Each partial derivative matrix is $6 \times n$ and relates parameter variations to task space variations

- Group all DH parameters in the $4n$ dimensional vector

$$\boldsymbol{\zeta} = \begin{pmatrix} \boldsymbol{\alpha} \\ \boldsymbol{a} \\ \boldsymbol{d} \\ \boldsymbol{\theta} \end{pmatrix}$$

- Let $\Delta\boldsymbol{\zeta} = \boldsymbol{\zeta} - \boldsymbol{\zeta}_n$ be the variation with respect to the nominal values $\boldsymbol{\zeta}_n$
- Define the $6 \times 4n$ calibration matrix (regressor)

$$\boldsymbol{\Phi} = \begin{pmatrix} \partial \boldsymbol{k}_e/\partial \boldsymbol{\alpha} & \partial \boldsymbol{k}_e/\partial \boldsymbol{a} & \partial \boldsymbol{k}_e/\partial \boldsymbol{d} & \partial \boldsymbol{k}_e/\partial \boldsymbol{\theta} \end{pmatrix}$$

- Evaluating at $\boldsymbol{\zeta}_n$ the linearized relation yields

$$\Delta \boldsymbol{y} = \boldsymbol{\Phi}(\boldsymbol{\zeta}_n) \Delta \boldsymbol{\zeta}$$

- Perform measurements at l different robot configurations
- For configuration i one has a relation of the form $\Delta \boldsymbol{y}_i = \boldsymbol{\Phi}_i \Delta \boldsymbol{\zeta}$
- Stack all deviations and regressors

$$\Delta \bar{\boldsymbol{y}} = \begin{pmatrix} \Delta \boldsymbol{y}_1 \\ \vdots \\ \Delta \boldsymbol{y}_l \end{pmatrix} = \begin{pmatrix} \boldsymbol{\Phi}_1 \\ \vdots \\ \boldsymbol{\Phi}_l \end{pmatrix} \Delta \boldsymbol{\zeta} = \bar{\boldsymbol{\Phi}} \Delta \boldsymbol{\zeta}$$

  where $\bar{\boldsymbol{\Phi}}$ is a $6l \times 4n$ regressor matrix built from the $\boldsymbol{\Phi}_i$
- A well-conditioned $\bar{\boldsymbol{\Phi}}$ requires $6l \gg 4n$ and a good spread of poses in the dextrous workspace

- The stacked linear system is solved in a least-squares sense using the pseudoinverse

$$\Delta\boldsymbol{\zeta} = \bar{\boldsymbol{\Phi}}^{\dagger}\Delta\bar{\boldsymbol{y}} = \left(\bar{\boldsymbol{\Phi}}^{T}\bar{\boldsymbol{\Phi}}\right)^{-1}\bar{\boldsymbol{\Phi}}^{T}\Delta\bar{\boldsymbol{y}}$$

  which provides an estimate of the deviations of all DH parameters from their nominal values
- Constant DH parameters are updated by adding the corresponding components of $\Delta\boldsymbol{\zeta}$
- Deviations of joint variables are added as constant corrections to all joint position measurements

- The original calibration problem is nonlinear because $\boldsymbol{k}_e$ depends nonlinearly on $\boldsymbol{\zeta}$
- The linear model is a first-order approximation around current parameter estimates
- An iterative procedure improves the estimate without new measurements by repeatedly updating the linearization point
- At iteration k the linear relation is written with $\bar{\boldsymbol{\Phi}}^{(k)}$ evaluated at $\boldsymbol{\zeta}^{(k)}$

- Iterative algorithm
  - [initialization] set $k = 0$, $\boldsymbol{\zeta}^{(0)} = \boldsymbol{\zeta}_n$ and collect l pose measurements $\boldsymbol{y}_{m,1}, \ldots, \boldsymbol{y}_{m,l}$
  - [cycle] compute deviations $\Delta \boldsymbol{y}_i^{(k)} = \boldsymbol{y}_{m,i} - \boldsymbol{y}_{e,i}$ for $i = 1, \ldots, l$ where each $\boldsymbol{y}_{e,i}$ is evaluated from $\boldsymbol{y}_e = \boldsymbol{k}_e(\boldsymbol{\alpha}, \boldsymbol{a}, \boldsymbol{d}, \boldsymbol{\theta})$ at $\boldsymbol{\zeta}^{(k)}$
  - build the stacked vector $\Delta \bar{\boldsymbol{y}}^{(k)}$ and the stacked regressor $\bar{\boldsymbol{\Phi}}^{(k)}$ from the current $\boldsymbol{\Phi}_i^{(k)}$
  - compute the parameter correction

$$\Delta \boldsymbol{\zeta}^{(k)} = \left(\bar{\boldsymbol{\Phi}}^{(k)} \right)^{\dagger} \Delta \bar{\boldsymbol{y}}^{(k)}$$

  - update the estimate $\boldsymbol{\zeta}^{(k+1)} = \boldsymbol{\zeta}^{(k)} + \Delta \boldsymbol{\zeta}^{(k)}$
  - [exit] if $\| \Delta \boldsymbol{\zeta}^{(k)} \|$ is below a chosen threshold, then stop and exit with $\boldsymbol{\zeta}^* = \boldsymbol{\zeta}^{(k+1)}$, else increase k and cycle

- Upon convergence the calibration procedure yields an accurate estimate ζ^* for the real manipulator
- Both constant DH parameters and constant corrections to joint measurements are identified
- In practice it may be convenient to calibrate only those DH parameters that are expected to have larger uncertainty
- Keeping the remaining parameters at nominal values often improves the conditioning of the regressor matrix $\bar{\mathbf{\Phi}}$

- Besides accuracy an important precision parameter is the repeatability of a robot manipulator
  - repeatability measures the ability to return to a previously taught position in the reachable workspace
  - it is always listed in industrial robot data sheets and is relevant for teaching by showing programming
  - repeatability mainly depends on the quality of mechanical structure, sensory equipment and control hardware
  - it is a manufacturer feature common to all manipulators of the same family
- Repeatability is typically better than calibrated accuracy with smaller position variations when returning to the same pose

Reachable Workspace

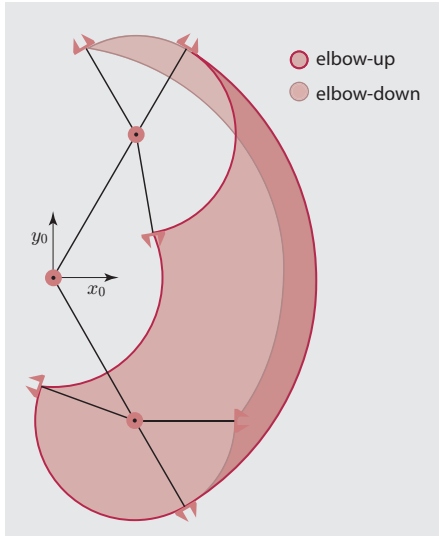
- An important feature of a robot manipulator is its reachable workspace $\mathcal{W}_r$
- $\mathcal{W}_r$ is the subset of the workspace $\mathcal{W}$ where it is possible to position the end-effector origin with at least one orientation
- Only the position of the origin of the end-effector frame is considered when defining $\mathcal{W}_r$

- The reachable workspace is determined by manipulator geometry and mechanical joint limits
- It can be built as the volume swept by the end-effector frame origin when the joints execute all admissible motions

$$\mathcal{W}_r = \{\boldsymbol{p} : \boldsymbol{p} = \boldsymbol{k}_p(\boldsymbol{q}), q_{im} \leq q_i \leq q_{iM}, i = 1, \ldots, n\}$$

  - $\boldsymbol{k}_p$ is the positional part of the direct kinematic map and q_{im}, q_{iM} are the joint limits
  - $\mathcal{W}_r$ is a bounded region of the workspace $\mathcal{W}$
- It is usually characterized by its boundary rather than by its interior volume
- Since joints are either revolute or prismatic the boundary consists of patches of planar, spherical, toroidal and cylindrical surfaces

- Consider the reachable workspace of a 2R planar arm with joint limits
- Two regions correspond to configurations with positive and negative values of q_2
- Their intersection identifies positions that can be reached with both elbow-up and elbow-down configurations



- The reachable workspace of a robot manipulator is typically shown graphically in the manufacturer data sheet
  - it is often represented by a top view and a side view of the region $\mathcal{W}_r$
  - these plots are important to evaluate whether the manipulator is suitable for a given application
- A refinement of the reachable workspace concept is the dexterous workspace
  - the dexterous workspace is the subset of $\mathcal{W}_r$ where the end-effector can reach each point with all admissible orientations
  - the dexterous workspace is in general strictly contained in the reachable workspace
  - the dexterous workspace of a spatial manipulator with less than six DoFs is empty

Inverse Kinematics

- Direct kinematics $\boldsymbol{y} = \boldsymbol{k}(\boldsymbol{q})$ maps generalized coordinates $\boldsymbol{q}$ to task coordinates $\boldsymbol{y}$
- The inverse kinematics problem consists in finding configurations $\boldsymbol{q}$ that realize a desired task vector $\boldsymbol{y}_d$
- It is essential for converting task space specifications into joint space commands
- The equations to solve are nonlinear and closed-form solutions are not always available
- For nonredundant robots with $m = n$ multiple solutions typically exist for the same task
- For redundant robots with $m < n$ infinite solutions exist and extra criteria are needed to select one configuration
- The number of inverse kinematics solutions may be different (none, only one, many, infinite), depending on the input data (i.e., the desired value of the task coordinates)
- Joint limits, obstacles and singular configurations may make some desired task values unattainable

- Closed-form inverse kinematics can be derived only for special kinematic structures
- When available, analytical solutions provide all admissible configurations satisfying a given task
- Computation of closed-form solutions requires either algebraic intuition to find the significant equations containing the unknowns, or geometric intuition to find the significant points in the structure allowing expressing position and/or orientation as a function of a reduced number of unknowns
- Otherwise one resorts to numerical schemes based on solving $\boldsymbol{y} = \boldsymbol{k}(\boldsymbol{q})$ or its differential counterpart
- In the following we present closed-form solutions for some important manipulators

3R Planar Arm

- Consider the 3R planar arm whose direct kinematics has the form

$$\begin{pmatrix} p_x \\ p_y \\ \phi \end{pmatrix} = \begin{pmatrix} a_1 c_1 + a_2 c_{12} + a_3 c_{123} \\ a_1 s_1 + a_2 s_{12} + a_3 s_{123} \\ q_1 + q_2 + q_3 \end{pmatrix}$$

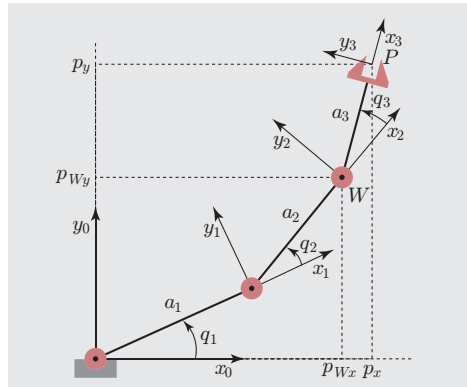
- The task is to find q_1, q_2, q_3 that realize a desired pose (p_x, p_y, ϕ)
- Several solutions may exist corresponding to different elbow configurations

- Introduce the wrist center W at distance a_3 from the end-effector along direction ϕ

$$p_{Wx} = p_x - a_3 c_\phi = a_1 c_1 + a_2 c_{12}$$

$$p_{Wy} = p_y - a_3 s_\phi = a_1 s_1 + a_2 s_{12}$$

- The position of W depends only on the first two joint variables q_1 and q_2



- Squaring and summing the equations for p_{Wx} and p_{Wy} gives

$$p_{Wx}^2 + p_{Wy}^2 = a_1^2 + a_2^2 + 2a_1 a_2 c_2$$

- Hence

$$c_2 = \frac{p_{Wx}^2 + p_{Wy}^2 - a_1^2 - a_2^2}{2a_1 a_2}$$

$$s_2 = \pm\sqrt{1 - c_2^2}$$

- Existence of a solution requires $-1 \leq c_2 \leq 1$ otherwise W is outside the reachable workspace of the first two links
- The two signs of s_2 correspond to elbow-down and elbow-up configurations

- The joint variable q_2 is obtained from

$$q_2 = \text{atan2}(s_2, c_2)$$

- Substituting q_2 into the equations for p_{Wx} and p_{Wy} yields a system in s_1 and c_1 with solution

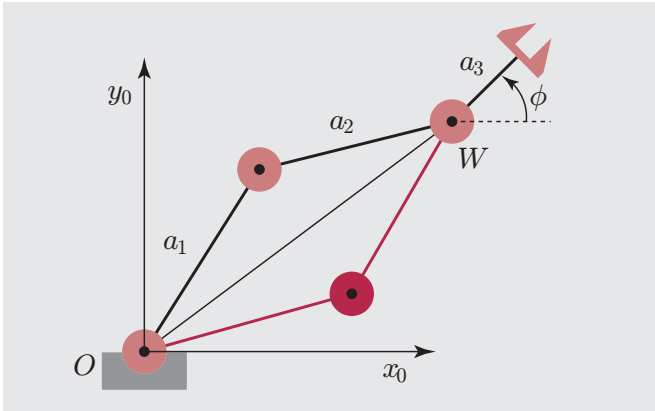
$$s_1 = \frac{(a_1 + a_2 c_2) p_{Wy} - a_2 s_2 p_{Wx}}{p_{Wx}^2 + p_{Wy}^2}$$

$$c_1 = \frac{(a_1 + a_2 c_2) p_{Wx} + a_2 s_2 p_{Wy}}{p_{Wx}^2 + p_{Wy}^2}$$

$$q_1 = \text{atan2}(s_1, c_1) \qquad q_3 = \phi - (q_1 + q_2)$$

- Configurations with $s_2 = 0$ correspond to kinematic singularities of the planar arm

- Admissible postures of a 3R planar arm



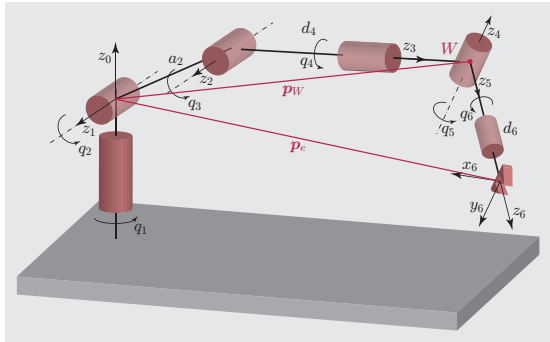
Manipulators with Spherical Wrist

- Many manipulators are composed of a positioning arm and a spherical wrist
- The first three joints determine the position of the wrist center
- The last three joints determine the orientation of the end-effector
- This structure decouples position and orientation in the inverse kinematics
- A 6-DoF 6R manipulator admits a closed-form inverse kinematics if three consecutive revolute joint axes intersect at a common point as in a spherical wrist or shoulder
- A closed-form solution also exists when three consecutive revolute joint axes are parallel
- Otherwise one typically resorts to numerical or approximate methods
- In all cases some algebraic or geometric intuition is needed to derive closed-form solutions

- For manipulators with spherical wrist the inverse kinematics is solved in two steps

$$\boldsymbol{p}_W = \boldsymbol{p}_6 - d_6 \boldsymbol{z}_6$$

- First solve the inverse position kinematics of the arm to obtain the wrist center position
- Then solve the inverse orientation kinematics of the spherical wrist to match the desired end-effector orientation
- This decoupling can be exploited for both planar and spatial manipulators



Spherical Wrist

- Consider the spherical wrist whose direct kinematics yields the rotation matrix ${}^3\boldsymbol{R}_6$
- The components of ${}^3\boldsymbol{R}_6$ can be written as

$$ {}^3\boldsymbol{R}_6 = \begin{pmatrix} {}^3n_x & {}^3s_x & {}^3a_x \\ {}^3n_y & {}^3s_y & {}^3a_y \\ {}^3n_z & {}^3s_z & {}^3a_z \end{pmatrix} $$

- The goal is to find q_4, q_5, q_6 that yield a desired matrix ${}^3\boldsymbol{R}_6$

- From the expression of ${}^3\boldsymbol{R}_6$ in terms of q_4, q_5, q_6 one obtains two sets of solutions
- For $q_5 \in (0, \pi)$

$$q_4 = \text{atan2}({}^3a_y, {}^3a_x)$$

$$q_5 = \text{atan2}\left(\sqrt{({}^3a_x)^2 + ({}^3a_y)^2}, {}^3a_z\right)$$

$$q_6 = \text{atan2}({}^3s_z, -{}^3n_z)$$

- For $q_5 \in (-\pi, 0)$

$$q_4 = \text{atan2}(-{}^3a_y, -{}^3a_x)$$

$$q_5 = \text{atan2}\left(-\sqrt{({}^3a_x)^2 + ({}^3a_y)^2}, {}^3a_z\right)$$

$$q_6 = \text{atan2}(-{}^3s_z, {}^3n_z)$$

- These correspond to the two ZYZ Euler angles solutions associated with ${}^3\boldsymbol{R}_6$

Anthropomorphic Arm

- Consider the 3R spatial anthropomorphic arm and denote by $\boldsymbol{p}$ the position of frame 3
- The direct kinematics for $\boldsymbol{p}$ is given by

$$p_x = c_1(a_2c_2 + a_3c_{23})$$

$$p_y = s_1(a_2c_2 + a_3c_{23})$$

$$p_z = a_2s_2 + a_3s_{23}$$

- The task is to find q_1, q_2, q_3 that realize a desired position $\boldsymbol{p}$

- Squaring and summing the first two equations yields

$$p_x^2 + p_y^2 = (a_2 c_2 + a_3 c_{23})^2$$

- Assuming $p_x^2 + p_y^2 > 0$, the common factor can be eliminated and two solutions for q_1 are obtained

$$q_1^f = \text{atan2}(p_y, p_x) \qquad q_1^b = \text{atan2}(-p_y, -p_x)$$

- The superscripts f and b denote configurations in which the arm is facing or backing the desired position $\boldsymbol{p}$

- Squaring and summing all three kinematic equations gives

$$\|\boldsymbol{p}\|^2 = p_x^2 + p_y^2 + p_z^2 = a_2^2 + a_3^2 + 2a_2a_3c_3$$

- Hence

$$c_3 = \frac{p_x^2 + p_y^2 + p_z^2 - a_2^2 - a_3^2}{2a_2a_3} \quad \text{and} \quad s_3 = \pm\sqrt{1 - c_3^2}$$

- Admissibility requires $|c_3| \leq 1$ or equivalently $|a_2 - a_3| \leq \|\boldsymbol{p}\| \leq a_2 + a_3$
- From c_3 and s_3 the joint variable q_3 follows as

$$q_3 = \operatorname{atan2}(s_3, c_3)$$

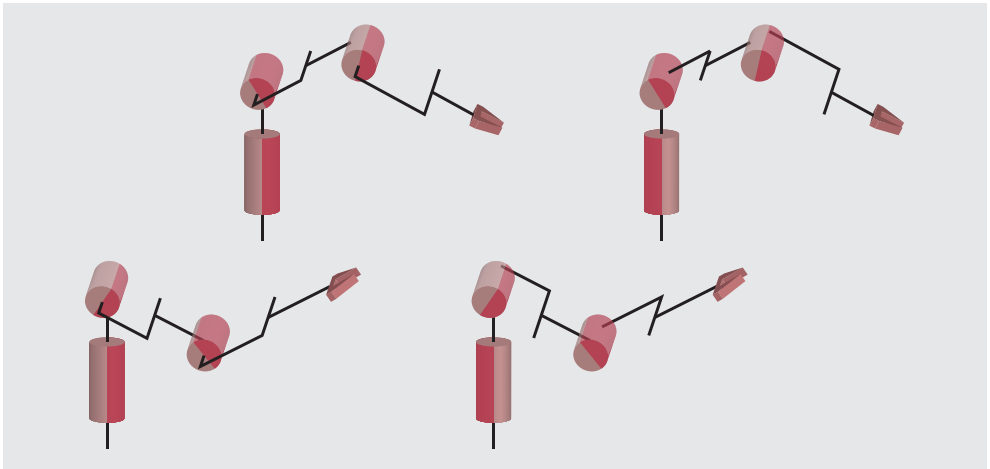
- Two solutions q_3^+ and $q_3^- = -q_3^+$ correspond to elbow-up and elbow-down configurations

- Using the known values of q_1 and q_3, the equations for $\boldsymbol{p}$ can be rearranged as a linear system in c_2 and s_2

$$\begin{pmatrix} c_1 p_x + s_1 p_y \\ p_z \end{pmatrix} = \begin{pmatrix} a_2 c_2 + a_3 c_{23} \\ a_2 s_2 + a_3 s_{23} \end{pmatrix} = \begin{pmatrix} a_2 + a_3 c_3 & -a_3 s_3 \\ a_3 s_3 & a_2 + a_3 c_3 \end{pmatrix} \begin{pmatrix} c_2 \\ s_2 \end{pmatrix}$$

- Solving the linear system yields c_2 and s_2 and then

$$q_2 = \operatorname{atan2}(s_2, c_2)$$

- Combining the two solutions for q_1 with the two solutions for q_3 yields four distinct postures
- These can be classified as shoulder-right and shoulder-left combined with elbow-up and elbow-down
- When $p_x^2 + p_y^2 = 0$ the desired position lies on the base axis and the arm is in a kinematic singular configuration
- In singular cases some joint variables may lose local influence on the end-effector motion

- Admissible postures of an anthropomorphic arm