Robotics Assignment #03

Paul Bienkowski, Konstantin Kobs

18. Mai 2015

Task 3.1. The gripper should rotate positively around the z-axis and simultaneously translate upwards in z-direction. In fact, it should move h upwards and rotate a given times, that is determined by the thread hight. We will call this variable *rotations*. Both motions have to been done in a linear motion, because we assume, that the angular velocity is constant. This way, we can't use non-linear velocity for the movement in z-direction. To calculate the constant translation velocity v_z, we assume, that $t \in [0, 1]$. This leads to $v_z = h$. w_z needs to be *rotations* $\cdot\ 360°$, because we want to rotate *rotations* times in the given time span from 0 to 1.

This gives us two functions, one to calculate the degrees, that were rotated until time step t ($degr(t)$) and the movement, that was made until this time step ($upw(t)$):

$$degr(t) = rotations \cdot 360° \cdot t$$
$$upw(t) = h \cdot t$$

Then, the overall transformation is given by

$$
\begin{aligned}
T(t) &= Rot_z(t) \cdot Trans_z(t) \\
&= \begin{pmatrix} \cos(degr(t)) & -\sin(degr(t)) & 0 & 0 \\ \sin(degr(t)) & \cos(degr(t)) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & upw(t) \\ 0 & 0 & 0 & 1 \end{pmatrix} \\
&= \begin{pmatrix} \cos(degr(t)) & -\sin(degr(t)) & 0 & 0 \\ \sin(degr(t)) & \cos(degr(t)) & 0 & 0 \\ 0 & 0 & 1 & upw(t) \\ 0 & 0 & 0 & 1 \end{pmatrix} \\
&= \begin{pmatrix} \cos(rotations \cdot 360° \cdot t) & -\sin(rotations \cdot 360° \cdot t) & 0 & 0 \\ \sin(rotations \cdot 360° \cdot t) & \cos(rotations \cdot 360° \cdot t) & 0 & 0 \\ 0 & 0 & 1 & h \cdot t \\ 0 & 0 & 0 & 1 \end{pmatrix}
\end{aligned}
$$

Task 3.2.   1) The coordinate frames of the manipulator are chosen with respect to the modified Denavit-Hartenberg-convention, because the coordinate frames are positioned at the joints themselves.

The DH-table is then specified by:

Link i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0°	0	0	0°
2	-90°	0	0	0°
3	0°	l_a	0	0°

2) We split up the transformation into three subtransformations

$$^0T_3 = A_1 A_2 A_3$$

These can be calculated with the matrix in the slides for the modified Denavit-Hartenberg convention.

$$A_i = \begin{pmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 & a_{i-1} \\ \sin(\theta_i)\cos(\alpha_{i-1}) & \cos(\theta_i)\cos(\alpha_{i-1}) & -\sin(\alpha_{i-1}) & -\sin(\alpha_{i-1})d_i \\ \sin(\theta_i)\sin(\alpha_{i-1}) & \cos(\theta_i)\sin(\alpha_{i-1}) & \cos(\alpha_{i-1}) & \cos(\alpha_{i-1})d_i \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$A_1 = \begin{pmatrix} \cos(0) & -\sin(0) & 0 & 0 \\ \sin(0)\cos(0) & \cos(0)\cos(0) & -\sin(0) & -\sin(0)\cdot 0 \\ \sin(0)\sin(0) & \cos(0)\sin(0) & \cos(0) & \cos(0)\cdot 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} \cos(0) & -\sin(0) & 0 & 0 \\ \sin(0)\cos(-90) & \cos(0)\cos(-90) & -\sin(-90) & -\sin(-90)\cdot 0 \\ \sin(0)\sin(-90) & \cos(0)\sin(-90) & \cos(-90) & \cos(-90)\cdot 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$A_3 = \begin{pmatrix} \cos(0) & -\sin(0) & 0 & l_a \\ \sin(0)\cos(0) & \cos(0)\cos(0) & -\sin(0) & -\sin(0)\cdot 0 \\ \sin(0)\sin(0) & \cos(0)\sin(0) & \cos(0) & \cos(0)\cdot 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & l_a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Now we can multiply these matrices to get the overall transformation

$$^0T_3 = A_1 A_2 A_3$$

$$= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & l_a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & l_a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & 0 & l_a \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Task 3.3. 1) To calculate $^{base}T_{object}$, we need to invert the transformation $^{camera}T_{base}$. Then we can determine the desired transformation by computing

$$^{base}T_{object} = {}^{camera}T_{base}{}^{-1} \cdot {}^{camera}T_{object}$$
$$= {}^{base}T_{camera} \cdot {}^{camera}T_{object}$$

The inverse of $^{camera}T_{base}$ is

$$
\begin{pmatrix}
0 & -1 & 0 & 25 \\
-1 & 0 & 0 & 15 \\
0 & 0 & -1 & 20 \\
0 & 0 & 0 & 1
\end{pmatrix}
$$

so we can compute the resulting matrix now

$$
\begin{aligned}
^{base}T_{object} &= {}^{camera}T_{base}{}^{-1} \cdot {}^{camera}T_{object} \\
&= \begin{pmatrix}
0 & -1 & 0 & 25 \\
-1 & 0 & 0 & 15 \\
0 & 0 & -1 & 20 \\
0 & 0 & 0 & 1
\end{pmatrix} \cdot \begin{pmatrix}
0 & -1 & 0 & 0 \\
-1 & 0 & 0 & -5 \\
0 & 0 & -1 & 19 \\
0 & 0 & 0 & 1
\end{pmatrix} \\
&= \begin{pmatrix}
1 & 0 & 0 & 30 \\
0 & 1 & 0 & 15 \\
0 & 0 & 1 & 1 \\
0 & 0 & 0 & 1
\end{pmatrix}
\end{aligned}
$$

We can see, that the resulting homogeneous transformation is only a translation. This makes sense, because both the coordinate frame axes of the base and the part are parallel to each other.

2) We assume, that the front surface of the object is the surface in the figure, that we can see in front. Further we assume, that the robot will grasp the object by rotating its tool tip by $-90°$ around the z^t-axis. Then, when grasping the object, x^t and x^p are aligned. To align the other axes, we need to have a transformation by $180°$ around the x^p-axis. We already now the transformation $^{base}T_{object}$, and we can determine $^{object}T_{tool}$ with the given rotation. Then we can compute the resulting transformation

$$
\begin{aligned}
^{base}T_{tool} &= {}^{base}T_{object} \cdot {}^{object}T_{tool} \\
&= \begin{pmatrix}
1 & 0 & 0 & 30 \\
0 & 1 & 0 & 15 \\
0 & 0 & 1 & 1 \\
0 & 0 & 0 & 1
\end{pmatrix} \cdot \begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & \cos(180°) & -\sin(180°) & 0 \\
0 & \sin(180°) & \cos(180°) & 0 \\
0 & 0 & 0 & 1
\end{pmatrix} \\
&= \begin{pmatrix}
1 & 0 & 0 & 30 \\
0 & \cos(180°) & -\sin(180°) & 15 \\
0 & \sin(180°) & \cos(180°) & 1 \\
0 & 0 & 0 & 1
\end{pmatrix} \\
&= \begin{pmatrix}
1 & 0 & 0 & 30 \\
0 & -1 & 0 & 15 \\
0 & 0 & -1 & 1 \\
0 & 0 & 0 & 1
\end{pmatrix}
\end{aligned}
$$

3)

4)

Task 3.4. The positioning accuracy depends on several things. These are for example:

- Calibration of the camera system: If the camera can't accuratly determine every position in the space, there won't be an accurate positioning.

- Joint position accuracy: If the joints are not precise in achieving their position, then the overall positioning is not very accurate.

- Temperature differences: If the temperature varies, the metal of the links (if metal is used) varies in its length and therefore leads to a more unprecise positioning of the endeffector.

- Position of the base: If the base is not properly positioned in space, then the endeffector won't be very accurate.

- Calculation impreciseness: The homogeneous transformation $^{Base}T_{Tooltip}$ can't be precise to 100%, because most of the sine- and cosine-calculations have irrational numbers as results. This leads to the questions, how many decimal places are required to satisfy the calculation results.

- Sensor noise: If sensors are giving the wrong signals, the system could calculate with the wrong numbers and would assume a false position, which will be "corrected". his would lead to a wrong position at the end.

A limit for the positioning accuracy in this setup (vision system based) is on pixel level. If the robot satisfies the position for the vision system, no more movements are made. Then the area of points around the specified position, where the vision system confirms the desired position, has unlimited points, but are lying in a specific radius (which is determined by the resolution of one pixel of the vision system). All of these points should be a good estimation for the real position.