pyMPC Documentation

Marco Forgione

May 21, 2019

1 Mathematical formulation

The MPC problem to be solved is:

$$
\begin{aligned}
&\arg \min_{u_0, u_1, \dots, u_{N_p-1}, x_0, x_1, \dots, x_{N_p}} \big(x_N - x_{ref}\big)^\top Q_{x_N} \big(x_N - x_{ref}\big) + \\
&\sum_{k=0}^{N_p-1} \big(x_k - x_{ref}\big)^\top Q_x \big(x_k - x_{ref}\big) + \big(u_k - u_{ref}\big)^\top Q_u \big(u_k - u_{ref}\big) + \Delta u_k^\top Q_{\Delta u} \Delta u_k && \text{(1a)}\\
&\text{subject to}\\
&x_{k+1} = A x_k + B u_k && \text{(1b)}\\
&u_{min} \leq u_k \leq u_{max} && \text{(1c)}\\
&x_{min} \leq x_k \leq x_{max} && \text{(1d)}\\
&\Delta u_{min} \leq \Delta u_k \leq \Delta u_{min} && \text{(1e)}\\
&x_0 = \bar{x} && \text{(1f)}\\
&u_{-1} = \bar{u} && \text{(1g)}
\end{aligned}
$$

where $\Delta u_k = u_k - u_{k-1}$.

In a typical implementation, the MPC input is applied in *receding horizon*. At each time step i, the problem (1) is solved with $x_0 = x[i]$, $u_{-1} = u[i-1]$ and an optimal input sequence $u_0, \dots, u_{N_p}$ is obtained. The first element of this sequence u_0 is the control input that is actually applied at time instant i. At time instant $i+1$, a new state $x[i+1]$ is measured (or estimated), and the process is iterated.

Thus, formally, the MPC control law is a (static) function of the current state and the previous input:

$$u_{MPC} = K(x[i], u[i-1]). \tag{2}$$

Note that this function possibly depends on the references x_{ref} and u_{ref} and on the system matrices A and B.

1.1 Quadratic Programming Formulation

The QP solver expets a problem with form:

$$\min \frac{1}{2} x^\top P x + q^\top x \tag{3a}$$

subject to

$$l \le Ax \le u \tag{3b}$$

The difficulty is to rewrite the MPC optimization problem (1) in form (3) to use the standard QP solver.