Introduction to Robotics

Julien Carpentier

Class notes by Antoine Groudiev

ENS | PSL★

Contents

1   Introduction

2   Position and Orientation

2.1   Introduction

Kinematics studies the movement of an object – in our case of a robot – without taking into acount the forces generating it. Instead, it only handles aspects such as position, orientation, speed and momentum of bodies in movement.

Consider for instance a robotic arm. We can design a simplified scheme of the robot and its environment, to create a kinematic pipeline and reference frames associated to each of these objects.
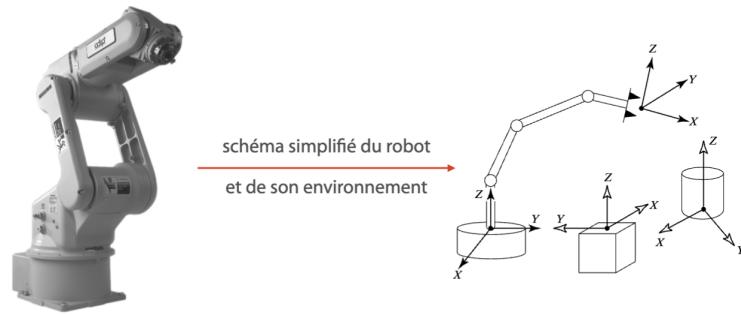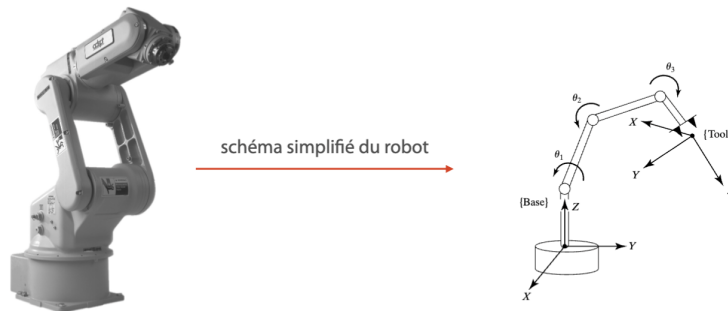

Figure 2.1: Simplified scheme of the robot feature the kinematic pipeline and reference frames.

Direct kinematics allows to compute the position and orientation of the terminal organ given, for instance, the angles of the articulations.

Invert kinematics answers the question the other way around: given the position and orientation of a boyd, how can we compute the values of the articulations angles. Invert kinematics is used for instance for trajectory tracking: given a reference trajectory, how can we compute the speed of the articulations?

2.2   Points, frames and transformations

2.2.1   Position of a point in space

Once that a reference frame $\{A\}$ is defined, we can localize any point of the universe given a *position vector*:
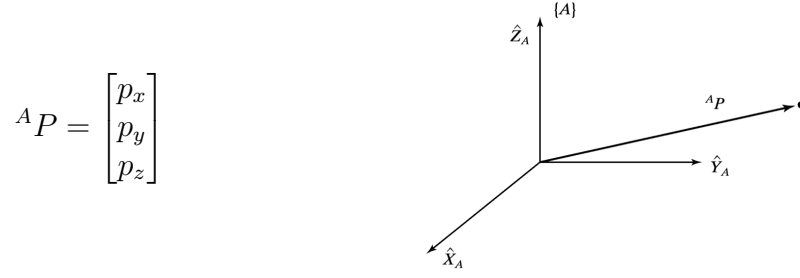
$$^AP = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}$$

Figure 2.2: Vector and position of the point established in the frame $\{A\}$.

2.2.2 Position and orientation of a body in space

To define the orientation of a body in space, we need to define a reference frame $\{B\}$ attached to this body. The orientation is therefore defined as the expression of this coordinate system in the reference frame $\{A\}$.

$$^A_B R = \begin{bmatrix} ^A\hat{X}_B & ^A\hat{Y}_B & ^A\hat{Z}_B \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$
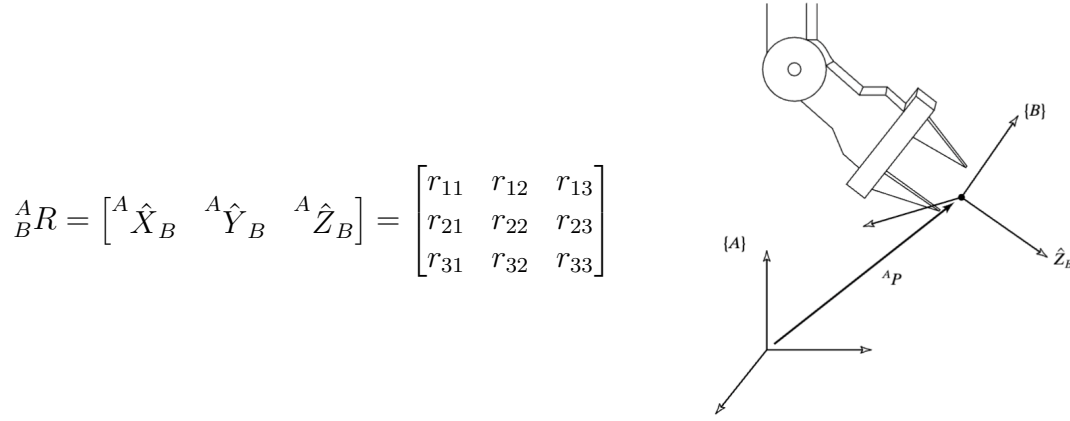

Figure 2.3: Expression of the coordinate system $\{B\}$ in the reference frame $\{A\}$, and the associated position and orientation of the body.

Each element of the matrix $^A_B R$ is the scalar product between the vectors of the two coordinate systems $\{B\}$ and $\{A\}$:

$$^A_B R = \begin{bmatrix} ^A\hat{X}_B & ^A\hat{Y}_B & ^A\hat{Z}_B \end{bmatrix} = \begin{bmatrix} \hat{X}_B \cdot \hat{X}_A & \hat{Y}_B \cdot \hat{X}_A & \hat{Z}_B \cdot \hat{X}_A \\ \hat{X}_B \cdot \hat{Y}_A & \hat{Y}_B \cdot \hat{Y}_A & \hat{Z}_B \cdot \hat{Y}_A \\ \hat{X}_B \cdot \hat{Z}_A & \hat{Y}_B \cdot \hat{Z}_A & \hat{Z}_B \cdot \hat{Z}_A \end{bmatrix}$$

The lines correspond to the axes of the frame $\{A\}$ expressed in the frame $\{B\}$. Note that the invert of a rotation matrix is its transpose:

$$^A_B R^T = ^A_B R^{-1} = ^B_A R$$

2.2.3 Rotation matrices

We showed that the orientation of the body in space could be expressed as a 3-dimensional rotation matrix. The group of 3-dimensional rotation matrices is denoted SO(3), and is composed of all the matrices that are orthonormal, that is orthogonal and with a determinant equal to 1:

$$\text{SO}(3) = \left\{ R \in \mathcal{M}_3(\mathbb{R}) \,\middle|\, RR^T = I_3 \text{ and } \det(R) = +1 \right\} \tag{2.2.1}$$

3

If we write:

$$R = \begin{bmatrix} \hat{X} & \hat{Y} & \hat{Z} \end{bmatrix}$$

Then $R \in \mathrm{SO}(3)$ if and only if:

$$\begin{cases} \hat{X} \cdot \hat{Y} = 0 \\ \hat{Y} \cdot \hat{Z} = 0 \\ \hat{Z} \cdot \hat{X} = 0 \end{cases} \quad \text{and} \quad \begin{cases} \hat{X} \cdot \hat{X} = 1 \\ \hat{Y} \cdot \hat{Y} = 1 \\ \hat{Z} \cdot \hat{Z} = 1 \end{cases}$$

Note that we have 9 degrees of freedom and 6 independent constraints, so the group $\mathrm{SO}(3)$ is 3-dimensional.

2.3 Rotation representations

There are several ways to represent a rotation matrix, each with its own advantages and drawbacks. The most common representations are:

- Orthonormal 3 by 3 matrices — 9 components
- Euler angles — 3 components
- Axis-angle representation — 3 components
- Quaternions — 4 components

The number of components used to be an important factor when computers were slow and memory was expensive. Nowadays, the choice of representation is more about the ease of use and the properties of the representation.

2.3.1 Euler angles

Each rotation can be represented by the composition of three elementary rotations around the fixed axes of the frame $\{A\}$.
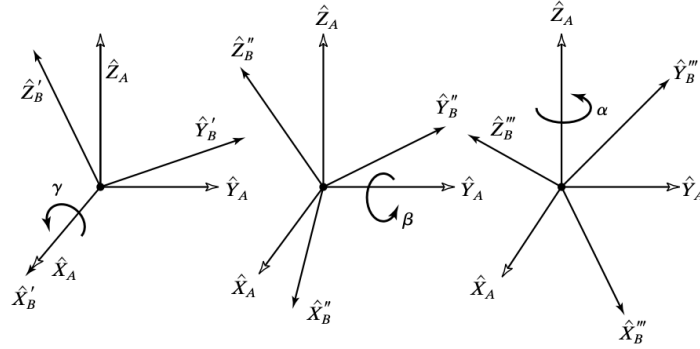

Figure 2.4: Euler angles representation of a rotation.

In terms of rotation matrices, any rotation matrix can be written as ${}^A_B R(\gamma, \beta, \alpha)$, defined by:

$$\begin{aligned} {}^A_B R(\gamma, \beta, \alpha) &= R_Z(\alpha) R_Y(\beta) R_X(\gamma) \\ &= \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{bmatrix} \end{aligned}$$

The Euler angles are not unique, as the same rotation can be represented by different sets of Euler angles. This is called gimbal lock, and is a major drawback of the Euler angles representation.

We can also express the Euler angles given the rotation matrix:

$$
{}_B^A R(\gamma, \beta, \alpha) = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \implies \begin{cases} \beta = \text{atan2}(-r_{31}, \sqrt{r_{11}^2 + r_{21}^2}) \\ \alpha = \text{atan2}(r_{21}/\cos\beta, r_{11}/\cos\beta) \\ \gamma = \text{atan2}(r_{32}/\cos\beta, r_{33}/\cos\beta) \end{cases}
$$

2.3.2 Axis-angle and quaternions

We are given a vector $\vec{k}$ and an angle θ; the rotation represented by these two elements is the rotation of angle θ around the axis $\vec{k}$. We can define:

$$
\begin{cases} \varepsilon_1 = k_x \sin\frac{\theta}{2} \\ \varepsilon_2 = k_y \sin\frac{\theta}{2} \\ \varepsilon_3 = k_z \sin\frac{\theta}{2} \\ \varepsilon_4 = \cos\frac{\theta}{2} \end{cases}
$$

we have $\varepsilon_1^2 + \varepsilon_2^2 + \varepsilon_3^2 + \varepsilon_4^2 = 1$, creating a unit quaternion. The rotation matrix associated to the quaternion is:

$$
R_\varepsilon = \begin{bmatrix} 1 - 2\varepsilon_2^2 - 2\varepsilon_3^2 & 2(\varepsilon_1\varepsilon_2 - \varepsilon_3\varepsilon_4) & 2(\varepsilon_1\varepsilon_3 + \varepsilon_2\varepsilon_4) \\ 2(\varepsilon_1\varepsilon_2 + \varepsilon_3\varepsilon_4) & 1 - 2\varepsilon_1^2 - 2\varepsilon_3^2 & 2(\varepsilon_2\varepsilon_3 - \varepsilon_1\varepsilon_4) \\ 2(\varepsilon_1\varepsilon_3 - \varepsilon_2\varepsilon_4) & 2(\varepsilon_2\varepsilon_3 + \varepsilon_1\varepsilon_4) & 1 - 2\varepsilon_1^2 - 2\varepsilon_2^2 \end{bmatrix}
$$

The invert operation is also simple to compute:

$$
{}_B^A R(\gamma, \beta, \alpha) = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \implies \begin{cases} \varepsilon_1 = \frac{r_{32} - r_{23}}{4\varepsilon_4} \\ \varepsilon_2 = \frac{r_{13} - r_{31}}{4\varepsilon_4} \\ \varepsilon_3 = \frac{r_{21} - r_{12}}{4\varepsilon_4} \\ \varepsilon_4 = \frac{1}{2}\sqrt{1 + r_{11} + r_{22} + r_{33}} \end{cases}
$$

2.4 Angular velocity

The angular velocity of a body is a vector that describes the rotation of the body in space. It is defined as the derivative of the rotation matrix with respect to time. Angular velocity matrices are skew-symmetric matrices, i.e. matrices of the tangent space of SO(3), denoted so(3):

$$
\text{so}(3) = \left\{ W \in \mathcal{M}_3(\mathbb{R}) \mid W^\top = -W \right\} = \left\{ \begin{bmatrix} 0 & -w_z & w_y \\ w_z & 0 & -w_x \\ -w_y & w_x & 0 \end{bmatrix} \middle| w_x, w_y, w_z \in \mathbb{R} \right\}
$$

Indeed, consider the rotation matrix $R(t) \in \text{SO}(3)$, and denote $\dot{R}(t) = \frac{\partial R(t)}{\partial t}$ the derivative of $R(t)$ with respect to time. We have:

$$
R(t)R(t)^\top = I_3
$$
$$
\dot{R}R^\top + R\dot{R}^\top = 0_3
$$
$$
\dot{R}R^\top = -R\dot{R}^\top
$$
$$
\dot{R}R^\top = -\left(\dot{R}R^\top\right)^\top
$$

Hence, by defining $W := \dot{R}R^\top$, we have $W \in \text{so}(3)$.

Note that $R(t)$ is the solution of the differential equation $\dot{R} = WR$, with $R(0) = R_0 \in \mathrm{SO}(3)$. The solution is:

$$R(t) = R_0 \exp(Wt) \quad \text{where} \quad \exp(Wt) = \sum_{n=0}^{+\infty} \frac{(Wt)^n}{n!}$$

This gives us a new parameterization of the group $\mathrm{SO}(3)$, using the angular velocity matrices. Recall that a matrix $W \in \mathrm{so}(3)$ is associated to a vector $w = (w_x, w_y, w_z) \in \mathbb{R}^3$; we denote $\hat{w} = W$. The main question associated to this representation is to compute the infinite sum of the exponential function:

$$\exp(tW) = \sum_{n=0}^{+\infty} \frac{(tW)^n}{n!} \tag{2.4.1}$$

We can use the fact that any skew-symmetric matrix $W \in \mathrm{so}(3)$ is nilpotent, i.e. there exists an integer $n \in \mathbb{N}$ such that $W^n = 0_3$. This allows us to compute the exponential function:

$$\hat{w} = W = \begin{bmatrix} 0 & -w_z & w_y \\ w_z & 0 & -w_x \\ -w_y & w_x & 0 \end{bmatrix} = w \times \cdot$$

Therefore, $\hat{w}^2 = ww^\top - I_3$, and $\hat{w}^3 = -\hat{w}$. We can then regroup the different terms of the exponential function, by transforming the exponents greater than 3:

$$\exp(tW) = I_3 + \sin(t)\hat{w} + \frac{1 - \cos(t)}{t}\hat{w}^2 \tag{2.4.2}$$

2.5 Exponential and logarithm map

We saw that the exponential map is an application from the tangent space of $\mathrm{SO}(3)$, $\mathrm{so}(3)$ to the group $\mathrm{SO}(3)$, defined by:

$$\exp : \mathrm{so}(3) \longrightarrow \mathrm{SO}(3)$$

$$tW \longmapsto \exp(tW) = I_3 + \sin(t)\hat{w} + \frac{1 - \cos(t)}{t}\hat{w}^2$$

This function is surjective and 2π-periodic.

We can define a reciprocal function, the logarithm map, that is the inverse of the exponential map:

$$\log : R \in \mathrm{SO}(3) \longrightarrow w \in \mathrm{so}(3)$$

Note that

$$\|w\| = \cos^{-1}\left(\frac{\mathrm{Tr}(R) - 1}{2}\right)$$

and therefore:

$$w = \frac{\|w\|}{2\sin(\|w\|)} \begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix}$$

This gives us a way to compute the distance between two rotations:

$$d(R_1, R_2) = \|\log(R_1^\top R_2)\|$$

Such a construction comes handy when we want to minimize the distance between two rotations, for instance in the context of trajectory tracking.

3   Forward Kinematics

Forward kinematics allows to compute the position and orientation of the kinematic chain given the joint parameters (e.g. angles, lengths, etc.). For instance, we can compute the position and orientation of the terminal organ given the angles of the articulations.

schéma simplifié du robot
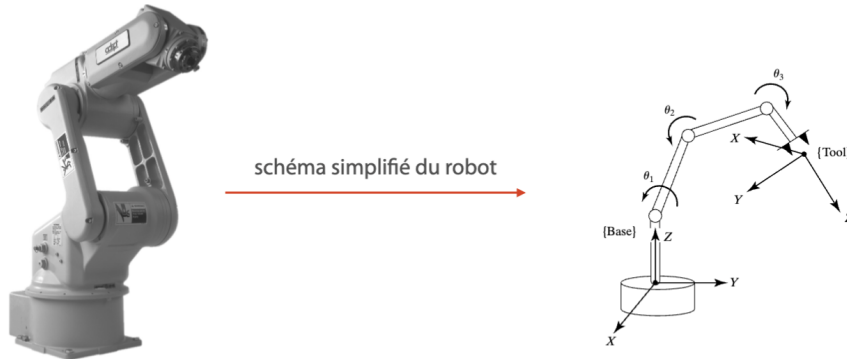
The kinematic chain is composed of rigid bodies interconnect by joints. The joins define the degrees of freedom of the cinematic chain, which are the parameters that we can control.
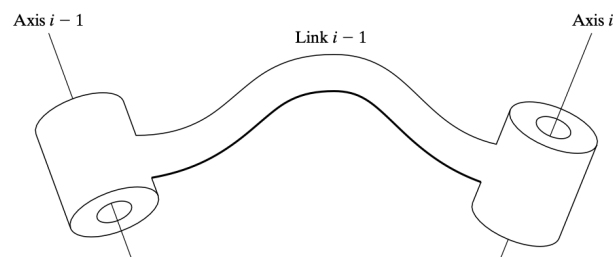
4   Inverse Kinematics

5   Reinforcement Learning

6   Locomotion