Two DOF planar robot model predictive control

R.B. Ashith Shyam

1 Equation of motion in state-space

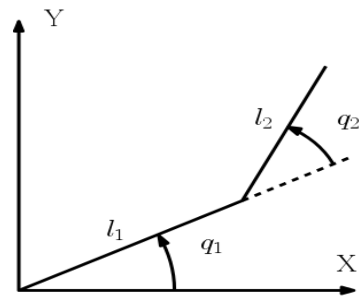

Figure 1: Schematic of a planar 2-DOF robot

The equation of motion of a 2R planar robot can be expressed as

$$[M]\begin{bmatrix}\ddot{q}_1\\ \ddot{q}_2\end{bmatrix} + C + G = \begin{bmatrix}u_1\\ u_2\end{bmatrix} \tag{1}$$

where $q_i's$ and $u_i's$, i $= 1, 2$ are the joint angles and the input torques in joint space
In state-space form this can be written as ($q_1 = x_1,\ q_2 = x_2,\ \dot{q}_1 = x_3$ and $\dot{q}_2 = x_4$)

$$\dot{X} = \begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\end{bmatrix} = \begin{bmatrix} x_3\\ x_4\\ M^{-1}\Big[\begin{pmatrix}u_1\\ u_2\end{pmatrix} - C - G\Big]\end{bmatrix} \tag{2}$$

2 Feedback Linearization

$$\begin{bmatrix}u_1\\ u_2\end{bmatrix} = \alpha \begin{bmatrix}u_1'\\ u_2'\end{bmatrix} + \beta \tag{3}$$

Choose $\alpha = M$ and $\beta = C + G$, équation (2) becomes

$$\dot{X} = \begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\end{bmatrix} = \begin{bmatrix}0&0&1&0\\0&0&0&1\\0&0&0&0\\0&0&0&0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\end{bmatrix} + \begin{bmatrix}0&0\\0&0\\1&0\\0&1\end{bmatrix}\begin{bmatrix}u_1\\ u_2\end{bmatrix} \tag{4}$$

To track the joint angles, the output equation becomes

$$Y = \begin{pmatrix}x_1\\ x_2\end{pmatrix} = \begin{bmatrix}1&0&0&0\\0&1&0&0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\end{bmatrix} \tag{5}$$

So the linearized system dynamics becomes,

$$\begin{aligned}\dot{X} &= A\ X + B\ U\\ Y &= C\ X\end{aligned} \tag{6}$$

where

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, X = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} \text{ and U} = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \tag{7}$$

3 Output trajectory tracking by MPC

Let y_r be the reference trajectory the tip of the end-effector should follow, N be the prediction horizon, the cost function, J, can be written as

$$J = (y_{N/k} - y_r)^T P'(y_{N/k} - y_r) + \sum_{i=0}^{N-1} (y_{i/k} - y_r)^T Q'(y_{i/k} - y_r) + u_{i/k}^T R u_{i/k} \tag{8}$$

$$\begin{aligned} &= (Cx_{N/k} - y_r)^T P'(Cx_{N/k} - y_r) + \sum_{i=0}^{N-1} (Cx_{i/k} - y_r)^T Q'(Cx_{i/k} - y_r) + u_{i/k}^T R u_{i/k} \\ &= x_{N/k}^T C^T P' C x_{N/k} - 2x_{N/k}^T C^T P' y_r + y_r^T P' y_r + \sum_{i=0}^{N-1} x_{i/k}^T C^T Q' C x_{i/k} - 2x_{i/k}^T C^T Q' y_r + y_r^T Q' y_r + u_{i/k}^T R u_{i/k} \end{aligned} \tag{9}$$

where P', Q', R are penalties. This can be further simplified and can be written as

$$J = y_r^T P' y_r + x_{0/k}^T Q x_{0/k} - 2x_{0/k} Qd y_r + \begin{pmatrix} x_1 \\ x_2 \\ : \\ : \\ x_N \end{pmatrix}_k^T \begin{bmatrix} Q & & & & \\ & Q & & & \\ & & : & & \\ & & & : & \\ & & & & P \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ : \\ : \\ x_N \end{pmatrix}_k - 2 \begin{pmatrix} x_1 \\ x_2 \\ : \\ : \\ x_N \end{pmatrix}_k^T \begin{bmatrix} Qd & & & & \\ & Qd & & & \\ & & : & & \\ & & & : & \\ & & & & Pd \end{bmatrix} \begin{pmatrix} y_r \\ y_r \\ : \\ : \\ y_r \end{pmatrix}_k$$

$$+ \begin{pmatrix} u_0 \\ u_1 \\ : \\ : \\ u_{N-1} \end{pmatrix}_k^T \begin{bmatrix} R & & & & \\ & R & & & \\ & & : & & \\ & & & : & \\ & & & & R \end{bmatrix} \begin{pmatrix} u_0 \\ u_1 \\ : \\ : \\ u_{N-1} \end{pmatrix}_k$$

$$J = y_r^T P' y_r + x_{0/k}^T \; Qd \; x_{0/k} - 2x_{0/k} Q y_r + X_k^T F X_k - 2X_k^T G y_{r_k} + U_k^T H U_k \tag{10}$$

where P $= C^T P' C$, Pd $= C^T P'$, Q $= C^T Q' C$, Qd $= C^T Q'$, F $= \begin{bmatrix} Q & & & & \\ & Q & & & \\ & & : & & \\ & & & : & \\ & & & & P \end{bmatrix}$, G $= \begin{bmatrix} Qd & & & & \\ & Qd & & & \\ & & : & & \\ & & & : & \\ & & & & Pd \end{bmatrix}$,

H $= \begin{bmatrix} R & & & & \\ & R & & & \\ & & : & & \\ & & & : & \\ & & & & R \end{bmatrix}$ and $Y_R = \begin{pmatrix} y_r \\ y_r \\ : \\ : \\ y_r \end{pmatrix}_k$

For a control horizon of N, we can write the following for the kth sampling time

$$
\begin{aligned}
x_1 &= Ax_0 + Bu_0\\
x_2 &= Ax_1 + Bu_1 = A\Big(Ax_0 + Bu_0\Big) + Bu_1 = A^2x_0 + ABu_0 + Bu_1\\
&:\\
&:\\
x_N &= Ax_{N-1} + Bu_{N-1} = A^Nx_0 + A^{N-2}Bu_1 + A^{N-3}Bu_2 + ...Bu_{N-1}
\end{aligned}
$$

$$
\begin{pmatrix} x_1 \\ x_2 \\ : \\ : \\ x_N \end{pmatrix}_k = \begin{pmatrix} A \\ A^2 \\ : \\ : \\ A^N \end{pmatrix} x_0 + \begin{bmatrix} B & 0 & .... & 0 \\ AB & B & .... & 0 \\ : & : & .... & : \\ : & : & .... & : \\ A^{N-1}B & A^{N-2}B & .... & B \end{bmatrix} \begin{pmatrix} u_0 \\ u_1 \\ : \\ : \\ u_{N-1} \end{pmatrix}_k
$$

$$
X_k = S_x x_0 + S_u U_k \tag{11}
$$

Substituting équation (11) in équation (10), the cost becomes

$$
J = y_r^T P' y_r + x_{0/k}^T \; Qd \; x_{0/k} - 2x_{0/k} Q y_r + (S_x x_0 + S_u U_k)^T F (S_x x_0 + S_u U_k) - 2(S_x x_0 + S_u U_k)^T G y_{r_k} + U_k^T H U_k \tag{12}
$$

Since U_k is the variable for optimization, the terms independent of U_k can be omitted in the cost function

$$
\begin{aligned}
J &= U_k^T (2S_u^T F S_x) x_0 + U_k^T (S_u^T F S_u + H) U_k - U_k^T (2S_u^T G) Y_R\\
&= \frac{1}{2} U_k^T W U_k + U_k^T K x_0 - U_k^T L Y_R
\end{aligned} \tag{13}
$$

where W = 2 $(S_u^T F S_u + H)$, K = $2S_u^T F S_x$ and L = $2S_u^T G$