MODEL PREDICTIVE CONTROL OF NONLINEAR PROCESSES USING ON-LINE SIMULATION

Radu Bălan, Vistrian Mătieş, Olimpiu Hancu

Technical University of Cluj-Napoca, Mechanical Faculty / Mechatronics dep.
B-dul Muncii 103-105 Cluj-Napoca, tel 064-155967 e_mail radubalan@yahoo.com

ABSTRACT

This paper presents a Model Predictive Control algorithm which uses on-line simulation and rule-based control. The basic idea is the on-line simulation of the future behaviour of control system, by using a few control sequences and based on nonlinear analytical model equations. Finally, the simulations are used to obtain the 'optimal' control signal. These issues will be discussed and nonlinear modeling and control of a single-pass, concentric-tube, counter flow heat exchanger will be presented as an example.

KEYWORDS: model-predictive control, nonlinear model, on-line simulation, rule-based control, heat exchanger.

1. INTRODUCTION.

Model Predictive Control (MPC) refers to a class of algorithms that utilize an explicit process model to compute the control signal by minimizing an objective function. When MPC is employed on nonlinear processes, the application of this typical linear controller is limited to relatively small operating regions. The accuracy of the model has significant effect on the performance of the closed loop system. Hence, the capabilities of MPC will degrade as the operating level moves away from its original design level of operation. A solution to avoid these problems is multiple model adaptive control approach (MMAC) which uses a bank of models to capture the possible input-output behavior of processes [3]. In most of these strategies, the controllers are based on linear models with fixed parameters so that the vast body of linear control theory can be applied. Other solutions include the use of a nonlinear analytical model, combinations of linear empirical models or some combination of both.

The performance objective typically penalizes predicted future errors and manipulated variable movement subject to constraints. The ideas appearing in greater or lesser degree in all the predictive control family are basically:

-explicit use of a model to predict the process output in the future;

-on line optimization of a cost objective function over a future horizon;

-receding strategy, so that at each instant, the horizon is displaced towards the future, which involves the application of the first control signal of the sequence calculated at each step.

Performance of MPC could become unacceptable due to a very inaccurate model, thus requiring a more accurate model. This task is an instance of closed-loop identification and adaptive control. Here it is important to remember that the model is only used as an instrument in creating the best combined performance of the controller and the actual system, so the model does not necessarily need to be a good open-loop model of the system. The performance measure should be able to capture as much of the closed loop behavior as possible. Let's consider that it is possible to compute:

- the predictions of system output over a finite horizon (*N*);
- the cost of an objective function,

for each possible sequence: $u(.)=\{u(t),u(t+1), \ldots ,u(t+N)\}$ (1)

and then to choose the first element of the optimal control sequence. For a first look, the advantages of the proposed algorithm include the following:

-the minimum of objective function is global;

-it is not necessary to invert a matrix, so potential difficulties are avoided;

-it can be applied to nonlinear processes if a nonlinear model is available;

-the constraints (linear or nonlinear) can easily be implemented.

The drawback of this scheme is a very long computational time, because there are possibly a lot of sequences. Therefore, the number of sequences must be reduced.

2. CONTROL ALGORITHM

The nonlinear equations of the process can be used directly in the control algorithm. The predictions of system output are calculated by integrating the nonlinear ordinary differential equations of the model over the prediction horizon, by using a few control sequences. For a first stage, are used the next four control sequences:

$$u_1(t)=\{u_{\min},u_{\min},..,u_{\min}\} \quad u_2(t)=\{u_{\max},u_{\min},..,u_{\min}\}$$
$$u_3(t)=\{u_{\min},u_{\max},..,u_{\max}\} \quad u_4(t)=\{u_{\max},u_{\max},..,u_{\max}\} \qquad (2)$$

where u_{min} and u_{max} are the limits of the control signal. There are two pair sequences: ($u_1(t)$, $u_2(t)$) and ($u_3(t)$, $u_4(t)$) which are different through the preponderance of u_{min} or u_{max} in the future control signal. The pair sequences are different only through the first term. In the second stage, depending by the behavior of control system, it is used an algorithm that modifies the limits of control signal:

$$u_{min} \le u_{minst}(t) \le u(t) \le u_{maxst}(t) \le u_{max} \qquad (3)$$
$$\Delta u_{min} \le \Delta u \le \Delta u_{max} \qquad (4)$$

In relations (2), the values of u_{max}, u_{min} are replaced with $u_{minst}(t)$, $u_{minst}(t)$. The control signal is computed using a set of rules based on the extremes (max_0, max_1, min_0, min_1) of the output error of predictions (a_i, $_{i=1..4}$ are predicted errors, d is dead time):

Rule 1: If the sequence $u_1(t)=\{u_{\min},u_{\min},..,u_{\min}\}$ leads to:

$$\max_0 = \max_{d<t<N}\{a_1(t)\} \qquad \max_0 > 0 \qquad (5)$$

and $a_1(d+1)<0$, then $u(t)=u_{minst}(t)$.

Rule 2 : If the sequence $u_2(t)=\{u_{\max},u_{\min},..,u_{\min}\}$ leads to:

$$\max_1 = \max_{d<t<N}\{a_2(t)\} \qquad \max_1 < 0 \qquad (6)$$

then $u(t)=u_{maxst}(t)$.

Rule 3: If the sequence $u_3(t)=\{u_{\min},u_{\max},..,u_{\max}\}$ leads to

$$\min_0 = \min_{d<t<N}\{a_3(t)\} \qquad \min_0 > 0 \qquad (7)$$

then $u(t)=u_{minst}(t)$.

Rule 4: If the sequence $u_4(t) = \{u_{max}, u_{max}, .., u_{max}\}$ leads to:

$$\min_1 = \min_{d<t<N} \{a_4(t)\} \qquad \min_1 < 0 \tag{8}$$

and: $a_4(d+1)>0$, then $u(t)=u_{maxst}(t)$.

Rule 5: In other cases it is used a linear relation:

$$u(t) = \frac{u_{minst}(t)\max_1 - u_{maxst}(t)\min_0}{\max_1 - \min_0} \tag{9}$$

A good behaviour of the control algorithm leads to a prevalence of rule 5. Other rules are used to modify the values of u_{maxst}, u_{minst} and to stabilise the control signal. This algorithm does not address processes where the gain of the process changes sign.

3. EXAMPLE: HEAT EXCHANGER

Heat exchangers are devices that facilitate heat transfer between two or more fluids at different temperatures. Usually, model predictive control (MPC) uses a linear model and an on-line least square algorithm (RLS) to determine the parameters. Heat exchangers are nonlinear processes. To apply the standard MPC algorithms it is possible to use multiple model adaptive control approach (MMAC) which uses a bank of models to capture the possible input-output behavior of processes [3]. Other solutions are based on neural networks and fuzzy logic [4], [5].

In this paper it is used an example from [6]: a heat exchanger with hot fluid - engine oil at 80ºC, cold fluid - water at 20º C, by using a single-pass counter flow concentric-tube. Other data and notations: length (L): 60m, heat transfer coefficients (k_1=1000 W/(m^2 ºC), k_2=80 W/(m^2 ºC)), the temperature profile of fluids and wall ($\theta_1(z,t)$, $\theta_2(z,t), \theta_w(z,t)$), specific heat ($c_1$, c_2, c_w), cross-sectional area for fluids flow and wall (S_1, S_2, S_w), density of fluids and wall (ρ_1, ρ_2, ρ_w), flow speed of fluids (v_1, v_2), transfer area(S) (fig. 1). If physical properties (density, heat capacity, heat transfer coefficients, flow speed) are assumed constant, the heat exchanger model is described using a shell energy balance as:

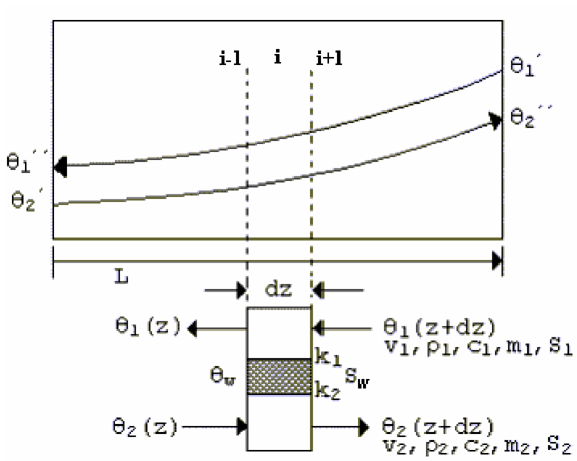

Fig. 1: Temperature distributions

-hot fluid:
$$c_1\rho_1 S_1 \frac{\partial\theta_1(z,t)}{\partial t} - c_1\rho_1 v_1 S_1 \frac{\partial\theta_1(z,t)}{\partial z} = \frac{k_1 S}{L}\left[\theta_w(z,t) - \theta_1(z,t)\right] \tag{10}$$

-cold fluid:
$$c_2\rho_2 S_2 \frac{\partial\theta_2(z,t)}{\partial t} + c_2\rho_2 v_2 S_2 \frac{\partial\theta_1(z,t)}{\partial z} = \frac{k_2 S}{L}\left[\theta_w(z,t) - \theta_2(z,t)\right] \tag{11}$$

-wall:
$$c_w\rho_w S_w \frac{\partial\theta_w(z,t)}{\partial t} = \frac{S}{L}\left[k_1\theta_1(z,t) + k_2\theta_2(z,t) - (k_1+k_2)\theta_w(z,t)\right] \tag{12}$$

Using general notation $\theta_{a(i,j)}$ with a=1 (hot fluid), a=2 (cold fluid), a=w (wall), i , j discrete elements in space respectively time, the discrete equations corresponding to partial differential equation (10),(11),(12) are:

$$\theta_1(i, j+1) = \theta_1(i, j)\left[1 - v_1\frac{\Delta t}{\Delta z} - \frac{k_1 S\Delta t}{Lc_1\rho_1 S_1}\right] + v_1\frac{\Delta t}{\Delta z}\theta_1(i+1, j) + \frac{k_1 S\Delta t}{Lc_1\rho_1 S_1}\theta_w(i, j) \quad (13)$$

$$\theta_2(i, j+1) = \theta_2(i, j)\left[1 + v_2\frac{\Delta t}{\Delta z} - \frac{k_2 S\Delta t}{Lc_2\rho_2 S_2}\right] - v_2\frac{\Delta t}{\Delta z}\theta_2(i+1, j) + \frac{k_2 S\Delta t}{Lc_2\rho_2 S_2}\theta_w(i, j) \quad (14)$$

$$\theta_w(i, j+1) = \theta_w(i, j) + \frac{S\Delta t}{L}\left[k_1\theta_1(i, j) + k_2\theta_2(i, j) + (k_1 + k_2)\theta_w(i, j)\right] \quad (15)$$

In a control application, these equations can not be used directly because v_1 and v_2 are not constant in time. Let's consider next assumptions:

- the speed of fluids is limited:

$$v_{1(min)}<v_1<v_{1(max)};\ v_{2(min)}<v_2<v_{2(max)};\ v_{max} = \max(v_{1(max)}, v_{2(max)}) \quad (16)$$

- the fluids speed is only time-functions:

$$v_1=v_1(t),\ dv_1/dz=0,\ v_2=v_2(t),\ dv_2/dz=0 \quad (17)$$

- the length of heat exchanger is divided in n intervals:

$$L=n\Delta z; \quad (18)$$

- in a time interval Δt, the fluids cover only a part of Δz:

$$n_v v_{max}\Delta t=\Delta z;\ \Delta t < L/(nn_v v_{max}) \quad (19)$$

- two variables Δz_1, Δz_2 are using to totalize the small fluid displacements:

$$\Delta z_1(t+\Delta t)= \Delta z_1(t)+v_1\Delta t; \qquad \Delta z_2(t+\Delta t)= \Delta z_2(t)+v_2\Delta t \quad (20)$$

- in simulations, the displacements of the fluids become effective only if $\Delta z_1>\Delta z$ or/and $\Delta z_2>\Delta z$; in these cases, $\Delta z_1\leftarrow \Delta z_1-\Delta z$ or/and $\Delta z_2\leftarrow \Delta z_2-\Delta z$ (21)

In other words, in simulations, the continue moves of fluids are replaced with small discrete displacements. As a result, the heat exchanger model is described by equations:

$$\theta_1(i, j+1) = \theta_1(i, j)\left[1 - \frac{k_1 S\Delta t}{Lc_1\rho_1 S_1}\right] + \frac{k_1 S\Delta t}{Lc_1\rho_1 S_1}\theta_w(i, j) \quad (22)$$

$$\theta_2(i, j+1) = \theta_2(i, j)\left[1 - \frac{k_2 S\Delta t}{Lc_2\rho_2 S_2}\right] + \frac{k_2 S\Delta t}{Lc_2\rho_2 S_2}\theta_w(i, j) \quad (23)$$

$$\theta_w(i, j+1) = \theta_w(i, j) + \frac{S\Delta t}{L}\left[k_1\theta_1(i, j) + k_2\theta_2(i, j) + (k_1 + k_2)\theta_w(i, j)\right] \quad (24)$$

In a practical implementation, there are used equations (20), (21), (22), (23), (24).

It is important the number and position of temperature sensors. Here, it is considered that only the inlet and outlet temperatures (hot fluid, cold fluid, wall) and the flow rate of fluids are measured. The temperatures inside heat exchanger are estimated. The quality of heat exchange depends especially by the heat transfer coefficients. These parameters depend by temperatures, accumulation of deposits of one kind or another on heat transfer surface, shape of tube, etc. The temperature distributions inside heat exchanger (process and model) are presented in

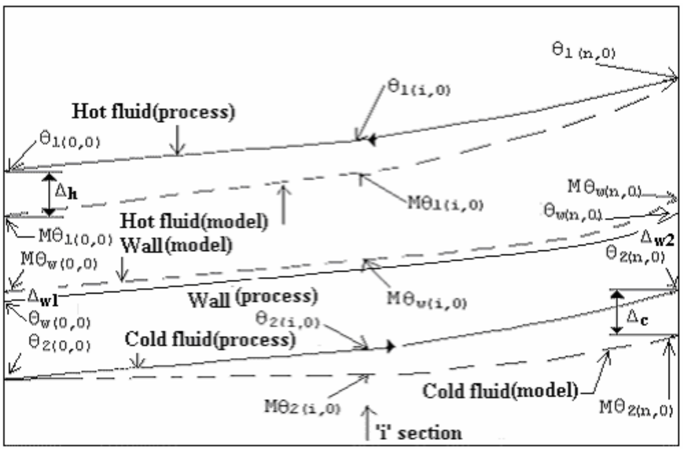

Fig. 2: Process and model - diagrams

fig. 2 using notation $\theta_a(i,j)$. Analogous, the notation $M\theta_a(i,j)$ is used for model. At every sample period, it is possible to compute Δ_h, Δ_c, Δ_{w1}, Δ_{w2}, the temperature prediction errors of outlet hot fluid, outlet cold fluid, wall. These predictions are used to correct the temperature distributions inside the model of heat exchanger, using translations and rotations of distributions. Also, prediction errors can be used to modify the parameters of the model using an algorithm based on rules.

4. EXPERIMENTAL RESULTS

The next applications show the main features of the MPC algorithm. The set point has a variable shape (42°C, 47°C, 52°C, 47°C, 42°C..). The limits of u(t) (hot fluid flow rate) are: $0.05 \le u(t) \le 0.5$ [kg/s]. The flow rate of cold fluid is constant (0.08 kg/s). The temperatures of cold fluid (20°) and hot fluid (80°) are constant. Some experiments with variable flow rate or/and variable temperature of cold fluid is presented in [2]. First, it is used an accurate model (Fig. 3, fig. 4). If the algorithm uses only 1..5 rules, the variance of $u(t)$ will be large. To reduce this variance, a solution is to use a funnel zone for control signal, based on inequality (3). For example, if rule 5 is active then u_{maxst} decreases and u_{minst} increases. Another solution is to limit Δu, using inequality (4). In steady-state regime, control signal is computed using average of past and new values. The algorithm do not uses directly an integral component. In figure 3, steps 50..80, the algorithm tries to reduce the error as fast as possible. As a result, a damped oscillation appears. To avoid this behavior, a solution is to use a reference trajectory.

In figure 5, 6, it is presented an adaptive case; the heat transfer coefficients depend by temperature: $k = k_0(1+\theta/200)$. Initial temperature of cold and hot fluids is 20°. The evolution of the estimations of heat transfer coefficients is presented in figure 7. To obtain these estimations, both rotations and translations of temperature distributions and rule based correction of heat transfer coefficients are used.

In figure 8 it is used the same conditions for heat transfer coefficients; it is

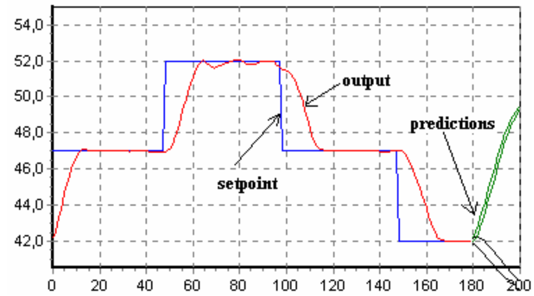

Fig. 3: Setpoint, output (accurate model)

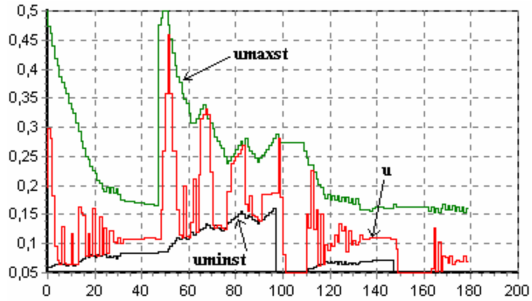

Fig.4:Controller output (accurate model)

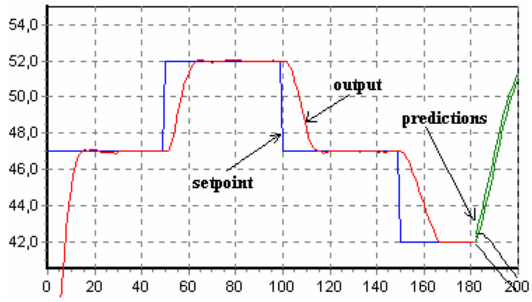

Fig. 5: Setpoint, output (adaptive case)

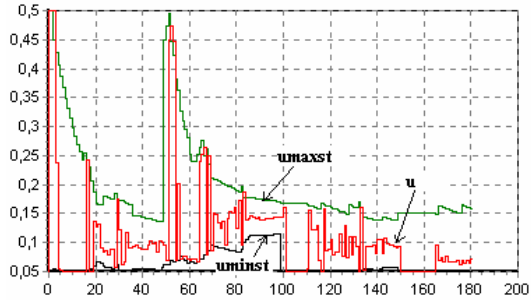

Fig. 6: Controller output (adaptive case)

not used the rotations and translations of temperature distributions. As a result, the quality of control algorithm decreases.

5. CONCLUSIONS

This paper presents the study of a model based predictive control algorithm applied to non-linear processes: the heat transfer in liquid-liquid heat exchangers. The algorithm uses on-line simulation and rule-based control. A non-linear model of the process, based on finite difference method, is used directly in control algorithm. Also, a set of rules is used for parameters estimations and a geometric method is used to correct the temperature distributions inside heat exchanger. The control algorithm is able to maintain better set point tracking performance and disturbance rejection capabilities over the range of nonlinear operation. The proposed approach can be seen as a method for adaptive control of other nonlinear processes.

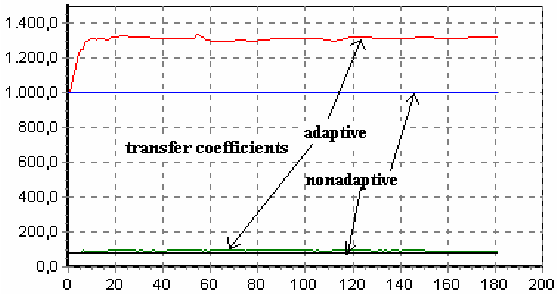

Fig. 7: Parameters identification

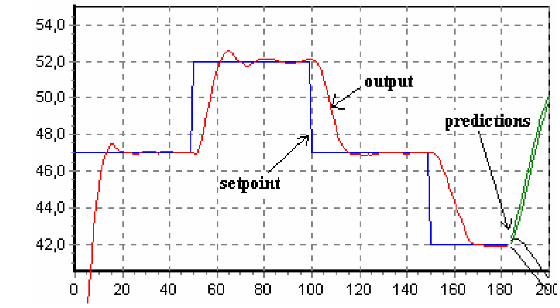

Fig. 8: Setpoint, output (adaptive case)

6. REFERENCES

[1] Camacho E., Bordons C. [1999], “Model Predictive Control” Spriger-Verlag
[2] Balan R. [2001] “Adaptive control systems applied to technological processes”, Ph.D. Thesis 2001, Technical University of Cluj-Napoca, Romania.
[3] Dougherty D., Cooper D., [2003], “A practical multiple model adaptive strategy for single-loop MPC”, Control Engineering Practice 11 (2003)
[4] Fischer M., Nelles O., Fink A., “Adaptive Fuzzy Model Based Control” Journal a, 39(3), Pages 22-28, 1998.
[5] Fink A., Topfer S., Isermann O., “Neuro and Neuro-Fuzzy Identification for Model-based Control”, IFAC Workshop on Advanced Fuzzy/Neural Control, Valencia, Spain, Pages 111-116, 2001.
[6] Ozisik M. N. [1985], “Heat Transfer - A Basic Approach”, McGraw-Hill Book Company
[7] Douglas I.M. [1972], “Process dynamics and control”, Prentice Hall Inc. 1972.