Euler angles

The **Euler angles** are three angles introduced to describe the **orientation of rigid body** with respect to a **fixed coordinate system**.[5] Any orientation can be achieved by composing three elemental rotations about the axes of a coordinate system. To be specific, rotate the angles of γ, β and α along the X, Y and Z axes from the reference frame, respectively then you will achieve the orientation. In the figure 5, the reference frame is shown in blue and the rotated orientation is shown in red. The angles γ, β and α can also be called **RPY (roll, pitch and yaw)**.

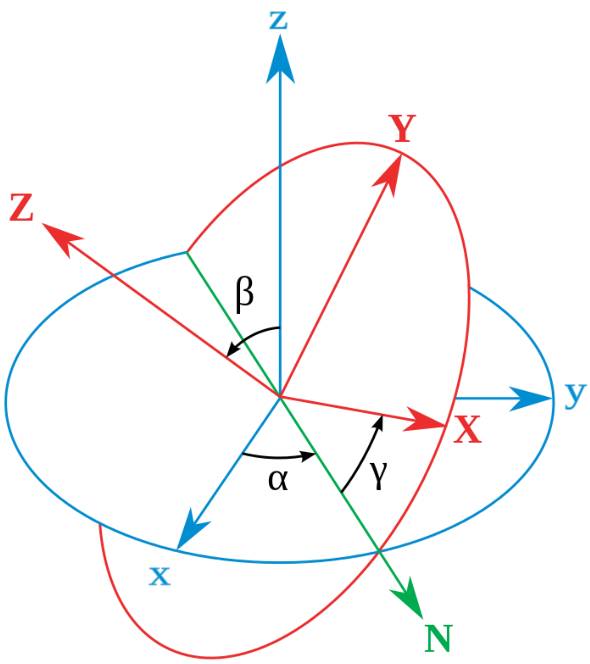

Figure 12 Proper Euler angles geometrical definition

The Euler angles are used to understand the robot orientation, but it should be converted to the rotation matrix for calculations such as kinematics and dynamics. Given Euler angles γ, β and α, the rotation matrix can be calculated by the following equations.[6]

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}$$

$$R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}$$

$$R_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

---

[5] *Euler angles*, Wikipedia (https://en.wikipedia.org/wiki/Euler_angles)

[6] *Planning Algorithms*, Steven M. LaValle (http://planning.cs.uiuc.edu/node103.html)

$$
\begin{aligned}
R &= R_z(\alpha)R_y(\beta)R_x(\gamma) \\
&= \begin{bmatrix} \cos\alpha\cos\beta & \cos\alpha\sin\beta\sin\gamma - \sin\alpha\cos\gamma & \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\gamma \\ \sin\alpha\cos\beta & \sin\alpha\sin\beta\sin\gamma + \cos\alpha\cos\gamma & \sin\alpha\sin\beta\cos\gamma - \cos\alpha\sin\gamma \\ -\sin\beta & \cos\beta\sin\gamma & \cos\beta\cos\gamma \end{bmatrix} \\
&= \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}
\end{aligned}
$$

A given rotation matrix R, the Euler angles are determined as below.

$$
\begin{aligned}
\beta &= \operatorname{atan2}\left(-r_{31}, \sqrt{{r_{11}}^2 + {r_{21}}^2}\right) \\
\alpha &= \operatorname{atan2}\left(\frac{r_{21}}{\cos\beta}, \frac{r_{11}}{\cos\beta}\right) \\
\gamma &= \operatorname{atan2}\left(\frac{r_{32}}{\cos\beta}, \frac{r_{33}}{\cos\beta}\right)
\end{aligned}
$$

The Euler angles are preferred because only three values are needed and geometrically easy to understand compared to other methods. However, there are two critical disadvantages in this representation. The values are not continuous so that calculations with the Euler angles are **complicated**. In addition, the final orientation depends on the sequence of rotations. In other words, even with the same amounts of roll, pitch and yaw, the orientations will be different whether it is rotated with roll first or yaw first as shown in the figure 6, which is **confusing**. In UR robots, RPY values can be monitored in the Move tab of teach pendant. Also, URScript functions *rpy2rotvec* and *rotvec2rpy* are available to support the Euler angles.

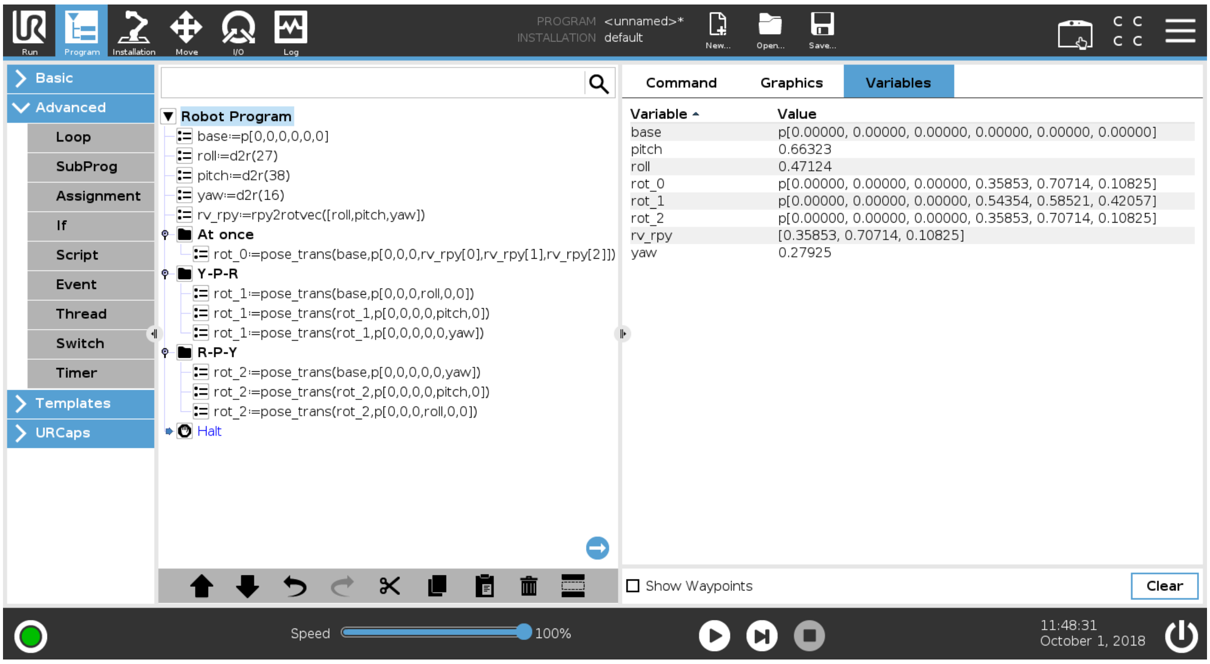

Figure 13 Orientations depends on the sequence of rotations

Rotation vector

The **axis-angle representation** parameterizes a rotation in the linear space by a unit vector u indicating the **direction of an axis of rotation**, and an angle θ describing the **magnitude of the rotation** about the axis.[7] The axis-angle representation can define the relationship between two orientations. For example, n figure 7, the blue orientation can be rotated by the angle θ along the direction vector u to the red orientation.

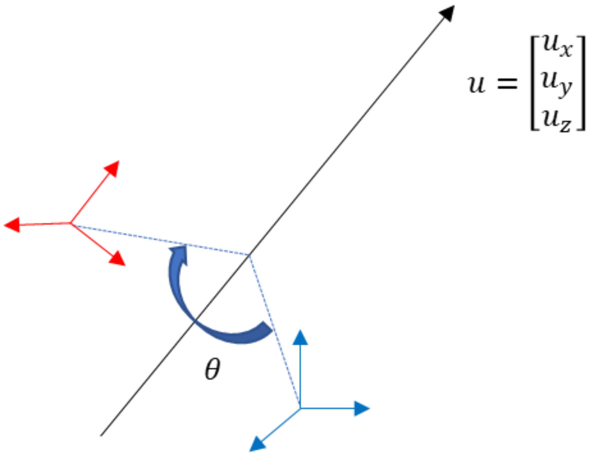

Figure 14 Axis-angle representation

If we have a fixed reference frame such as the base frame in UR robots, we can represent any orientation by four numbers: three elements for the unit direction vector and an angle. If we multiply the angle to each element of the unit direction vector, we can reduce to the three numerical values, which we can call the **rotation vector**.

$$r = \begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} = \begin{bmatrix} \theta u_x \\ \theta u_y \\ \theta u_z \end{bmatrix}$$

The axis-angle representation is useful in the robot control because it is free from continuity and rotational sequence issues, the limitations of the Euler angles. However, it is hard to match between the physical orientation and the numerical values of the rotation vectors. In UR robots, we use the rotation vector to represent the orientation of the robot pose.

To convert to the rotation matrix, the rotation vector should be divided into the angle and unit direction vector.

[7] Axis-angle representation, Wikipedia (https://en.wikipedia.org/wiki/Axis%E2%80%93angle_representation)

$$\theta = \sqrt{{r_x}^2 + {r_y}^2 + {r_z}^2}$$
$$u = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} = \begin{bmatrix} r_x/\theta \\ r_y/\theta \\ r_z/\theta \end{bmatrix}$$

Then, further calculations are needed as follows.

$$\mathrm{c} = \cos\theta$$
$$\mathrm{s} = \sin\theta$$
$$\mathrm{C} = 1 - \cos\theta$$

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} u_x u_x C + c & u_x u_y C - u_z s & u_x u_z C + u_y s \\ u_y u_x C + u_z s & u_y u_y C + c & u_y u_z C - u_x s \\ u_z u_x C - u_y s & u_z u_y C + u_x s & u_z u_z C + c \end{bmatrix}$$

Given a rotation matrix, the rotation vector can be calculated as follows.

$$\theta = \mathrm{acos}\left(\frac{\mathrm{Trace}(R) - 1}{2}\right)$$
$$u = \frac{1}{2\sin\theta}\begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix}$$