

Nonlinear Model Algorithmic Control of a pH Neutralization Process

ZOU Zhiyun (邹志云)*, YU Meng (于蒙), WANG Zhizhen (王志甄), LIU Xinghong (刘兴红), GUO Yuqing (郭宇晴), ZHANG Fengbo (张风波) and GUO Ning (郭宁)

Research Institute of Pharmaceutical Chemistry, Beijing 102205, China

Abstract Control of pH neutralization processes is challenging in the chemical process industry because of their inherent strong nonlinearity. In this paper, the model algorithmic control (MAC) strategy is extended to nonlinear processes using Hammerstein model that consists of a static nonlinear polynomial function followed in series by a linear impulse response dynamic element. A new nonlinear Hammerstein MAC algorithm (named NLH-MAC) is presented in detail. The simulation control results of a pH neutralization process show that NLH-MAC gives better control performance than linear MAC and the commonly used industrial nonlinear proportional plus integral plus derivative (PID) controller. Further simulation experiment demonstrates that NLH-MAC not only gives good control response, but also possesses good stability and robustness even with large modeling errors.

Keywords model algorithmic control, nonlinear model predictive control, Hammerstein model, pH neutralization process, control simulation

1 INTRODUCTION

Control of pH neutralization processes is a challenging problem and has received considerable attention because of its importance in the chemical process industry. These processes are difficult to control because of their inherent nonlinearity, high sensitivity at or near the neutralization point, and time-varying gains when uncertainties in flows and concentrations of neutralization agents are present. Thus pH neutralization process is considered as a benchmark for testing nonlinear controllers [1–4].

Model algorithmic control (MAC) algorithm is one of the most popular model predictive control (MPC) algorithms widely used in chemical process industries [5–8]. MAC, first developed at the end of 1970s [9, 10], has been widely accepted in industry, mainly by petrochemical industries. Nowadays MAC is something more than an algorithm and a part of its success is due to the fact that the commercial product of MAC covers topics such as model identification and global plant optimization [6].

MAC is characterized by two distinctive features: the use of linear nonparametric impulse response model as its output prediction model, and calculation of control actions through a receding-horizon strategy.

Industrial processes usually involve complex nonlinearities. Linear MPC such as MAC uses linearization models, in which the process operating point is an aid to predicting the response of the controlled process. This strategy is proved acceptable even in controlling mildly nonlinear processes [11, 12]. However, when the degree of nonlinearity is higher, the mismatch between the actual process and the representative model will be greater, resulting in a deterioration of controller performance. Therefore, if the process is highly nonlinear, a nonlinear model will be necessary to describe the process behavior, which results in the

development and application of nonlinear model predictive control (NMPC) strategies. Several nice overviews of NMPC are provided [13–16].

Lei and Datta [17] considered the design and analysis of an adaptive model algorithmic controller based on an internal model structure. This adaptive strategy may be acceptable to mild nonlinear processes, but its on-line model identification and parameter estimation requires too much real-time computation.

Nonlinear input-output block-oriented models, such as Hammerstein model [18–24], do not require much fundamental knowledge about a system, and they are relatively easy to be built using process data. Hammerstein model consists of a static nonlinear element followed in series by a linear dynamic element. It corresponds to processes with linear dynamics but a nonlinear gain, and can adequately represent many of the nonlinearities commonly encountered in industrial processes such as pH neutralization processes. Hammerstein model is particularly useful in representing the nonlinearities of a process without introducing the complications associated with general nonlinear operators [16, 25]. Due to the static nature of its nonlinearity, it can be effectively removed from the control problem, allowing us to get an analytic solution in the unconstrained case or use simple linear programming algorithms in the constrained case.

2 THE NLH-MAC ALGORITHM

The structure of the nonlinear Hammerstein model algorithmic controller (NLH-MAC) is presented in Fig. 1. It is mainly composed of four parts: the Hammerstein model, output prediction, reference trajectory, and calculation of control action. Based on a predictive control performance function, NLH-MAC is to make the predicted output as close to the reference

Received 2012-06-10, accepted 2012-10-17.

* To whom correspondence should be addressed. E-mail: zouzhiyun@sohu.com

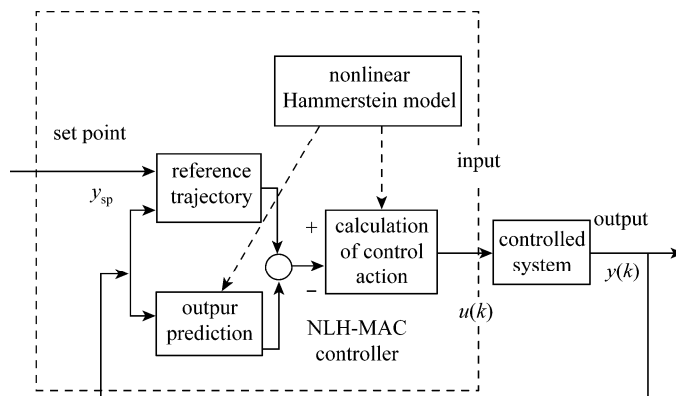


Figure 1 The NLH-MAC system

trajectory as possible, and to reach the set point quickly and smoothly at the same time.

2.1 Nonlinear Hammerstein model

The Hammerstein model consists of a nonlinear static polynomial gain

$$x(k) = \sum_{i=1}^l r_i u^i(k) \quad (1)$$

and a linear dynamic impulse response sub-model sector

$$y_M(k) = \sum_{i=1}^N h_i x(k-i) \quad (2)$$

where l is the order of the nonlinear static gain, r_i is the model parameter of the nonlinear static gain, $u(k)$ is system input, $x(k)$ is an intermediate variable, $y_M(k)$ is the model output, and h_i is the unit impulse response coefficient of the linear dynamic sector. As the impulse response is truncated to N elements, the system is assumed to be stable and causal.

It is assumed that the model order, parameters and coefficients are known. If this is not the case, they can be estimated by the application of system identification methods [26–28].

2.2 Output prediction

The NLH-MAC algorithm predicts the future output of the nonlinear system with its Hammerstein model and past input-output data.

Suppose that at time k , a prediction of the output at time $k+1$ to $k+p$ is needed. In order to avoid the violent fluctuation of system response, the modification of the control action is only limited in m steps ($m \leq p$). Therefore

$$u(k+m-1) = u(k+m) = \dots = u(k+p-1) \quad (3)$$

$$x(k+m-1) = x(k+m) = \dots = x(k+p-1) \quad (4)$$

In order to reduce the effects of model error and immeasurable disturbances on the accuracy of output

prediction, the current prediction error, $d(k) = y(k) - y_M(k)$, is used to modify the model output prediction. When the errors between the real system output $y(k)$ and the Hammerstein model predicted output $y_M(k)$ are constant along the prediction horizon, that is

$$d(k+j|k) = d(k) = y(k) - y_M(k) \quad j=1, 2, \dots, p \quad (5)$$

the predicted output values along the prediction horizon will be

$$\begin{aligned} y_c(k+j|k) &= \sum_{i=1}^N h_i x(k+j-i) + d(k+j|k) \\ &= \sum_{i=1}^j h_i x(k+j-i) + \\ &\quad \sum_{i=j+1}^N h_i x(k+j-i) + d(k+j|k) \end{aligned} \quad j=1, 2, \dots, p \quad (6)$$

where $y_c(k+j|k)$ ($j=1, 2, \dots, p$) is the modified output prediction, which is also called the closed-loop output prediction.

Substituting Eqs. (2), (4) and (5) into Eq. (6) gives

$$y_c(k+j|k) = f_o(k+j|k) + f_r(k+j|k) + d(k+j|k) \quad (7)$$

where $f_r(k+j|k)$ is the free response of the system (the part of the response that does not depend on future control actions) and $f_o(k+j|k)$ is the forced response (the additional component of output response due to the proposed set of future control actions).

The modified output prediction can be written as

$$\mathbf{Y}_C(K) = \mathbf{H}_1 \mathbf{X}_+(K) + \mathbf{H}_2 \mathbf{X}_-(K) + \mathbf{D}(K) \quad (8)$$

where

$$\mathbf{Y}_C(K) = [y_c(k+1|k), y_c(k+2|k), \dots, y_c(k+p|k)]^T$$

$$\mathbf{X}_+(K) = [x(k), x(k+1), \dots, x(k+m-1)]^T$$

$$\mathbf{X}_-(K) = [x(k-N+1), x(k-N+2), \dots, x(k-1)]^T$$

$$\mathbf{D}(K) = [d(k+1|k), d(k+2|k), \dots, d(k+p|k)]^T$$

$$H_1 = \begin{bmatrix} h_1 & 0 & 0 & 0 \\ h_2 & h_1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ h_m & h_{m-1} & \cdots & h_1 \\ \vdots & \vdots & \vdots & \vdots \\ h_p & h_{p-1} & \cdots & h_{p-m+1} \end{bmatrix}$$

$$H_2 = \begin{bmatrix} h_N & h_{N-1} & \cdots & \cdots & \cdots & h_2 \\ 0 & h_N & h_{N-1} & \cdots & \cdots & h_3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & h_N & \cdots & h_{p+1} \end{bmatrix}$$

2.3 Reference trajectory

The goal of industrial process control generally requires that the system output reach the set point quickly and smoothly. Thus the system output should follow a reference trajectory.

Based on the first order approximation, the calculation algorithm of the trajectory is given by

$$y_r(k+j|k) = \alpha^j y(k) + (1-\alpha^j) y_{sp} \quad j=1, 2, \dots, p \quad (9)$$

where $y_r(k+j|k)$ is the output reference trajectory at time $k+j$, y_{sp} is the set point of the output, p is the number of coincidence points, and α is a parameter ($0 < \alpha < 1$). When α is smaller, the output trajectory reaches the set point with less time.

The reference trajectory can also be calculated in an iterative way as follows

$$y_r(k+j|k) = \alpha y_r(k+j-1|k) + (1-\alpha) y_{sp} \quad j=1, 2, \dots, p \quad (10)$$

2.4 Calculation of control action

The objective of an NLH-MAC controller is to drive the output as close to the reference trajectory as possible in a least-squares sense with the possibility of the inclusion of a penalty term as the input moves. The performance function of the NLH-MAC is given as

$$J = \sum_{j=1}^p [y_c(k+j|k) - y_r(k+j|k)]^2 + \lambda \sum_{j=1}^m [x(k+j-1)]^2 \quad (11)$$

where λ is the weighting factor.

The calculation of control action is performed in two steps.

Step1: If there are no constraints, the solution of the cost function J can be obtained analytically by computing the derivate of J and making it equal to

zero, which gives

$$X_+^*(K) = T \{ H_1^T [Y_r(K) - H_2 X_-(K) - D(K)] \} \quad (12)$$

where $T = [H_1^T H_1 + \lambda I]^{-1}$ is a matrix for control action calculation. From Eq. (12), we can obtain the optimal sequence $x^*(k), x^*(k+1), \dots, x^*(k+m-1)$.

Step 2: Using Eq. (1) with $x^*(k+j-1)$ ($j=1, 2, \dots, m$), the optimal control action sequence $u^*(k+j-1)$ ($j=1, 2, \dots, m$) can be derived by calculating the roots of the nonlinear polynomial Eq. (1).

If the order l of the nonlinear polynomial Eq. (1) is odd, a real root of Eq. (1) is always available. We can select an odd order for the nonlinear polynomial sector of the Hammerstein model when modeling. If there are several real roots for Eq. (1), we can choose the one nearest to the last control action in order to prevent excessive movement of control inputs.

Usually only the first element of the optimal control sequence is really sent to the plant. It is not advisable to implement the entire sequence over the next m intervals in automatic succession. Because it is impossible to perfectly estimate the prediction error vector due to model mismatch, so that it is also impossible to anticipate precisely the unavoidable disturbances that cause the actual output to differ from the predictions that are used to compute the sequence of control actions. Furthermore, the reference trajectory can also change over the next m intervals.

If the controlled nonlinear system has a time delay d , the performance function will be modified as

$$J = \sum_{j=1}^p [y_c(k+d+j|k) - y_r(k+d+j|k)]^2 + \sum_{j=1}^m [x(k+j-1)]^2 \quad (13)$$

The optimal control action can be obtained using the similar algorithm for the system without time delay [22, 29].

3 SIMULATION OF pH PROCESS CONTROL

In order to demonstrate the effectiveness of the above NLH-MAC algorithm, some simulation experiments of the NLH-MAC algorithm are carried out for the control of a pilot plant pH process [29] as shown in Fig. 2. Its control results are compared with those from a linear MAC (LMAC) and a nonlinear proportional plus integral plus derivative (NL-PID) controller.

The nonlinear PID controller is composed of a linear PID controller described by

$$u(k) = e(k) + k_I \sum_{i=1}^k e(i) + k_D [e(k) - e(k-1)] \quad (14)$$

and a sector of nonlinear static gain as shown in Fig. 3. In Eq. (14), $e(k)$ is the error, k_I is the integral coefficient and k_D is the derivative coefficient. In Fig. 3, k_1 , k_2 and k_3 are the static gains.

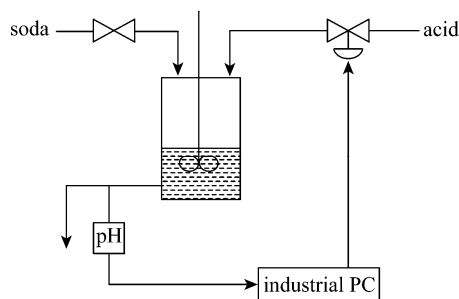


Figure 2 A pilot plant pH process

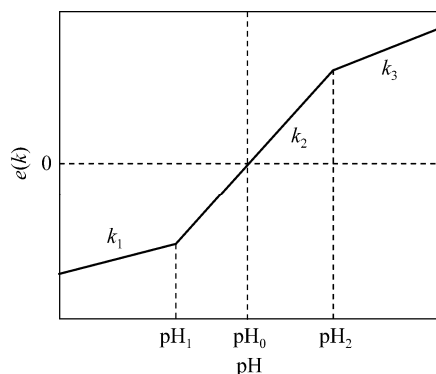


Figure 3 The nonlinear static gains of the NL-PID controller

The Hammerstein model of the pH process [29] is

$$x(k) = u(k) - 1.207u^2(k) + 1.15u^3(k) \quad (15)$$

$$y(k) - 1.558y(k-1) + 0.597y(k-2) = 0.0185x(k-2) + 0.0173x(k-3) + 0.00248x(k-4) \quad (16)$$

where $u(k) = u_m(k) - u_s$, $y(k) = y_m(k) - y_s$, $u_m(k)$ is the electric current to adjust the valve position of the acid flow, and $y_m(k)$ is the pH value of the neutral liquid. $u_s = 2.5$ mA, $y_s = 5.713$ pH, and the sampling period $T_s = 30$ s. Where the subscript s indicates the steady state value, and the subscript m means the measured value.

In simulation, the parameters of NLH-MAC are $\alpha = 0.75$, $\lambda = 0$, $p = 3$, and $m = 1$; the parameters of LMAC are $\alpha = 0.95$, $\lambda = 0$, $p = 3$, and $m = 1$; the parameters of the NL-PID achieved by optimal tuning are $k_1 = 0.15$, $k_D = 0$, $k_I = 0.12$, $k_2 = 0.60$, $k_3 = 0.12$, $pH_1 = 4.8$, and $pH_2 = 7.3$. It needs to mention that the parameter α of LMAC reference trajectory is assigned to 0.95, not the same as that in NLH-MAC. If the value of α is assigned to 0.75 or smaller than 0.95, it will lead the LMAC control system to unstable and its output response will be oscillating. Thus α is assigned to 0.95 for LMAC simulation, in order to get a stable output response to compare with NLH-MAC.

3.1 Set point tracking

For the set point of pH raised from 5.713 to

7.213 and from 5.713 to 7.713, the control results of the NLH-MAC, LMAC and NL-PID are illustrated in Figs. 4 and 5, respectively. Fig. 4 shows that the NLH-MAC controller has a short set point tracking time and a good dynamic output response, LMAC has a sluggish and oscillating output response, and NL-PID controller has a long tracking time and an oscillating output response. Fig. 5 shows that NLH-MAC gives a fast and stable set point tracking output response, LMAC has a long tracking time and an oscillating output response, and NL-PID gives an unstable output response with large oscillation.

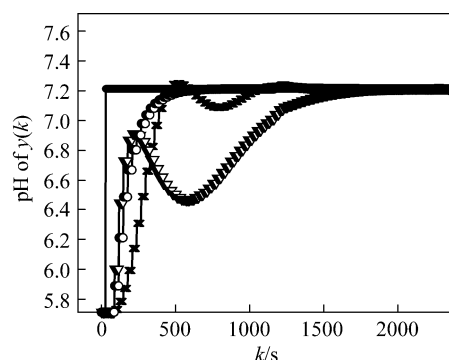


Figure 4 Output response of set point tracking for pH set point from 5.713 to 7.213

● set point; ○ NLH-MAC; ▽ LMAC; × NL-PID

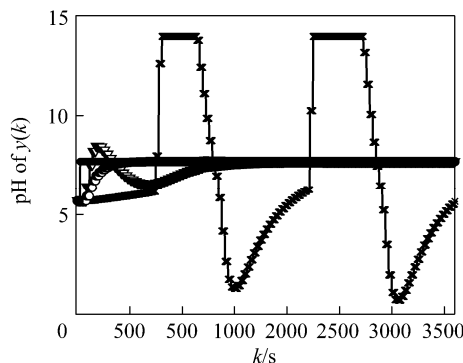


Figure 5 Output response of set point tracking for pH set point from 5.713 to 7.713

● set point; ○ NLH-MAC; ▽ LMAC; × NL-PID

For the pH neutralization process subject to a pulse set point with amplitude from 4.713 to 5.713 and period of 6000 s, the output responses of NLH-MAC, LMAC and NL-PID are presented in Fig. 6. NLH-MAC gives the best pulse set point tracking with stable and fast output response, LMAC and NL-PID have a good response to the positive pulse, while NL-PID has a small over-shoot. Negative pulse set point results in much deterioration of tracking response for LMAC and NL-PID, while LMAC gives a sluggish and oscillating output response, and NL-PID has a much oscillating output with large over-shoot.

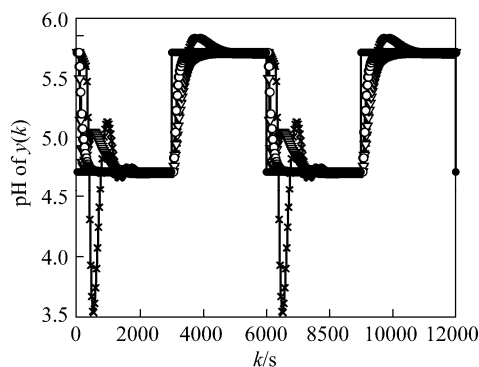


Figure 6 Output response of pulse set point tracking
 ● set point; ○ NLH-MAC; ▽ LMAC; × NL-PID

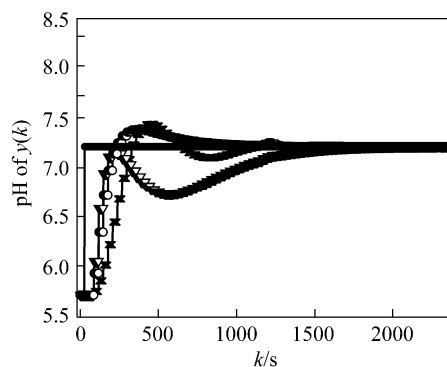


Figure 8 Set point tracking with nonlinear gain mismatch
 ● set point; ○ NLH-MAC; ▽ LMAC; × NL-PID

3.2 Disturbance rejection

When the pH value in the neutralization process is stabilized at 5.713, a deterministic step disturbance signal of amplitude 1 is added to the output of the pH neutralization process. The control result is shown in Fig. 7. The pH output of the NLH-MAC quickly returns to the set point with a short transient response without over-shooting. LMAC returns to its set point with long and oscillating response but no over-shoot. The over-shooting of the NL-PID controller is high, and its transient response time is much longer than that of the NLH-MAC.

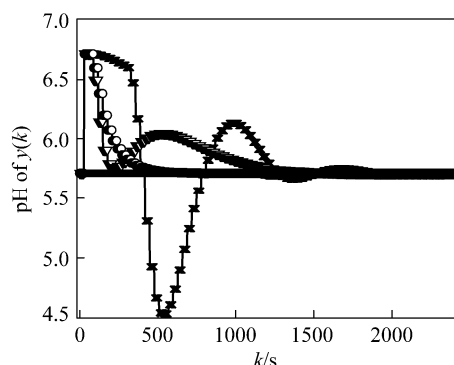


Figure 7 Output response of disturbance rejection
 ● set point; ○ NLH-MAC; ▽ LMAC; × NL-PID

3.3 Model mismatch

In order to demonstrate the robustness of NLH-MAC, simulation of pH process control using NLH-MAC subject to model mismatch in its nonlinear static gains or in its linear dynamic sector is conducted. The control results of set point tracking from 5.713 to 7.213 with nonlinear gain or linear dynamics mismatch are illustrated in Figs. 8 and 9. In Fig. 8, the nonlinear static gain equation for the pH process is changed to $x(k) = 2u(k) - 1.207u^2(k) + 1.15u^3(k)$, so there exists significant model mismatch for the nonlinear gain. In Fig. 9, the linear dynamic sector of the pH process is changed to $2y(k) - 3.116y(k-1) +$

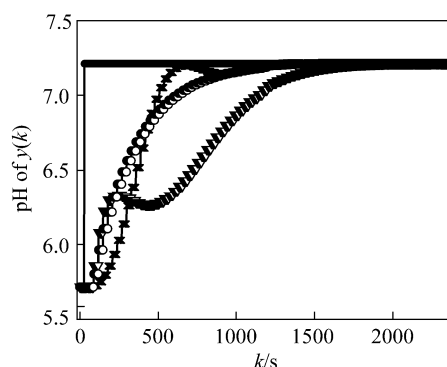


Figure 9 Set point tracking with linear dynamics mismatch
 ● set point; ○ NLH-MAC; ▽ LMAC; × NL-PID

$1.194y(k-2) = 0.0185x(k-2) + 0.0173x(k-3) + 0.00248x(k-4)$, so the coefficients of output sequence $y(k)$, $y(k-1)$ and $y(k-2)$ are doubled. Although there is some deterioration for the set point tracking performance of NLH-MAC due to model mismatch, NLH-MAC still gives better control than LMAC and NL-PID with fewer over-shoots or less oscillations.

4 CONCLUSIONS

The simulation results for the control of pH neutralization process show that NLH-MAC algorithm can make better performance to control nonlinear process than LMAC and NL-PID controller in set point tracking and disturbance rejection. NLH-MAC also gives good control responses even when large model mismatch occurs.

These simulation experiments demonstrate that NLH-MAC is an effective method for the control of nonlinear systems with strong static nonlinear gains. NLH-MAC also possesses good stability and robustness when large modeling errors are present. Moreover, NLH-MAC algorithm is simple and its implementation is convenient and flexible. Thus NLH-MAC algorithm is applicable to a large number of industrial systems with strong static nonlinearities.

Future work will focus on the extension of present

work to constrained multivariable nonlinear systems and extension to open-loop unstable systems.

NOMENCLATURE

$d(k)$	prediction error at step k
$e(k)$	control error at step k
$f_i(k+j k)$	free response of the controlled system
$f_o(k+j k)$	forced response of the controlled system
h_i	unit impulse response coefficients of the linear dynamic sector of Hammerstein model
J	performance function
k	time step
k_1, k_2, k_3	static gains of nonlinear PID controller
k_D	derivative coefficient of nonlinear PID controller
k_I	integral coefficient of nonlinear PID controller
l	order of the nonlinear static gain of Hammerstein model
m	control action steps
N	truncated number of the impulse response of linear dynamic sector of Hammerstein model
p	output prediction horizon
r_i	model parameters of the nonlinear static gain of Hammerstein model
T_s	sampling period
$x(k)$	intermediate variable of Hammerstein model
$y_c(k+j k)$	j step ahead closed-loop output prediction at step k
$y_M(k)$	model output
y_{sp}	set point of the system output
$y_r(k+j k)$	output reference trajectory of time $k+j$ calculated at time k
$u(k)$	system input, mA
$y(k)$	system output, pH
α	parameter of output reference trajectory
λ	weighting factor of intermediate variable

Subscripts

c	closed loop output
M	model
r	reference trajectory

REFERENCES

- Alvarez, H., Londono, C., "pH neutralization process as a benchmark for testing nonlinear controllers", *Ind. Eng. Chem. Res.*, **40**, 2467–2473 (2001).
- Nejati, A., Shahrokhi, M., Mehrabani, A., "Comparison between backstepping and input–output linearization techniques for pH process control", *J. Process Control*, **22** (1), 263–271 (2012).
- Altinten, A., "Generalized predictive control applied to a pH neutralization process", *Comput. Chem. Eng.*, **31** (10), 1199–1204 (2007).
- Bao, Z.J., Pi, D.Y., Sun, Y.X., "Nonlinear model predictive control based on support vector machine with multi-kernel", *Chin. J. Chem. Eng.*, **15** (5), 691–697 (2007).
- Garcia, C.E., Prett, D.M., Morari, M., "Model predictive control: Theory and practice—a survey", *Automatica*, **25** (3), 335–348 (1989).
- Qin, S.J., Badgwell, T.A., "A survey of industrial model predictive control technology", *Control Eng. Practice*, **11** (7), 733–764 (2003).
- Lu, W.X., Zhu, Y., Huang, D.X., Jiang, Y.H., Jin, Y.H., "A new strategy of integrated control and on-line optimization on high-purity distillation process", *Chin. J. Chem. Eng.*, **18** (1), 66–79 (2010).
- Zhao, C., Su, H.Y., Gu, Y., Chu, J., "A pragmatic approach for assessing the economic performance of model predictive control systems and its industrial application", *Chin. J. Chem. Eng.*, **17** (2), 241–250 (2009).
- Rault, J., Richalet, A., Testud, J.L., Papon, J., "Model predictive heuristic control: application to industrial processes", *Automatica*, **14** (5), 413–428 (1978).
- Mehra, R., Rouhai, R., "Model algorithm control: Review and recent developments", In: *Proceedings of Engineering Foundation Conference on Chemical Process Control II*, Sea Island, GA, 287–310 (1982).
- Zhou, M.F., Wang, S.Q., Jin, X.M., Zhang, Q.L., "Iterative learning model predictive control for a class of continuous/batch processes", *Chin. J. Chem. Eng.*, **17** (6), 976–982 (2009).
- Gu, B.F., Gupta, Y.P., "Control of nonlinear processes by using linear model predictive control algorithms", *ISA Transactions*, **47** (2), 211–216 (2008).
- Henson, M.A., "Nonlinear model predictive control: current status and future directions", *Comput. Chem. Eng.*, **23** (2), 187–202 (1998).
- Qin, S., Badgwell, T.A., "An overview of nonlinear model predictive control applications", In *Nonlinear Predictive Control*, Allgower, F., Zheng, A., eds, Birkhauser, 369–393 (2000).
- Morari, M., Lee, J.H., "Model predictive control: Past, present and future", *Comput. Chem. Eng.*, **23** (4–5), 667–682 (1999).
- Manenti, F., "Considerations on nonlinear model predictive control techniques", *Comput. Chem. Eng.*, **35** (11), 2491–2509 (2011).
- Lei, X., Datta, A., "Adaptive model algorithm control", In: *Proceedings of the American Control Conference*, Arlington, VA, 4155–4160 (2001).
- Patikirikorala, T., Wang, L.P., Colman, A., Han, J., "Hammerstein-Wiener nonlinear model based predictive control for relative QoS performance and resource management of software systems", *Control Eng. Practice*, **20** (1), 49–61 (2012).
- Huo, H.B., Zhu, X.J., Hu, W.Q., Tu, H.Y., Li, J., Yang, J., "Nonlinear model predictive control of SOFC based on a Hammerstein model", *Journal of Power Sources*, **185** (1), 338–344 (2008).
- Harnischmacher, G., Marquardt, W., "Nonlinear model predictive control of multivariable processes using block-structured models", *Control Eng. Practice*, **15** (10), 1238–1256 (2007).
- Dolanc, G., Strmčnik, S., "Design of a nonlinear controller based on a piecewise-linear Hammerstein model", *Systems & Control Letters*, **57** (4), 332–339 (2008).
- Fruzzetti, K.P., Palazoğlu, A., McDonald, K.A., "Nonlinear model predictive control using Hammerstein models", *J. Process Control*, **7** (1), 31–41 (1997).
- Man, H., Shao, C., "Neural network predictive control of continuous stirred-tank reactor based on Hammerstein-Wiener model", *CIESC Journal*, **62** (8), 2275–2280 (2011).
- Ławryńczuk, M., "On-line set-point optimisation and predictive control using neural Hammerstein models", *Chemical Engineering Journal*, **166** (1), 269–287 (2011).
- Pearson, R.K., "Selecting nonlinear model structures for computer control", *J. Process Control*, **13** (1), 1–26 (2003).
- Bai, E.W., "A blind approach to the Hammerstein-Wiener model identification", *Automatica*, **38** (6), 967–979 (2002).
- Vanbeylen, L., Pintelon, R., Schoukens, J., "Blind maximum likelihood identification of Hammerstein systems", *Automatica*, **44** (12), 3139–3146 (2008).
- Ding, F., Liu, X.P., Liu, G.J., "Identification methods for Hammerstein nonlinear systems", *Digital Signal Processing*, **21** (2), 215–238 (2011).
- Zou, Z.Y., Liu, G.P., Guo, N., "Predictive control of nonlinear Hammerstein systems and application to pH process", In: *Proceedings of European Control Conference 2003*, Cambridge, UK, 166 (2003).