I J C T A, 9(41), 2016, pp. 149-155 ISSN: 0974-5572

Analysis of constrained problems using Model Predictive Control

Neha Raghu* and Gurleen Dhillon**

ABSTRACT

This paper explains the concept of linear state space model predictive control algorithm and frequently used operational constraints. It has been applied to constrained water heater problem.

Keywords: Model Predictive Control, Quadratic DMC

1. INTRODUCTION

Model Predictive Control was developed to fulfill the control needs of power plants and petroleum refineries. Later on, MPC'suse was found in other application areas like chemical industries and food processing. MPC has become popular because it can handle constraints and multivariable control problems. The MPC has to consider the limits of physical system being controlled. It is rarely that a physical system does not have boundaries. The performance of control system can deteriorate significantly when the control signals from the original design meet with operational constraints. With a small modification, the degree of performance deterioration can be reduced if the constraints are incorporated in the implementation, leading to the idea of constrained problem[1]

2. MODEL PREDICTIVE CONTROL

2.1. Basic theory [2]

In traditional feedback controllers, control action is adjusted in response to a change in the output set-point of a system. Model predictive control (MPC) is a technique that focuses on constructing controllers that can adjust the control action before a change in the output set-point actually occurs. This predictive ability, when combined with traditional feedback operation, enables acontroller to make adjustments that are smoother and closer to the optimal control action values. Fig. 1 shows the block diagram of model predictive control. A model of the process is used to predict the future evolution of the process to optimize the control signal. At each control interval, MPC algorithm attempts to optimize the future plant behavior by computing a sequence of future manipulated variable adjustments. The first input in the optimal sequence is then sent into the plant and the entire calculation is repeated at subsequent control intervals. The basic MPC controller can be designed with proper restrictions on the prediction horizon and model length. The prediction horizon has to be kept sufficiently larger than the control horizon. The MPC calculations are done in the prediction and controller blocks and are carried out quite often (e.g., every 1-10 minutes). The prediction block predicts the future trajectory of all controlled variables, and the controller achieves the desired response while keeping the process within limits. The targets for the MPC calculations are generated by solving a steady-state optimization problem based on a linear process model, which also finds the best path to achieve the new targets [3]. These calculations may be performed as often as the MPC calculations.

*,** Assistant Professor, School of Electronics and Electrical Engineering, Lovely Professional University, Punjab, *Emails: neha.17822@lpu.co.in, gurleen.18593@lpu.co.in*

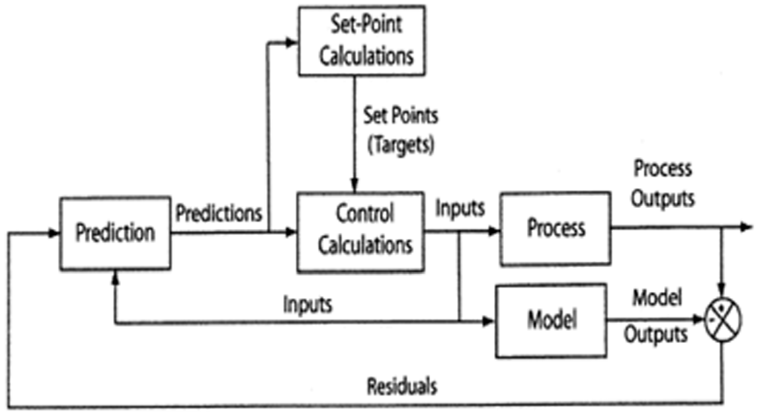

Figure 1: Block diagram for model predictive control [4].

2.2. Objective

MPC is an advanced control technique which is mostly used in discrete-time applications. The overall objectives of MPC are, firstly, to prevent violations of input and output constraints. Second objective is to drive some output variables to their optimal set points while maintaining other outputs within specified ranges. And last, to prevent excessive movement of the input variables [5].

3. OPERATIONAL CONSTRAINTS

There are three major types of operational constraints. The first two types deals with constraints imposed on the control variable u(k), third type deals with constraint on output variable y(k) or state variable x(k) constraints.

3.1. Constraint on rate of change of control variable

These are hard constraints on the size of control signal movements, i.e., on the rate of change of control variables ($\Delta u(k)$). Suppose that for a single input, the upper limit is Δu^{max} and the lower limit is Δu^{min}, the constraints are specified in the form given by eq.1.

$$\Delta u^{min} \leq \Delta u(k) \leq \Delta u^{max} \tag{1}$$

The rate of change constraints can be used to impose directional movement constraints on control variable. If $\Delta u(k)$ can only increase and not decrease, then we can select $0 \leq \Delta u(k) \leq \Delta u^{max}$. The constraint on Δu can be used to cope with the cases where the rate of change of control amplitude is limited in value [1].

3.2. Constraint on amplitude of control variable

These are the physical hard constraints on the system and are most commonly encountered. They are of the form given by eq. 2.

$$u^{min} \leq u(k) \leq u^{max} \tag{2}$$

Here, u(k) is only an incremental variable, not the actual physical variable. These are the most commonly encountered constraints among all the constraint types [1].

3.3. Output Constraints

The actual operating range on the plant output can also be specified. Suppose that the output y(k) has an upper limit y^{max} and a lower limit y^{min}, then the output constraints are specified by eq. 3.

$$y^{min} \leq y(k) \leq y^{max} \tag{3}$$

Output constraints are often implemented as "soft" constraints in the way that a slack variable $s_v > 0$ is added to the constraints, forming

$$y^{min} - s_v \leq y(k) \leq y^{max} + s_v \quad (4)$$

Output constraints often cause large changes in both the control and incremental control variable when they are enforced or become active. When that happens, the control and incremental control variables can violate their own constraints and the problem of constraint conflict occurs. In the situations where the constraints on control variables are more essential to plant operation, output constraints are often relaxed by selecting a larger slack variable, s_v to solve the conflict problem [1].

3.4. Constraints in a MIMO setting

If there is more than one input, then the constraints are specified for each input independently. In the multi-input case, suppose that the constraints are given for the upper limit by eq. 5

$$[\ \Delta u_1^{max}\ \Delta u_2^{max}\ \ldots\ldots\ \Delta u_m^{max}\ ] \quad (5)$$

and lower limits is given by eq.6.

$$[\ \Delta u_1^{min}\ \Delta u_2^{min}\ \ldots\ldots\ \Delta u_m^{min}\ ] \quad (6)$$

Each variable with rate of change is specified as

$$\begin{aligned} \Delta u_1^{min} &\leq \Delta u_1(k) \leq \Delta u_1^{max} \\ \Delta u_2^{min} &\leq \Delta u_2(k) \leq \Delta u_2^{max} \\ \Delta u_m^{min} &\leq \Delta u_m(k) \leq \Delta u_m^{max} \end{aligned} \quad (7)$$

Similarly, suppose that the constraints for the upper limit of control signal are given by eq. 8.

$$[\ u_1^{max}\ u_2^{max}\ \ldots\ldots\ u_m^{max}\ ] \quad (8)$$

and lower limit are given by eq.9.

$$[\ u_1^{min}\ u_2^{min}\ \ldots\ldots\ u_m^{min}\ ] \quad (9)$$

Then the amplitude of each control signal is required to satisfy the constraints

$$\begin{aligned} u_1^{min} &\leq u_1(k) \leq u_1^{max} \\ u_2^{min} &\leq u_2(k) \leq u_2^{max} \\ u_m^{min} &\leq u_m(k) \leq u_m^{max} \end{aligned} \quad (10)$$

4. HARD AND SOFT CONSTRAINTS

Hard constraints are the physical limitations of the real processes e.g. actuator extreme positions.

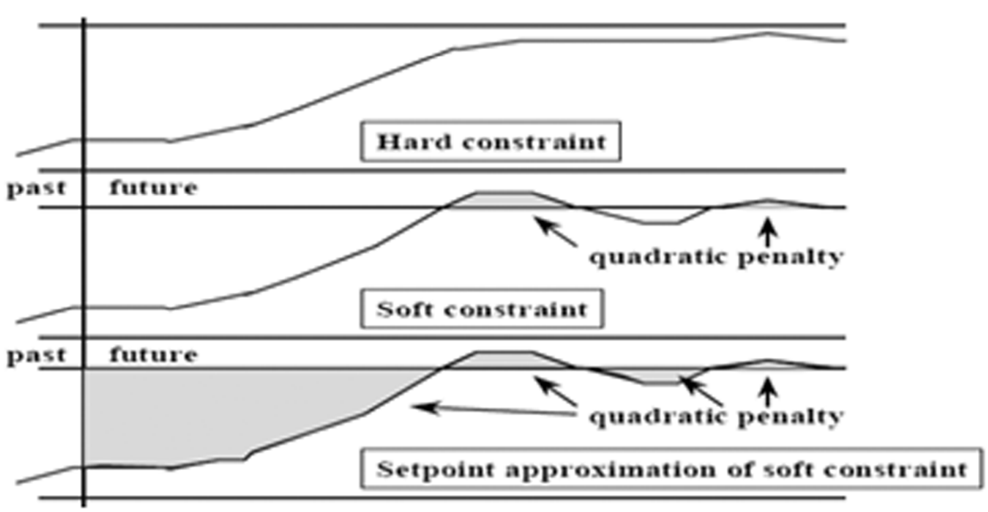

Figure 2: Hard and soft constraints [6]

Hard constraints must not be violated. Soft constraints can be violated though at some penalty e.g. loss of product quality, constraints for the system inner variables. The violation can be penalised in the objective function. Fig. 6.1 shows the hard and soft constraints and also the quadratic penalty for violation of soft constraint.

($\Delta u^{min} \leq \Delta u(k) \leq \Delta u^{max}$) and ($u^{min} \leq u(k) \leq u^{max}$) are the hard constraints.

($y^{min} \leq y(k) \leq y^{max}$) is the soft constraint.

5. QUADRATIC DMC (QDMC)

QDMC considers constraints on the manipulated inputs. To use a standard quadratic program (QP), the input constraints need to be written in terms of the control moves, Δu_{k+i}. since the previously implemented control action (Δu_{k-l}) is known, it can be written in the form of eqn.11.

$$u_k = u_{k-l} + u_k$$
$$u_{k+l} = u_{k-l} + \Delta u_k + \Delta u_{k+l} \tag{11}$$

Since the manipulated input constraints are enforced over the control horizon of M steps, the input constraints and eq.11 yield

$$\begin{bmatrix} u_{\min} \\ u_{\min} \\ \\ u_{\min} \end{bmatrix} \leq \begin{bmatrix} u_{k-1} \\ u_{k-1} \\ \\ u_{k-1} \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & \cdots & 0 \\ & & \cdots & \cdots & \cdots \\ 1 & 1 & 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} \Delta u_k \\ \Delta u_{k+1} \\ \\ \Delta u_{k+M-1} \end{bmatrix} \leq \begin{bmatrix} u_{\max} \\ u_{\max} \\ \\ u_{\max} \end{bmatrix}$$

Most QP codes use a “one-sided” form

$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & \cdots & 0 \\ & & \cdots & \cdots & \cdots \\ 1 & 1 & 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} \Delta u_k \\ \Delta u_{k+1} \\ \\ \Delta u_{k+M-1} \end{bmatrix} \geq \begin{bmatrix} u_{\min} - u_{k-1} \\ u_{\min} - u_{k-1} \\ \\ u_{\min} - u_{k-1} \end{bmatrix} \tag{12a}$$

$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & \cdots & 0 \\ & & \cdots & \cdots & \cdots \\ 1 & 1 & 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} \Delta u_k \\ \Delta u_{k+1} \\ \\ \Delta u_{k+M-1} \end{bmatrix} \geq \begin{bmatrix} u_{k-1} - u_{\max} \\ u_{k-1} - u_{\max} \\ \\ u_{k-1} - u_{\max} \end{bmatrix} \tag{12b}$$

which have the form $A\Delta u_f \geq b$.

The velocity constraints are implemented as bounds on the control moves given by eq.13a.

$$\begin{bmatrix} \Delta u_{\min} \\ \Delta u_{\min} \\ \\ \Delta u_{\min} \end{bmatrix} \leq \begin{bmatrix} \Delta u_k \\ \Delta u_{k+1} \\ \\ \Delta u_{k+M-1} \end{bmatrix} \leq \begin{bmatrix} \Delta u_{\max} \\ \Delta u_{\max} \\ \\ \Delta u_{\max} \end{bmatrix}$$

The majority of constrained MPC problems can be solved based on the input constraints considered above. It is desirable to predict the process outputs to be within a range of minimum and maximum values

$$y_{min} \leq \hat{y}_{k+i}^{c} \leq y_{max} \tag{14}$$

Using matrix-vector notation, the quadratic programming problem is stated by eq. 15.

$$\min \phi = \frac{1}{2}\Delta u_f^T H \Delta u_f + c^T \Delta u_f \tag{15}$$

$$\text{s.t. } A\Delta u_f \geq b$$

$$\Delta u_{min} \leq \Delta u_f \leq \Delta u_{max} \tag{16}$$

where the matrices and vectors in the objective function are given by equations 17 and 18. [5]

$$H = S^T{}_f\, S_f + W \tag{17}$$

$$c^T = E^T S_f \tag{18}$$

6. SIMULATION RESULTS

A temperature control problem has been considered. MATLAB's model predictive toolbox has been used. Default input constraints are applied.

6.1. Problem Definition

The water temperature in a heated boiler is related to the heater power, q, and the ambient air temperature, θ, is related to the heater input power, q, and the ambient air temperature, θ_a, according to the eq. 19.

$$T(d\theta/dt) = kq + \theta a - \theta \tag{19}$$

where it is assumed that T = 1 hour and k = 0.2 °C/kW.

Predictive control is to be applied to keep the water at a desired temperature, and a sampling interval Ts = 0.25 hour is to be used.

Suppose the air temperature follows a sinusoidal variation with amplitude 10°C given by eq. 20.

$$\theta a(t) = 15 + 10 \sin(2\pi t/24) \tag{20}$$

Consider the problem if there is constraint on the input power given by eq.21.

$$-50 < q < 50 \tag{21}$$

6.2. Implementation in MATLAB

Function *scmpc*of MATLAB control toolbox is used. Itdesigns MPC controller for constrained problems. The function usage is as follows:

[y, u, ym] = scmpc(pmod, imod, ywt, uwt, M, P, tend, …r, ulim, ylim, Kest, z, v, w, wu)

pmod - plant model
imod - internal model used as basis for controller design
ywt - weight for setpoint tracking
uwt - weight for change in manipulated variable
ulim - input constraints
ylim - output constraints
z - measurement noise
v - measured disturbance

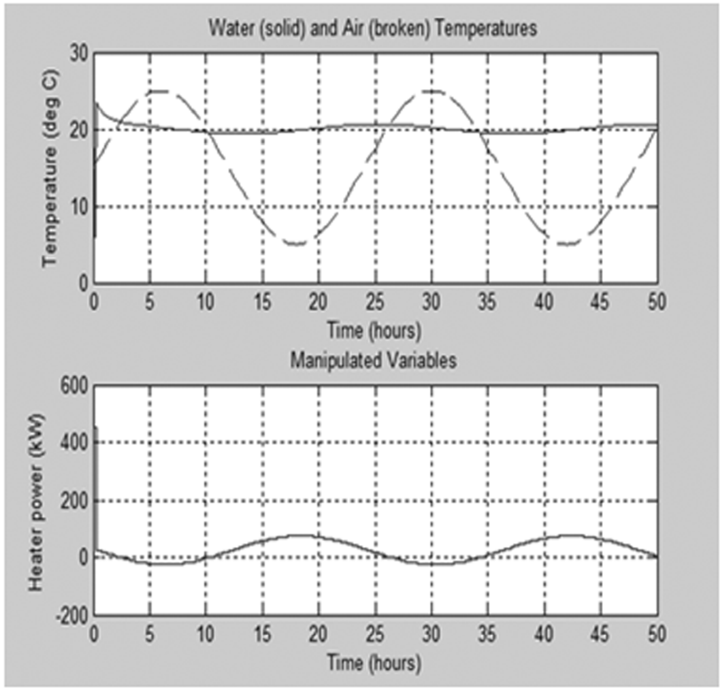

Figure 3: Output of MPC for controlling water temperature.

6.3. Inference

Fig. 3 shows that it is not possible to compensate a sinusoidal variation of air temperature using a standard DMC disturbance model, if air temperature is not measured. If constraints are not present on heater power, then water temperature oscillates with an amplitude of about 0.5 ^{0}C. It is seen that if the air temperature is measured and used for feed forward control, then air temperature is perfectly compensated, providing that model is perfect. If some modeling error occurs, then residual oscillation remains. But if we want to remove

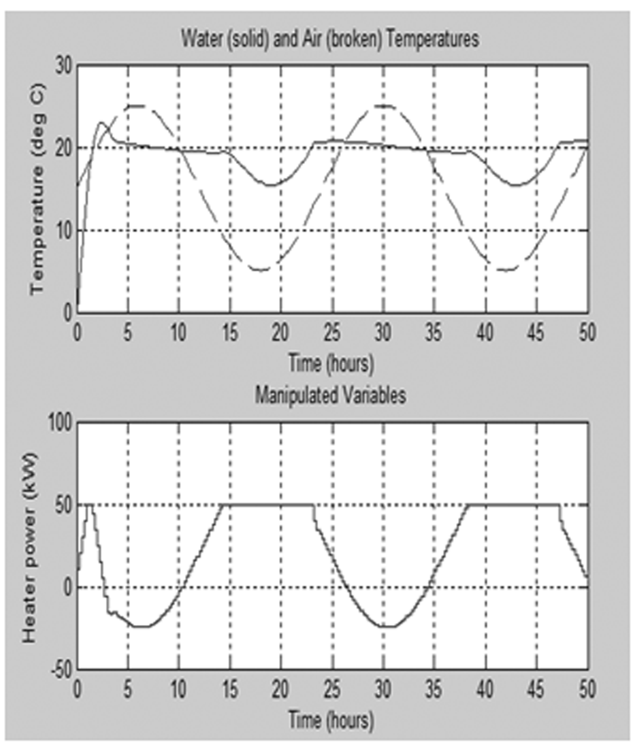

Figure 4: Output after applying MPC to control of water temperature, with constraint on input power.

it completely, then we have to model the sinusoidal disturbance and design a suitable observer, even if the air temperature is not measured.

Fig. 4 shows that water temperature oscillates with amplitude of about 5 0C when there is an input constraint on the heater power. The control action does not exceed 50 kW.

REFERENCES

[1] Wang, L., Model Predictive Control system Design and Implementation using MATLAB, Springer, pp. 43-83, 2009.

[2] Ruchika, Neha Raghu, Analysis of Van de Vusse Reactor using Model Predictive Control, International Journal of Emerging Technology and Advanced Engineering, Vol. 3, Issue 6, June 2013.

[3] Backx, *Reduction and Predictive Control Design for a computational Fluid Dynamic model,* Proceedings of the 41st IEEE conference on Decision and Control, vol. 3, 2002

[4] Seborg, D.E., Edgar, T.F., and Mellichamp, D.A.,Process *Dynamicsand Control*. John Wiley and Sons, Inc., New Jersey, New York, 2004

[5] Bequette, B.W., *Process Control - Modeling, Design, and Simulation.* New Jersey: PHI, 2003, ch.16.

[6] *Qin, S. J., Badgwell, T. A., Industrial Model Predictive Control – An Updated Overview, Survey hand-out, Aspen Tech, 2002*