《高阶会员专属视频》
-第10期

模型预测控制（MPC）
要如何使用计算机实现？
带你导读MPC经典著作！

Liuping Wang

AIC

Advances in Industrial Control

Model Predictive Control System Design and Implementation Using MATLAB®

Springer

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

1.2.1 Single-input and Single-output System

For simplicity, we begin our study by assuming that the underlying plant is a single-input and single-output system, described by:

$$x_m(k+1) = A_m x_m(k) + B_m u(k), \tag{1.1}$$
$$y(k) = C_m x_m(k), \tag{1.2}$$

where u is the manipulated variable or input variable; y is the process output; and x_m is the state variable vector with assumed dimension n_1. Note that this plant model has $u(k)$ as its input. Thus, we need to change the model to suit our design purpose in which an integrator is embedded.

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

- STEP 1: $\frac{dx(t)}{dt} = Ax(t) + Bu(t)$
- STEP 2: $\frac{x(k+1)-x(k)}{\Delta T} = A\mathrm{x}(k) + Bu(k)$
- STEP 3: $x(k+1) - x(k) = \Delta T \cdot A \cdot x(k) + \Delta T \cdot B \cdot u(k)$
- STEP 4: $x(k+1) = (\Delta T \cdot A + 1) \cdot x(k) + \Delta T \cdot B \cdot u(k)$

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

Note that a general formulation of a state-space model has a direct term from the input signal $u(k)$ to the output $y(k)$ as

$$y(k) = C_m x_m(k) + D_m u(k).$$

However, due to the principle of receding horizon control, where a current information of the plant is required for prediction and control, we have implicitly assumed that the input $u(k)$ cannot affect the output $y(k)$ at the same time. Thus, $D_m = 0$ in the plant model.

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

Taking a difference operation on both sides of (1.1), we obtain that

$$\underline{x_m(k+1) - x_m(k)} = A_m(\underline{x_m(k) - x_m(k-1)}) + B_m(\underline{u(k) - u(k-1)}).$$

Let us denote the difference of the state variable by

$$\Delta x_m(k+1) = x_m(k+1) - x_m(k); \quad \Delta x_m(k) = x_m(k) - x_m(k-1),$$

and the difference of the control variable by

$$\Delta u(k) = u(k) - u(k-1).$$

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

These are the increments of the variables $x_m(k)$ and $u(k)$. With this transformation, the difference of the state-space equation is:

$$\boxed{\Delta x_m(k+1) = A_m \Delta x_m(k) + B_m \Delta u(k). \qquad (1.3)}$$

Note that the input to the state-space model is $\Delta u(k)$. The next step is to connect $\Delta x_m(k)$ to the output $y(k)$. To do so, a new state variable vector is chosen to be

$$x(k) = \left[\, \Delta x_m(k)^T \; y(k) \,\right]^T,$$

where superscript ^{T} indicates matrix transpose. Note that

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

$$y(k+1) - y(k) = C_m(x_m(k+1) - x_m(k)) = C_m \Delta x_m(k+1)$$
$$= C_m A_m \Delta x_m(k) + C_m B_m \Delta u(k). \tag{1.4}$$

Putting together (1.3) with (1.4) leads to the following state-space model:

$$\overbrace{\begin{bmatrix} \Delta x_m(k+1) \\ y(k+1) \end{bmatrix}}^{x(k+1)} = \overbrace{\begin{bmatrix} A_m & o_m^T \\ C_m A_m & 1 \end{bmatrix}}^{A} \overbrace{\begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix}}^{x(k)} + \overbrace{\begin{bmatrix} B_m \\ C_m B_m \end{bmatrix}}^{B} \Delta u(k)$$
$$y(k) = \overbrace{\begin{bmatrix} o_m & 1 \end{bmatrix}}^{C} \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix}, \tag{1.5}$$

where $o_m = \overbrace{\begin{bmatrix} 0 \; 0 \ldots 0 \end{bmatrix}}^{n_1}$. The triplet (A, B, C) is called the augmented model, which will be used in the design of predictive control.

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

Example 1.1. Consider a discrete-time model in the following form:

$$
\begin{aligned}
x_m(k+1) &= A_m x_m(k) + B_m u(k) \\
y(k) &= C_m x_m(k)
\end{aligned}
\tag{1.6}
$$

where the system matrices are

$$
A_m = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}; B_m = \begin{bmatrix} 0.5 \\ 1 \end{bmatrix}; C_m = \begin{bmatrix} 1 & 0 \end{bmatrix}.
$$

Find the triplet matrices (A, B, C) in the augmented model (1.5) and calculate the eigenvalues of the system matrix, A, of the augmented model.

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

Solution. From (1.5), $n_1 = 2$ and $o_m = [0\ 0]$. The augmented model for this plant is given by

$$\begin{aligned} x(k+1) &= Ax(k) + B\Delta u(k) \\ y(k) &= Cx(k), \end{aligned} \tag{1.7}$$

where the augmented system matrices are

$$A = \begin{bmatrix} A_m & o_m^T \\ C_m A_m & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix};\ B = \begin{bmatrix} B_m \\ C_m B_m \end{bmatrix} = \begin{bmatrix} 0.5 \\ 1 \\ 0.5 \end{bmatrix};$$

$$C = \begin{bmatrix} o_m & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}.$$

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

1.3.1 Prediction of State and Output Variables

Assuming that at the sampling instant k_i, $k_i > 0$, the state variable vector $x(k_i)$ is available through measurement, the state $x(k_i)$ provides the current plant information. The more general situation where the state is not directly measured will be discussed later. The future control trajectory is denoted by

$$\Delta u(k_i), \Delta u(k_i + 1), \ldots, \Delta u(k_i + N_c - 1),$$

where N_c is called the control horizon dictating the number of parameters used to capture the future control trajectory. With given information $x(k_i)$, the future state variables are predicted for N_p number of samples, where N_p is called the prediction horizon. N_p is also the length of the optimization window. We denote the future state variables as

$$x(k_i + 1 \mid k_i),\ x(k_i + 2 \mid k_i),\ \ldots,\ x(k_i + m \mid k_i),\ \ldots,\ x(k_i + N_p \mid k_i),$$

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！ 带你导读MPC经典著作！

where $x(k_i+m \mid k_i)$ is the predicted state variable at k_i+m with given current plant information $x(k_i)$. The control horizon N_c is chosen to be less than (or equal to) the prediction horizon N_p.

Based on the state-space model (A, B, C), the future state variables are calculated sequentially using the set of future control parameters:

$$
\begin{aligned}
x(k_i+1 \mid k_i) &= Ax(k_i) + B\Delta u(k_i) \\
x(k_i+2 \mid k_i) &= Ax(k_i+1 \mid k_i) + B\Delta u(k_i+1) \\
&= A^2 x(k_i) + AB\Delta u(k_i) + B\Delta u(k_i+1) \\
&\vdots \\
x(k_i+N_p \mid k_i) &= A^{N_p} x(k_i) + A^{N_p-1} B\Delta u(k_i) + A^{N_p-2} B\Delta u(k_i+1) \\
&\quad + \ldots + A^{N_p-N_c} B\Delta u(k_i+N_c-1).
\end{aligned}
$$

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

From the predicted state variables, the predicted output variables are, by substitution

$$
\begin{aligned}
y(k_i+1 \mid k_i) &= CAx(k_i) + CB\Delta u(k_i) \qquad (1.10)\\
y(k_i+2 \mid k_i) &= CA^2x(k_i) + CAB\Delta u(k_i) + CB\Delta u(k_i+1)\\
y(k_i+3 \mid k_i) &= CA^3x(k_i) + CA^2B\Delta u(k_i) + CAB\Delta u(k_i+1)\\
&\quad + CB\Delta u(k_i+2)\\
&\ \ \vdots\\
y(k_i+N_p \mid k_i) &= CA^{N_p}x(k_i) + CA^{N_p-1}B\Delta u(k_i) + CA^{N_p-2}B\Delta u(k_i+1)\\
&\quad + \ldots + CA^{N_p-N_c}B\Delta u(k_i+N_c-1). \qquad (1.11)
\end{aligned}
$$

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

Note that all predicted variables are formulated in terms of current state variable information $x(k_i)$ and the future control movement $\Delta u(k_i+j)$, where $j = 0, 1, \ldots N_c - 1$.

Define vectors

$$Y = \left[y(k_i + 1 \mid k_i) \; y(k_i + 2 \mid k_i) \; y(k_i + 3 \mid k_i) \ldots y(k_i + N_p \mid k_i) \right]^T$$

$$\Delta U = \left[\Delta u(k_i) \; \Delta u(k_i + 1) \; \Delta u(k_i + 2) \ldots \Delta u(k_i + N_c - 1) \right]^T,$$

where in the single-input and single-output case, the dimension of Y is N_p and the dimension of ΔU is N_c. We collect (1.10) and (1.11) together in a compact matrix form as

$$Y = Fx(k_i) + \Phi \Delta U, \tag{1.12}$$

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

where in the single-input and single-output case, the dimension of Y is N_p and the dimension of ΔU is N_c. We collect (1.10) and (1.11) together in a compact matrix form as

$$Y = Fx(k_i) + \Phi \Delta U, \tag{1.12}$$

where

$$F = \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ \vdots \\ CA^{N_p} \end{bmatrix}; \Phi = \begin{bmatrix} CB & 0 & 0 & \dots & 0 \\ CAB & CB & 0 & \dots & 0 \\ CA^2B & CAB & CB & \dots & 0 \\ \vdots & & & & \\ CA^{N_p-1}B & CA^{N_p-2}B & CA^{N_p-3}B & \dots & CA^{N_p-N_c}B \end{bmatrix}.$$

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

1.3.2 Optimization

For a given set-point signal $r(k_i)$ at sample time k_i, within a prediction horizon the objective of the predictive control system is to bring the predicted output as close as possible to the set-point signal, where we assume that the set-point signal remains constant in the optimization window. This objective is then translated into a design to find the 'best' control parameter vector ΔU such that an error function between the set-point and the predicted output is minimized.

Assuming that the data vector that contains the set-point information is

$$R_s^T = \overbrace{\begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix}}^{N_p} r(k_i),$$

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

we define the cost function J that reflects the control objective as

$$J = (R_s - Y)^T (R_s - Y) + \Delta U^T \bar{R} \Delta U, \tag{1.13}$$

where the first term is linked to the objective of minimizing the errors between the predicted output and the set-point signal while the second term reflects the consideration given to the size of ΔU when the objective function J is made to be as small as possible. $\bar{R}$ is a diagonal matrix in the form that $\bar{R} = r_w I_{N_c \times N_c}$ $(r_w \geq 0)$ where r_w is used as a tuning parameter for the desired closed-loop performance. For the case that $r_w = 0$, the cost function (1.13) is interpreted as the situation where we would not want to pay any attention to how large the ΔU might be and our goal would be solely to make the error $(R_s - Y)^T (R_s - Y)$ as small as possible. For the case of large r_w, the cost function (1.13) is interpreted as the situation where we would carefully consider how large the ΔU might be and cautiously reduce the error $(R_s - Y)^T (R_s - Y)$.

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

To find the optimal ΔU that will minimize J, by using (1.12), J is expressed as

$$J = (R_s - Fx(k_i))^T (R_s - Fx(k_i)) - 2\Delta U^T \Phi^T (R_s - Fx(k_i)) + \Delta U^T (\Phi^T \Phi + \bar{R}) \Delta U. \tag{1.14}$$

From the first derivative of the cost function J:

$$\frac{\partial J}{\partial \Delta U} = -2\Phi^T (R_s - Fx(k_i)) + 2(\Phi^T \Phi + \bar{R}) \Delta U, \tag{1.15}$$

the necessary condition of the minimum J is obtained as

$$\frac{\partial J}{\partial \Delta U} = 0,$$

from which we find the optimal solution for the control signal as

$$\boxed{\Delta U = (\Phi^T \Phi + \bar{R})^{-1} \Phi^T (R_s - Fx(k_i)), \qquad (1.16)}$$

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

with the assumption that $(\Phi^T\Phi + \bar{R})^{-1}$ exists. The matrix $(\Phi^T\Phi + \bar{R})^{-1}$ is called the Hessian matrix in the optimization literature. Note that R_s is a data vector that contains the set-point information expressed as

$$R_s = \overbrace{[1\ 1\ 1\ \ldots\ 1]}^{N_p}{}^T r(k_i) = \bar{R}_s r(k_i),$$

where

$$\bar{R}_s = \overbrace{[1\ 1\ 1\ \ldots\ 1]}^{N_p}{}^T .$$

The optimal solution of the control signal is linked to the set-point signal $r(k_i)$ and the state variable $x(k_i)$ via the following equation:

$$\boxed{\Delta U = (\Phi^T\Phi + \bar{R})^{-1}\Phi^T(\bar{R}_s r(k_i) - Fx(k_i)). \qquad (1.17)}$$

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！ 带你导读MPC经典著作！

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！ 带你导读MPC经典著作！

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！带你导读MPC经典著作！

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！ 带你导读MPC经典著作！

《高阶会员专属-第10期》模型预测控制（MPC）要如何使用计算机实现？！ 带你导读MPC经典著作！