Assignment 2

ELEC 442 - Introduction to Robotics

Sondre Myrberg (81113433) Ola Helbaek (68776772)

October 27, 2017

THE UNIVERSITY OF BRITISH COLUMBIA

Solving the inverse kinematics problem for the PUMA 560 robot

For the first angle, θ_1 we use that the centre of the spherical wrist, $\underset{\sim}{o_4}$, compared to the xy-plane is only dependent on θ_1. We also know that for $\theta_1 = 0$ $\underset{\sim}{o_4}$ must be in the plane defined by $y - 149.09\text{mm} = 0$. This gives us the relation between u and w, where w is the vector between the origin and $\underset{\sim}{o_4}$ and u is the vector to where $\underset{\sim}{o_4}$ would be if $\theta_1 = 0$, such that we kan use the inplemented function `KahanP2`. The relation is given as

$$\begin{aligned}
\|w\|^2 = o_{4,x}^2 + o_{4,y}^2 + o_{4,z}^2 &= u_x^2 + u_y^2 + u_z^2 = \|u\|^2 \\
\implies o_{4,x}^2 + o_{4,y}^2 + \cancel{o_{4,z}^2} &= u_x^2 + 149.09^2 + \cancel{o_{4,z}^2} && \text{as desired } z\text{-value doesn't} \\
\implies u_x &= \sqrt{o_{4,x}^2 + o_{4,y}^2 - 149.09^2} && \text{change with } \theta_1
\end{aligned}$$

Given this and that $\underset{\sim}{o_0} = \begin{pmatrix} 0 & 0 & 0 \end{pmatrix}^\top$ we can use the implemented function `KahanP2` with $s = \boldsymbol{k}_0$, $\hat{\boldsymbol{u}} = \begin{bmatrix} u_x & 149.09 & o_{4,z} \end{bmatrix}$ and $\boldsymbol{w} = \underset{\sim}{o_4} - \underset{\sim}{o_0}$. The function is implemented as shown in Listing 1. In this implementation the function itself normalizes the vectors such that we do not have to think about feeding this into the function.

```
function theta = KahanP2(s,u,w)
    s_hat = s/norm(s);
    u_hat = u/norm(u);
    w_hat = w/norm(w);
    if s_hat'*cross(s_hat,u_hat) == s_hat'*cross(s_hat,w_hat)
        theta = 2*atan(norm(cross(s_hat,(u_hat-w_hat)))/norm(cross(s_hat,(u_hat+w_hat))));
        if w_hat'*cross(s_hat,(u_hat-w_hat)) < 0
            theta = -theta;
        end
    else
        theta = 'The solution does not exist';
    end
end
```

Listing 1: MATLAB implementation of the Kahan P2 problem

For θ_3 we use the implementation of `KahanP4` given in Listing 2. Here we use $a = \boldsymbol{o}_2 - \boldsymbol{o}_0$, $b = \boldsymbol{o}_4 - \boldsymbol{o}_2$ and $c = \boldsymbol{o}_4 - \boldsymbol{o}_0$ where all these are projected on to the $\boldsymbol{i}_1, \boldsymbol{j}_1$-plane since all change in θ_2 and θ_3 will only give a change in the position of $\underset{\sim}{o_4}$ with respect to this plane. For the home position of the manipulator, we sill get a small angle for θ_3. This angle is the offset of θ_3 and is taken care of by doing `KahanP4` with the distances a, b and c as in the home postion and subtracting it from the solution shown over.

```
function theta = KahanP4(a,b,c)
    if a+b>=c && c>=abs(a-b)
        theta = 2*atan(((a+b)^2 -c^2)/(c^2 - (a-b)^2));
    else
        theta = 'No solution';
    end
end
```

Listing 2: MATLAB implementation of the Kahan P4 problem

For θ_2 we used the law of cosine with the same triangle as for θ_3. This would be less accurate for small angles, and probably not the best way to do this.

After finding the angles of the arm, we use forward kinematics to identidy the wrist centre, as this is only dependent on the joint variables of the arm, in addition to the base frame $\underline{\boldsymbol{C}}_3$. We need $\underline{\boldsymbol{C}}_3$ in order to find the remaining joint angles. These can be found on the same way as shown in the lectures and is shown in (1).

$$\begin{aligned} \underline{\boldsymbol{k}}_4 &= \pm \frac{\underline{\boldsymbol{k}}_3 \times \underline{\boldsymbol{k}}_6}{\|\underline{\boldsymbol{k}}_3 \times \underline{\boldsymbol{k}}_6\|} \\ \exp\{(\pi - \theta_6)\underline{\boldsymbol{k}}_6 \times\}\underline{\boldsymbol{j}}_6 &= \exp\{-(\theta_6 - \pi)\underline{\boldsymbol{k}}_6 \times\}\underline{\boldsymbol{j}}_6 \quad = \underline{\boldsymbol{k}}_4 \\ &\exp\{(\theta_4 + \pi)\underline{\boldsymbol{k}}_3 \times\}\underline{\boldsymbol{j}}_3 \quad = \underline{\boldsymbol{k}}_4 \\ &\exp\{\theta_5 \underline{\boldsymbol{k}}_4 \times\}\underline{\boldsymbol{k}}_3 \quad = \underline{\boldsymbol{k}}_6 \end{aligned} \tag{1}$$

With these four equations we can use `KahanP2` three times with the vectors from the equations as input, but since we didn't find any perticularly good solutions for θ_2 and θ_3 this has not been inplemented. The general form of a Kahan P2-problem solving for θ is however $\exp\{\theta \underline{\hat{\boldsymbol{s}}} \times\}\underline{\hat{\boldsymbol{u}}} = \underline{\hat{\boldsymbol{w}}}$, so we easily see from (1) which vectors to use as inputs in our `KahanP2`-function. This would obviously not give all solutions, but they can be found looking at joint restrictions and polarity. Joint 2 and 3 are good examples, as for a given location for $\underset{\sim}{\boldsymbol{o}}_4$ we have two different ways of traversing from $\underset{\sim}{\boldsymbol{o}}_0$, one with 'elbow' up and one with 'elbow' down.