

Linear Reference Tracking MPC

Timothy Werner

Abstract—This paper briefly introduces the implemented Linear Reference Tracking MPC scheme and a slight generalization to nonlinear systems in the sense of linear time varying approximations.

I. THE TRACKING PROBLEM

We consider linear discrete time systems, sampled at rate δ of the form

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + B\mathbf{u}_k \quad (1a)$$

$$\mathbf{y}_k = C\mathbf{x}_k, \quad (1b)$$

where $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^n$ are the states, $\mathbf{u} \in \mathcal{U} \subseteq \mathbb{R}^p$ are the inputs and $\mathbf{y} \in \mathcal{Y} \subseteq \mathbb{R}^q$ are the outputs of the system. The goal is to steer the output along a known reference trajectory $\mathbf{r} \in \mathcal{Y}$, i.e. to render the reference error

$$\mathbf{e}_k := \mathbf{y}_k - \mathbf{r}_k \triangleq C\mathbf{x}_k - \mathbf{r}_k \rightarrow 0,$$

as $k \rightarrow \infty$. We assume a given trajectory \mathcal{T} , such that

$$\mathcal{T}(t_k) : \mathbb{R} \rightarrow \mathbb{R}^q, t_k \rightarrow \mathbf{r}_k.$$

Here t_k refers to a discrete point in time. In that sense $\mathcal{T}(t_k)$ can be a continuous mapping or a set of discrete points. Note that we consider the state-space, the input-space and the output-space to be constrained. We will also consider rate constraints because all physical systems are rate-constrained. This extension also appears natural for two more reasons which will be discussed briefly in the following. To this end, we define the rate as

$$\Delta \mathbf{u}_k := \mathbf{u}_k - \mathbf{u}_{k-1}. \quad (2)$$

II. THE LINEAR TRACKING MPC

It appears natural to consider the tracking error as defined above as the objective to be minimized. We can thus formulate the cost

$$\mathcal{J}_j := \sum_{k=0}^{N-1} \|\mathbf{e}_{j+k}\|_Q^2 + \|\Delta \mathbf{u}_{j+k}\|_{R_\Delta}^2 \quad (3)$$

where N is the discrete MPC horizon over which the optimal input sequence is to be found and we denoted $\|\mathbf{e}\|_Q^2 := \mathbf{e}^\top Q \mathbf{e}$. Note that a rate-regularization was introduced to the cost, which ensures a smooth tracking, which is one of the two main reasons to consider input rates.

III. IMPLEMENTATION

A. The Cost Function

To enforce stability on the controller, we introduce a terminal constraint that drives the predicted trajectory onto the desired trajectory \mathbf{r} near the end of the prediction horizon. We further introduce a direct penalty on the inputs \mathbf{u} due to a state augmentation to be introduced later. We have thus

$$\mathcal{J}_j := \sum_{k=0}^{N-1} \|\mathbf{e}_{j+k}\|_Q^2 + \|\Delta \mathbf{u}_{j+k}\|_{R_\Delta}^2 + \|\mathbf{u}_{j+k}\|_R^2 + \|\mathbf{e}_{j+N}\|_P^2 \quad (4)$$

If we further investigate the terms, omitting the constant terms w.r.t to \mathbf{x} and \mathbf{u} , we find

$$\begin{aligned} \mathcal{J}_j := \sum_{k=0}^{N-1} & \mathbf{x}_{j+k}^\top C^\top Q C \mathbf{x}_{j+k} \\ & + \mathbf{u}_{j+k}^\top R \mathbf{u}_{j+k} \\ & + \Delta \mathbf{u}_{j+k}^\top R_\Delta \Delta \mathbf{u}_{j+k} \\ & - 2\mathbf{r}_{j+k}^\top Q C \mathbf{x}_{j+k} \\ & + \mathbf{x}_{j+N}^\top C^\top P C \mathbf{x}_{j+N} \\ & - 2\mathbf{r}_{j+N}^\top P C \mathbf{x}_{j+N} \end{aligned} \quad (5)$$

B. State Augmentation

We define the new state $\tilde{\mathbf{x}}_k := [\mathbf{x}_k^\top \quad \mathbf{u}_{k-1}^\top]^\top$ and combining (1a) and (2), we find the new augmented system

$$\tilde{\mathbf{x}}_{k+1} = \begin{bmatrix} A & B \\ I & \end{bmatrix} \tilde{\mathbf{x}}_k + \begin{bmatrix} B \\ I \end{bmatrix} \Delta \mathbf{u}_k, \quad (6)$$

which we will refer to as

$$\tilde{\mathbf{x}}_{k+1} = \tilde{A} \tilde{\mathbf{x}}_k + \tilde{B} \Delta \mathbf{u}_k.$$

This state augmentation was mainly performed to ensure zero steady state error independent of the weights chosen for error penalty and input penalty. With a mere error and input penalty, the steady state error would be directly dependent on the weights chosen for either term. Imagine to this end a high weight on the inputs and a low weight on the reference error. The controller would not put a lot of effort in tracking the reference, since it would cost a lot of input. A high weight on tracking error and a low weight on the inputs on the other hand would lead to a bang-bang behavior which is in general not a desired control strategy. Choosing the reference error and the rates to be minimized on the other hand resolves in a well posed problem, where the weights on error and input rates shift the pareto-solution between aggressive and smooth tracking. The drawback of this method is that the

dimension of the problem is raised from $(N+1)n + Np$ to $(N+1)(n+p) + Np$. Rewriting the cost function in terms of (6), we find

$$\begin{aligned} \mathcal{J}_j := & \sum_{k=0}^{N-1} \tilde{\mathbf{x}}_{j+k}^\top \tilde{Q} \tilde{\mathbf{x}}_{j+k} \\ & + \Delta \mathbf{u}_{j+k}^\top R_\Delta \Delta \mathbf{u}_{j+k} \\ & - C_Q^\top \tilde{\mathbf{x}}_{j+k} \\ & + \tilde{\mathbf{x}}_{j+N}^\top \tilde{P} \tilde{\mathbf{x}}_{j+N} \\ & - C_P^\top \tilde{\mathbf{x}}_{j+N}, \end{aligned} \quad (7)$$

with

$$\tilde{Q} := \begin{bmatrix} C^\top Q C & \\ & R \end{bmatrix},$$

$$\tilde{P} := \begin{bmatrix} C^\top P C & \\ & R \end{bmatrix},$$

$$C_Q^\top := [2\mathbf{r}_k^\top Q C \quad 0],$$

and

$$C_P^\top := [2\mathbf{r}_k^\top P C \quad 0]$$

C. The Optimization Problem

With this defined, we can formulate the QP to be solved at time instance $t = j$.

$$\text{minimize} \quad \mathcal{J}_j \quad (8a)$$

$$\text{subject to} \quad \tilde{\mathbf{x}}_0^\top = \left[\mathbf{x}_j \quad \left(\mathbf{u}_{0|j-1}^\top \right)^* \right]^\top \quad (8b)$$

$$\tilde{\mathbf{x}}_{j+k+1} = \tilde{A} \tilde{\mathbf{x}}_{j+k} + \tilde{B} \Delta \mathbf{u}_{j+k} \quad (8c)$$

$$\underline{\mathbf{x}} \leq \mathbf{x}_{j+k} \leq \bar{\mathbf{x}} \quad (8d)$$

$$\underline{\mathbf{y}} \leq C \mathbf{x}_{j+k} \leq \bar{\mathbf{y}} \quad (8e)$$

$$\underline{\mathbf{u}} \leq \mathbf{u}_{j+k} \leq \bar{\mathbf{u}} \quad (8f)$$

$$\underline{\Delta \mathbf{u}} \leq \Delta \mathbf{u}_{j+k} \leq \overline{\Delta \mathbf{u}} \quad (8g)$$

The solution of this yields the optimal input sequence Ω_j^* and the corresponding predicted state trajectory Ξ_j^* at time instance j

$$\Omega_j^* := \left\{ \mathbf{u}_{0|j}^* \quad \mathbf{u}_{1|j}^* \quad \cdots \quad \mathbf{u}_{N-1|j}^* \right\} \quad (9)$$

$$\Xi_j^* := \left\{ \mathbf{x}_{0|j}^* \quad \mathbf{x}_{1|j}^* \quad \cdots \quad \mathbf{x}_{N|j}^* \right\} \quad (10)$$

$$(11)$$

where $\{k|j\}$ indicates the value at MPC stage $k \in [0, N]$ at time instance $t = j$. The controller will then output $\mathbf{u}(\mathbf{x}_j, j) = \mathbf{u}_{0|j}^*$ and apply it to (1a). Note that (8b) depends on the first stage of the optimal input sequence of the previous time step $\left(\mathbf{u}_{0|j-1}^\top \right)^*$. This ensures satisfaction of the rate constraints of the resulting closed-loop input sequence

$$\Omega_j := \left\{ \mathbf{u}_{0|0}^* \quad \mathbf{u}_{1|\delta}^* \quad \cdots \right\}. \quad (12)$$

D. Different formulation

In general one directly calls a QP solver and handles the matrices defining the objective to be minimized and the constraints the problem is subject to. More sophisticated solvers allow for a slightly more loose problem formulation and handle the matrix stacking in the background. This increases the robustness and allows for a formulation that is tailored for the specific QP solver which increases the run time effectively. In this sense, another common formulation of the objective is

$$\begin{aligned} \mathcal{J}_j := & \sum_{k=0}^{N-1} \begin{bmatrix} C \mathbf{x}_{j+k} - \mathbf{r}_{j+k} \\ \mathbf{u}_{j+k} \end{bmatrix}^\top \begin{bmatrix} Q & \\ & R \end{bmatrix} \begin{bmatrix} C \mathbf{x}_{j+k} - \mathbf{r}_{j+k} \\ \mathbf{u}_{j+k} \end{bmatrix} \\ & + \Delta \mathbf{u}_{j+k}^\top R_\Delta \Delta \mathbf{u}_{j+k} \\ & + \begin{bmatrix} C \mathbf{x}_{j+N} - \mathbf{r}_{j+N} \\ \mathbf{u}_{j+N} \end{bmatrix}^\top \begin{bmatrix} P & \\ & R \end{bmatrix} \begin{bmatrix} C \mathbf{x}_{j+N} - \mathbf{r}_{j+N} \\ \mathbf{u}_{j+N} \end{bmatrix} \end{aligned} \quad (13)$$

IV. GENERALIZATION TO NONLINEAR SYSTEMS