A Comparative Study of Combined Feedforward/ Feedback Model Predictive Control for Nonlinear Systems

Ji-Zheng Chu,[2] Shi-Shang Jang[1*] and Yu-Nan Chen[1]

[1] Chemical Engineering Department National Tsing-Hua University, Hsin-Chu 30 043 Taiwan
[2] Department of Automation, Beijing University of Chemical Technology, Beijing.

Tighter performance specifications from worldwide competition and ever increasing constraints from environmental and safety considerations give the practical driving force for the development of advanced control technology (Findeisen and Allgöwer, 2001). Large amounts of industrial practice and academic research have made model predictive control (MPC) the de facto standard algorithm for advanced control in the process industry (Nikolaou, 2001). MPC is a general and mathematically feasible scheme to integrate our knowledge about a target, process into controller design and operation, which allows flexible and efficient exploitation of our understanding of a target, and thus optimal performance of a system under various constraints. Three key factors responsible for the great success of MPC are the incorporation of a process model, an algorithm considering plant behaviour over a future horizon in time, and explicit treatment of constraints (Qin and Badgwell, 1999), with the model being foremost and fundamental.

For an accurate and reliable prediction of a model, sufficient information input is the most important factor, though algorithms in the model are equally important regarding the use of the input information. It is well known that combined feedforward plus feedback control can significantly improve performance over simple feedback control whenever a major disturbance exists. It can be measured before it affects the process output, and methods for designing linear feedforward/feedback control systems are well documented in standard textbooks such as that of Stephanopoulos (1984). Such methods are also suitable for nonlinear systems where the effect of measured disturbances (DVs) can be separated from that of manipulated variables (MVs). However, for systems where DVs and MVs are closely coupled together, a unified model correlating both kinds of variables is necessary and should be inverted online. At this time, MPC is a ready solution. Just as pointed out by Economou et al. (1986), MPC has the capability to combine the advantages of open-loop (feedforward) and feedback control. A future horizon in time considered in the MPC algorithm means that the effects of measured and unmeasured disturbances can be predicted and eliminated (Qin and Badgwell, 1999).

Developing a valid model for process dynamics is often the major work in the implementation of an advanced control. More than 75% of the expenditure in an advanced control project normally goes to modelling. Artificial neural networks (ANNs) as a process model for control purpose are superior to other conventional modelling methods for reasons of complexity, accuracy, flexibility, generality, execution speed and cost (Bhat and McAvoy, 1990; MacMurray and Himmelblau, 1995; Hussain, 1995). They have been widely studied in various model-based control strategies. Various types of neural networks have been studied in the literature of process control, and the multilayered feedfor-

* *Author to whom correspondence may be addressed. E-mail address: ssjang@che.nthu.edu.tw*

Model predictive control (MPC) provides a natural framework to realize feedforward and feedback control for nonlinear systems where the effect of disturbances (DVs) cannot be separated from that of manipulated variables (MVs). This study examines the performance of MPC with measured DVs as partial inputs of the model used, which is termed as combined feedforward/feedback MPC (CMPC) in contrast to conventional MPC using a model without input of any measured DV. In the simulation of a pH process, we demonstrate the clear superiority of CMPC over MPC. In the experiment with a bench-scale ethanol and water distillation column, CMPC and MPC using artificial neural network (ANN) models are applied to the dual temperature control problem. External recurrent neural networks (ERNs) with and without a measured DV (feed rate of the column) as their partial input are built and employed in the experiment, with a result that inclusion of the measured DV in the model makes CMPC perform significantly better than MPC. To strengthen practical experience in applying ANN-based MPC, a detailed procedure of the experiment is also documented.

Le contrôle prédictif par modèles (MPC) fournit un cadre naturel pour réaliser la régulation anticipée et asservie de systèmes non linéaires dans lequel l'effet des perturbations (DV) ne peut être séparé des variables manipulées (MV). On examine dans cette étude la performance du MPC avec des DV mesurés comme entrées partielles du modèle ; on appelle ce modèle le MPC anticipé/asservi (CMPC) par contraste avec le MPC conventionnel sans entrée de DV mesuré. Dans la simulation d'un procédé de régulation du pH, nous démontrons la supériorité évidente du CMPC sur le MPC. Dans des expériences de distillation d'éthanol et d'eau menées dans un colonne de laboratoire, le CMPC et le MPC utilisant des modèles à réseaux neuronaux artificiels (ANN) sont appliqués au double problème du contrôle de la température. Des réseaux neuronaux externes récurrents (ERN) avec ou sans DV mesuré (débit d'alimentation de la colonne) comme entrée partielle sont construits et employés dans l'expérience, avec le résultat que l'introduction du DV mesuré dans le modèle permet une performance du CMPC significativement meilleure que celle du MPC. Pour renforcer l'expérience pratique quant à l'application du MPC basé sur les ANN, la description détaillée de l'expérience est également fournie.

Keywords: model predictive control, artificial neural network, external recurrent neural network, distillation, pH process.

ward type is the one most popularly used (Hussain, 1995). Shaw et al. (1997) discussed different structures of ANNs suitable for representing process dynamics. However, Hussain (1995) after reviewing 100 relevant papers on the subject of the application of ANNs to model-based control design, concluded that real successful on-line applications are rare. In his review paper, Hussain summarized 24 experimental studies found in the literature up to 1999. Doherty (1999) also noted that the use of neural networks for process modelling and control is quite rare in industry.

In our previous work (Chu et al., 2003), two kinds of the most commonly used neural networks, feedforward networks (FFNs) and external recurrent networks (ERNs), are examined with experiment and simulation, in the framework of multistep MPC, in order to show the static offset problem of MPC using a FFN model. The experimental results on the dual temperature control problem of a lab-scale distillation column for ethanol and water and on a pilot-scale distillation system for i-butane and n-butane, demonstrate the superiority of ERN based MPC over conventional PI control and linear MPC. In the experiment of MPC on a lab-scale in-line (tubular) pH process, Doherty (1999) used a multi-input and single-output model constructed with a radial basis function neural network (RBFN) that takes disturbances (flow rate and pH value of the inlet waste water) as its partial input, and obtained satisfactory performance in both the setpoint tracking and the disturbance rejecting.

The objective of this paper is to examine the performance of MPC with a measured DV as partial input of the model used, with emphasis on an experimental multi-input and multi-output (MIMO) system and ANN models. In the following context, MPC configured in such a way is termed as combined feedforward/feedback MPC (CMPC), in contrast to MPC where no DV is measured and included into the model explicitly. In the second section, the algorithm of MPC and CMPC is stated in detail. In the third section, a pH process under both MPC and CMPC is simulated, and clear superiority of CMPC over MPC is shown. On this simulated pH process, a highly nonlinear and sensitive system where the contributions of the MV and the DV to the CV are closely coupled, we also demonstrate an unsuccessful attempt to separate the effect of the DV from that of the MV. In the fourth section, CMPC and MPC using ERN models are implemented on a bench-scale ethanol and water distillation column, and their performance is compared for this dual temperature control. In order to strengthen practical experience of application of ANNs as process models, we describe the datasets for training and testing the ERNs in detail. In the last section, some concluding remarks are made.

The reason for using different processes in our simulation and experimental studies is that pH neutralization is a familiar and typical single-input and single-output (SISO) process with strong nonlinear characteristics, and is therefore suitable as a concise and effective supplement to our experimental work on the MIMO distillation column in revealing the effect of including disturbance into the entry of models. The main purpose of this paper is to illustrate the usefulness of including disturbance in the entry of a model built in any suitable framework. The main reason of using ERN models for the distillation column and using first-principle models for the pH process is their ease of implementation and their familiarity to the authors, though the minimal experience in experimental implementation of ANN-based MPC is another reason for using ERN models in the case of the distillation column.

Algorithm of CMPC and MPC

Whether the effects of MVs and DVs are separable is of great significance to the job of modelling, especially with empirical modelling approaches such as ANNs. The data needed for a complete model increases exponentially with the number of coupled variables, which is a problem of the so-called combinatory explosion. For linear systems or nonlinear systems where the effects of MVs and DVs on controlled variables (CVs) are separable, the principle of superposition holds, and the contribution of each MV or DV is modelled separately. In such cases, the combined feedforward/feedback model predictive control (CMPC) can be generally represented in Figure 1. In this figure, vectors *u*, *y*, *d* and *d′* contain the MVs, the CVs, the measured DVs and the unmeasured DVs, *M* is the model of the corresponding process *G*, and M^{-1} is the inverse of *M*. It should be noted that the only requirement of Figure 1 is the superposition principle regardless of whether the system is linear or not.

However, for most nonlinear systems, it is impossible to separate the effects of MVs and DVs on CVs, and CMPC can be expressed in Figure 2. The model has an input array of MVs, measured DVs and CVs predicted by the model itself:

$$y_t^m = f\left(y_{t-1}^m,\cdots,y_{t-k}^m;u_t,\cdots,u_{t-L+1};d_{t-1},\cdots,d_{t-D}\right) \quad (1)$$

where *f* is a nonlinear function in some proper form, *K*, *L* and *D* are the orders of the model with respect to the CVs, the MVs and the measured DVs. For predition into the future, we assume $d_i = d_{t-1}$ for $i \geq t$ with *t* being the current instant of time. The reason for using predicted CVs recurrently in the model instead of the measured ones is that models taking measured CVs as the partial input of the model will result in static offset with MPC (Chu et al., 2003).

In the dotted line box of Figure 2, iteration is performed in every sampling interval to minimize the following objective function:

$$J = \sum_{i=1}^{m}\sum_{p=1}^{P}\left[y_{i,t+P}^{s} - \left(y_{i,t+p}^{m} + h_{i,t}\right)\right]^2 + \sum_{j=1}^{N}\sum_{c=1}^{C} q_{j,c}\Delta u_{j,t+c}^2 \quad (2)$$

through searching for a set of increments for MVs and subjected to constraints such as:

$$P \geq C \quad (3)$$

$$u_{j,t+c} = u_{j,t+C}\left(c > C, j = 1,2,\cdots,N\right) \quad (4)$$

$$\left|\Delta u_{j,t+c}\right| < \Delta u_{j,\max} \quad \left(j = 1,2,\cdots,N; c = 1,2,\cdots,C\right) \quad (5)$$

$$u_{j,\min} \leq u_{j,t+c} \leq u_{j,\max}\left(j = 1,2,\cdots,N\right) \quad (6)$$

$$y_{i,\min} \leq y_{i,t+p}^{m} \leq y_{i,\max}\begin{pmatrix} i = 1,2,\cdots,M; \\ p = 1,2,\cdots,P \end{pmatrix} \quad (7)$$

where *M* = number of CVs, *N* = number of MVs; *P* = length of prediction horizon in time steps; *C* = length of control horizon in time steps; u_j = MV j; = Δu_j = an increment of MV_j, defined as

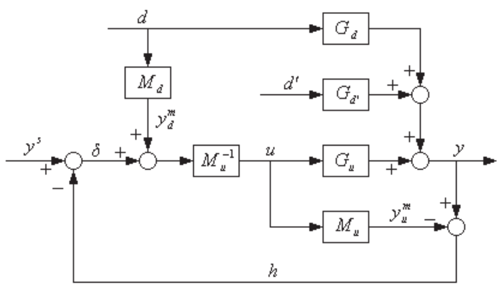


Figure 1. Architecture of CMPC for linear systems and systems where the effects of the DVs and the MVs on the CVs are separable.

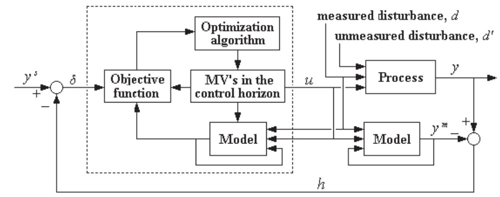


Figure 2. Architecture of CMPC for systems where the effects of the DVs and the MVs on the CVs are not separable.

$$\Delta u_{j,t+c} = u_{j,t+c} - u_{j,t+c-1} \begin{pmatrix} j = 1,2,\cdots,N; \\ c = 1,2,\cdots,C \end{pmatrix}. \tag{8}$$

$\Delta u_{j,\,\max}$ = upper bound for increment of MV j; $u_{j,\,\min}$ = lower bound of MV j; $u_{j,\max}$ = upper bound of MV j; y_i^m = predicted value for CV i as evaluated by the model; y_i^s = setpoint value for CV i; $q_{j,c}$ = weight for MV j in the c-th increment in its control horizon; $h_{i,t}$ = difference between the measured and predicted values of CV i, namely

$$h_{i,t} = y_{i,t} - y_{i,t}^m \tag{9}$$

y_i = measured value for CV i.
In this study, the Levenberg-Marquardt algorithm (Marquardt, 1963), as recommended by Ramchandran and Rhinehart (1995), is adopted in searching for $\Delta u_{j,t+c}$ ($j = 1,2,\cdots$, N and $c = 1, 2, \cdots, C$) that which give a minimum of J as defined in Equation 2. It should also be noted that in the minimization iteration, measured DVs are assumed constant at their current values for the time instants ahead.

MPC has the same algorithm for CMPC as stated above, except that no measured DV enters the model for MPC. Both CMPC and MPC follow the same solution method and possess three attractive properties, namely, dual stability, perfect control and zero offset under assumptions such as perfect model, input-output stability of the model and its inverse, etc., as analyzed by Economou et al. (1986) in the framework of the internal model control (IMC). It is clear from Figure 2 and the objective function in Equation 2 that CMPC provides a feedforward action by including the term y^m, the predicted effect of MVs and measured DVs (if any) on the process through the model, as well as a feedback mechanism by including h, model mismatch and predicted effect of unmeasured DVs on the process. It should also be noted that a feedforward action about unmeasured disturbances also exists in both CMPC and MPC for their prediction horizon into the future. By the way, there are some variants for the objective function in Equation (2) for the purpose of stablility, robustness, and efficient computation, as introduced by Findeisen and Allgöwer (2001). We also noted that the length of prediction horizon (P) has a fundamental influence on the stability of MPC and CMPC in our simulation. In this work, P and C are decided based on some principles such as: short horizons are desirable from a computational point of view, but long horizons are required for closed-loop stability and in order to achieve the desired performance (approximation to the optimality with infinite horizons).

Before getting into specific case studies, it is important to note that constraints are critical to safety and can be very helpful to the performance of a controller. Constraints may stem from safety considerations and physical limits, and also from the requirements of smooth operation, as a measure to improve the behaviour of bad controllers. It takes great expertise to pose suitable constraints for a controller in industry. In this paper, the constraints in all the cases are posed on the upper and lower limits of MVs known from the required operation ranges and are the results of steady state analysis and previous experience of operation, whereas the maximum increment of a MV is set to be approximately one tenth the whole range set by the two limits in the case of the distillation column and the whole operation range for the pH process.

A Simulated pH Process under CMPC and MPC

The pH process is a continuous stirred tank of reactions (CSTR) and has been studied by Palancar et al.(1996, 1998) and by the authors (Tsai et al., 2002). There are two inlet streams to the CSTR, the acid flow, an aqueous solution of acetic acid (AcH) and propionic acid (PrH) with flowrate Q_A and concentrations $C_{AcH,A}$ and $C_{PrH,A}$, and the base flow, an aqueous solution of sodium hydroxide (NaOH) with flowrate Q_B and concentration $C_{NaOH,B}$. The outlet stream has a flowrate $Q \equiv Q_A + Q_B$. In the following context of this section, all the flowrates are in L/s, and concentrations are in mol/L. In this single input and single output system, the MV is Q_B and the CV is the pH value of the system. Material balance about the reactor produces the following equations:

$$Q_A C_{AcH,A} = QC_{AcH} + V\frac{dC_{AcH}}{dt} \tag{10}$$

$$Q_A C_{PrH,A} = QC_{PrH} + V\frac{dC_{PrH}}{dt} \tag{11}$$

$$Q_B C_{NaOH,B} = QC_{NaOH} + V\frac{dC_{NaOH}}{dt} \tag{12}$$

where t is the time in s, C_{AcH}, C_{PrH} and C_{NaOH} are concentrations of components AcH, PrH and NaOH in the reactor, and V = 1.75 L, is the volume of the reactor. Note that the above three equations have an analytical solution one time interval ahead, which is helpful to fast execution of the simulation program. The pH value can be derived from dissociation equilibrium:

$$\frac{C_{AcH}}{1+\dfrac{10^{-pH}}{K_{AcH}}}+\frac{C_{PrH}}{1+\dfrac{10^{-pH}}{K_{PrH}}}+10^{(pH-14)}=C_{NaOH}+10^{-pH} \tag{13}$$

where $K_{AcH} = 10^{-4.75}$, $K_{PrH} = 10^{-4.87}$ at 25°C. At this temperature, the system has an equivalence point around pH = 8.9. For the convenience of statement in the following, we use the functionality pH = *fph* (Q_A, Q_B) to denote the relationship between pH and Q_A, Q_B and *t* in Equations (10) to (13). Note that time *t* has been omitted in the functionality simplicity.

With functionality *fpH* (●) above, three models are fabricated for CMPC and MPC,

$$pH_{m1}=f_{pH}(Q_A,Q_B) \tag{14}$$

$$pH_{m2}=f_{pH}(Q_{A,0},Q_B)+f_{pH}(Q_A,Q_{B,0})-f_{pH}(Q_{A,0},Q_{B,0}) \tag{15}$$

$$pH_{m3}=f_{pH}(Q_{A,0},Q_B) \tag{16}$$

where subscript 0 stands for a steady state at which models PH_{m2} and PH_{m3} are calibrated to be exact. It is seen that the first model (PH_{m1}) is a perfect model in which the contribution of Q_A and Q_B to pH is exactly described, the second model (PH_{m2}) considers the contribution of Q_A and Q_B to pH with two separate terms, and the third model (pH_{m3}) does not include the effect of Q_A on pH.

CMPC as coupled with models pH_{m1} and pH_{m2}, and MPC as coupled with model pH_{m3} are tested in this pH process. To test the real nature of MPC and CMPC, no penalty on the MV (Q_B) increments is used and the constraints are loosely set to be $0 \le QB \le 0.0045$ $|\Delta Q_B| \le 0.0045$ in all the tests. A control horizon $M = 1$ and a prediction horizon $C = 60$ are used. In view of stability, it is favourable that *C* is large enough to include the main transient process of the target according to Findeisen and Allgöwer (2001). Of course, *C* is limited by computation resource. Sampling time has a profound effect on the performance of both MPC and CMPC as shown in Figure 3. For better performance, smaller sampling time should be used. However, sampling time is also restricted by computation resource and by the physical limits of sampling apparatus. In all the tests of this simulation study, the sampling time is chosen to be 5 seconds. The starting point of all the tests is a steady state of $Q_A = Q_B = 0.003$, $C_{AcH,A} = C_{PrH,A} = 0.1$, $C_{NaOH,B} = 0.2$, and pH = 7.0, and models pH_{m2} and pH_{m3} is also calibrated at this state, namely, subscript 0 in Equations 15 and 16 means this state.

The tests are designed to examine the capability of CMPC and MPC as stated above in rejecting disturbances caused by Q_A fluctuating in the following way:

$$Q_A=0.003(1\pm 0.01 d_A) \tag{17}$$

where d_A is set to be 5, 10, 15, or random numbers between 0 and 10. Figures 4, 5 and 6 show the results of three tests at different levels of disturbance. From the curves in these figures, it is clear that CMPC with the perfect model pH_{m1} to account for the effect of the disturbance Q_A, is much superior over MPC using the model pH_{m3} without considering the disturbance and over CMPC using the model pH_{m2} that attempts to separate the effect of the disturbance (Q_A) from that of the MV (Q_B), especially when the disturbance fluctuates continuously. These figures also show, as expected for general nonlinear systems, that the attempt to separate the effect of Q_A from that of Q_B is useless for this pH process. As a well-known fact, separation of effects of DVs and MVs for a nonlinear system often constitutes a meaningful and hard job by itself, and is out of the scope of this paper.

At the end of this section, we should emphasize that this simulation is oversimplified as compared with a real pH process with characteristics such as delays of the control action, valve and pH electrode, asymmetric response of the electrode to positive or negative pH perturbation, etc. This simulated pH process is mainly used to demonstrate the difference of CMPC and MPC, and is not practical enough to be a study on pH control itself. It should also be noted that CMPC is better than MPC only for cases where the test disturbance is measured and included in the model used. For unmodelled perturbations in setpoint, concentration of the basic stream and buffering of the inlet acidic stream, CMPC and MPC will have the same performance.

A Distillation Column under CMPC and MPC

The Column

This test is carried out on a bench-scale distillation column for ethanol and water mixture. The column is depicted in Figure 7 together with the proportional-integral (PI) controllers. Table 1 lists the structural and operational parameters of it. It should be noted that the isotropic mixture (78.15 °C, 0.8943 mole fraction ethanol, at 1 atm) is likely to form at the top of the column.

From measured step response curves, the four classic transfer functions for this column are derived as:

$$G_{11}=\frac{y_1(s)}{u_1(s)}=\frac{-0.0194e^{-30s}}{68.75s+1} \tag{18}$$

$$G_{12}=\frac{y_1(s)}{u_2(s)}=\frac{0.004e^{-166.25s}}{13.75s+1} \tag{19}$$

$$G_{21}=\frac{y_2(s)}{u_1(s)}=\frac{-0.2607e^{-96.25s}}{518.75s+1} \tag{20}$$

$$G_{22}=\frac{y_2(s)}{u_2(s)}=\frac{0.1427e^{-93.75s}}{410s+1} \tag{21}$$

where, u_1 = reflux valve opening, %, u_2 = reboiler heating steam pressure, kPa, y_1 = top temperature, °C, and y_2 = bottom temperature, °C. The relative gain array (RGA) of Bristol (1966) is calculated as:

$$\begin{bmatrix} 1.6043 & -0.6043 \\ -0.6043 & 1.6043 \end{bmatrix} \tag{22}$$

which indicates the strong nonlinear characteristics of the column.

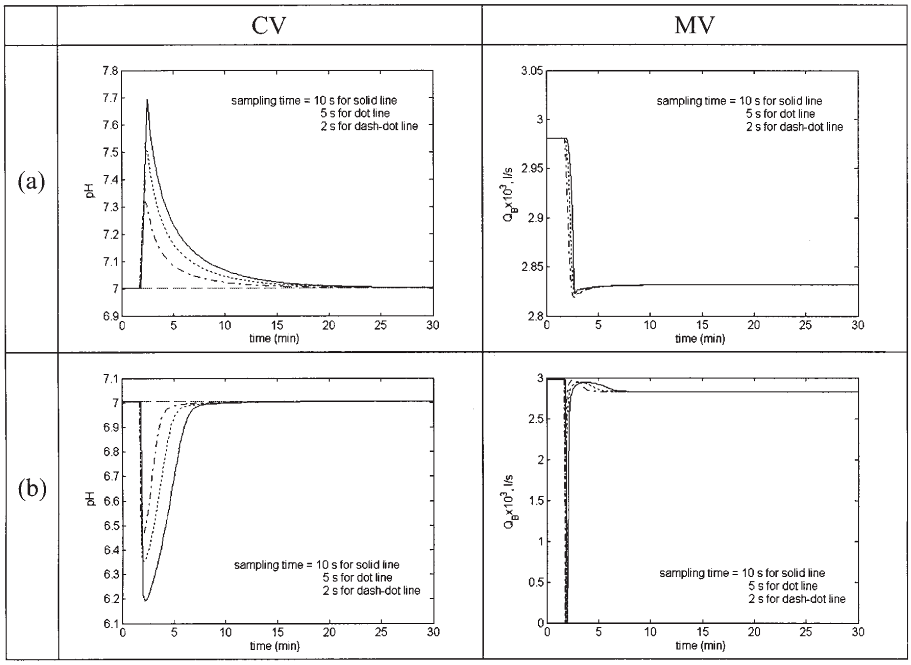


Figure 3. Effect of sampling time on the performance of (a) MPC using model pH_{m3} and (b) CMPC using model pH_{m2} for the simulated pH process subjected to –5% change in the flowrate of the acid stream. The dash line stands for the setpoint.

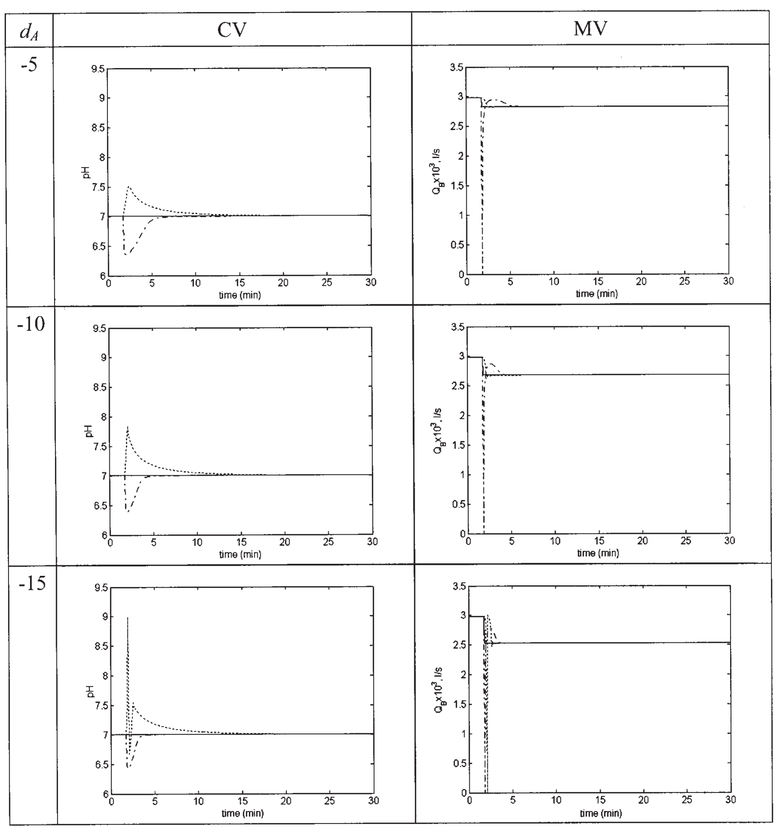


Figure 4. Transient curves of the simulated pH process in response to decreases of the flowrate of the acid stream from 0.003 L/s by 5, 10 and 15%: dash line = setpoint, dot line = MPC using model pH_{m3}, dash-dot line = CMPC using model pH_{m2}, and solid line = CMPC using model pH_{m1}.

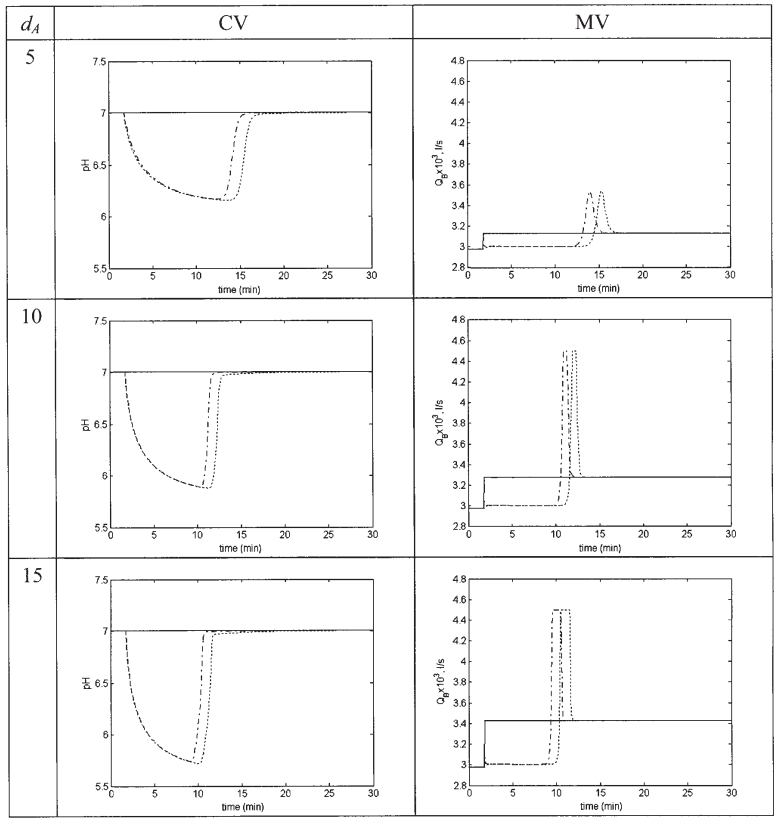


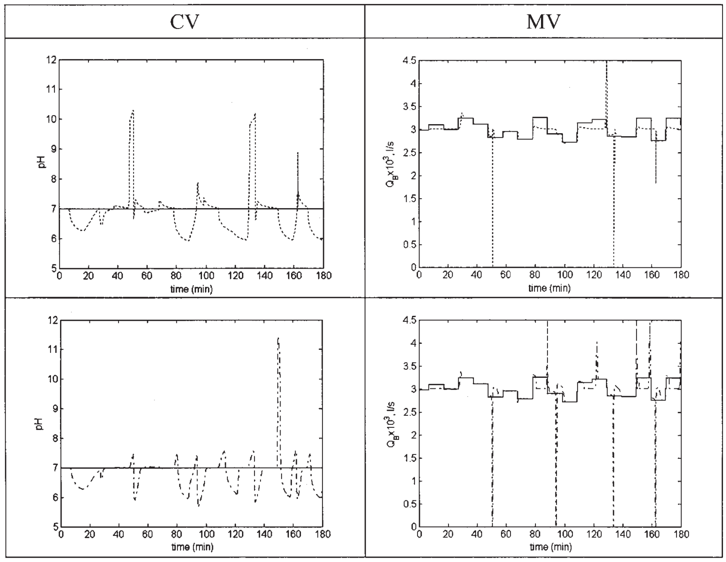
Figure 5. Transient curves of the simulated pH process in response to increases of the flowrate of the acid stream from 0.003 L/s by 5, 10 and 15%: dash line = setpoint, dot line = MPC using model pH_{m3}, dash-dot line = CMPC using model pH_{m2}, and solid line = CMPC using model pH_{m1}.

The Artificial Neural Network Models

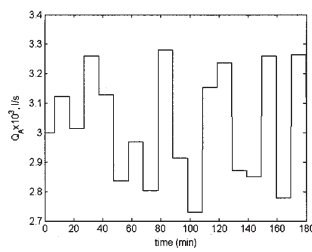
MacMurray and Himmelblau (1995) and Doherty (1999) addressed the issue of output feedback of feedforward networks. The authors (Chu et al., 2003) further observed the phenomenon of steady state offset of multistep MPC using a series-parallel model such as a FFN with measured CVs as its partial input, and accounted for such a phenomenon through mathematical analysis. Therefore, as a model of MPC or CMPC, the neural networks used in this study are the external recurrent neural network (ERN) as shown in Figure 8, which is referred to as a parallel model and can be expressed formally by Equation (1).

It is evident that training and testing datasets are crucial to the neural network models trained from them. In our experience with the distillation column, if neural networks are trained with datasets only from simultaneously changing top reflux and bottom heating steam, oscillatory behaviour will happen with MPC or CMPC using such networks as the model, because neural network models can not distinguish effects of MVs on CVs correctly. For the distillation column in this work, there are two MVs and one measured DV (the feed rate), the following scheme is designed to build the training and testing datasets: change the two MVs (top reflux u_1 and bottom heating steam u_2) sequentially in four patterns, namely (1) both u_1 and u_2 constant, (2) both u_1 and u_2 changing randomly, (3) u_1 constant and u_2 changing randomly, and (4) u_1 changing randomly and u_2 constant. To incorporate the feed rate into the neural networks, in each of the four MV patterns, the feed rate is set at three different levels (270, 300, 330 ml/min). Figures 9 to 11 depict the data in the training dataset. It should be noted that the above scheme for collecting training and testing datasets is necessary for systems such as the pH process shown above and this distillation column, where the effects of MVs and measured DVs are not separable. Otherwise, much less effort is needed by modelling the effects of MVs and measured DVs on the system output separately by independent datasets. Of course, there are other economical schemes for plant testing which help to save the labour in collecting data necessary for a good model, but this is another topic beyond this paper.

Doherty (1999) discussed the difference between FFNs and ERNs, and mentioned that if an ERN is used to predict P steps ahead, it is important to train and validate it over a prediction horizon of P steps. For the distillation column in this work, however, we have found that ERNs from one step ahead training and validation work well, and we do not try multistep training. The conventional error back-propagation algorithm is adopted in training the neural networks. As there are no general rules for determining the structure (number of hidden layers and number of nodes in each hidden layer, etc.) of neural networks, structures of ERNs are fixed in a trial-and-error method in the training stage. In order to avoid over-fitting, the model-validation technique used by Psichogios and Ungar (1991) is adopted, namely, the performance of networks in predicting the testing data is checked after each epoch of learning and the learning process is terminated if the predic-



(a)



(b)

Figure 6. (a) Transient curves of the simulated pH process in response to random fluctuations of the flowrate of the acid stream: dash line = setpoint, dot line = MPC using model pH_{m3}, dash-dot line = CMPC using model pH_{m2}, and solid line = CMPC using model pH_{m1}. (b) Random fluctuations of the flowrate of the acid stream from 0.003 L/s in a range of ±10%.

tion error on the testing data is increased by further training. The structural parameters determined for the final two ERNs (for the top and bottom temperatures respectively) are listed in Table 2. Figure 12 presents the testing results from the two best-trained ERNs, and it is clear that the predicted curves by the two ERNs deviate observably from the measured ones. These two ERNs are to be used in CMPC.

Another two ERNs without the measured DV (feed rate) as their partial input, trained with training and testing dataset collected at the feed rate of 300 ml/min as reported in our previous paper (Chu et al., 2003), are used in MPC for comparison to demonstrate the improvement of CMPC using ERNs with the feed rate as their partial input over MPC using ERNs without input of the feed rate.

At the end of this subsection, it may be useful to mention some details of our practice with network modelling: (1) Training and testing were performed with the toolbox of Matlab (version 6.5, The MathWorks Inc.) in batches, and most default values of the toolbox functions such as learning rate were adopted. In every batch, about 10 networks were set up and trained. At the end of a batch, the best performing network was chosen for further test. Several batches may be necessary to get a suitable network model. For every batch we gave an upper limit number (100, for instance) of epochs. (2) Both the hyperbolic tangent and the sigmoid are common used nonlinear functions in neural networks for process control use (Hussain, 1999). The former was used without much consideration or testing. (3) The input order was initially set to be the time of response, and was further adjusted under the principle of suitable fitting accuracy with as small number of input data as possible. (4) The input and output data were normalized. (5) In the training phase, the identification error was calculated by comparing the measured data with those predicted one step ahead the current instant. Since the prediction was made with measured inputs, the identification error at the end of training was virtually within the measurement accuracy. However, as the trained network was used in the ERN prediction mode where recurrent CVs were used instead of the measured ones, obvious state deviation was observed as shown in Figure 12. Though ERN models have errors in predicting states, they performed well in MPC because they predict correct gain, as shown in our last paper (Chu et al., 2003). By the way, ANN models are able to smooth noise if the training dataset is large enough as shown in Figures 11 and 12.

Performance of CMPC and MPC

In our last paper (Chu et al., 2003), we have shown the performance of MPC based on ERN models without any

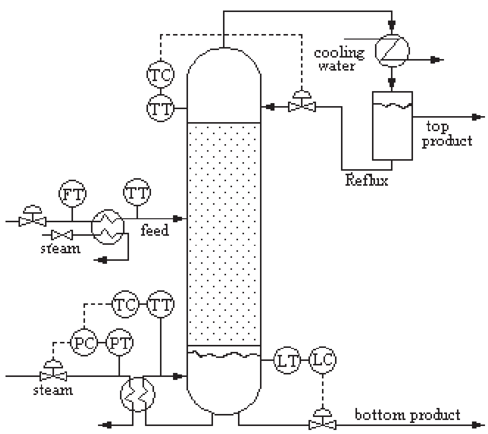


Figure 7. Configuration of the ethanol and water distillation column.

Table 1. Parameters of the ethanol-water column	
pressure	1 atm
inner diameter	0.1 m
packing height	1.1 m
stripping section height	0.9 m
packing porosity	0.968
feed rate	400 mL/min
feed composition	0.30 mole fraction ethanol
reboiler holdup	0.0257 m^3
bottom holdup	0.0263 m^3
reflux drum holdup	0.0135 m^3

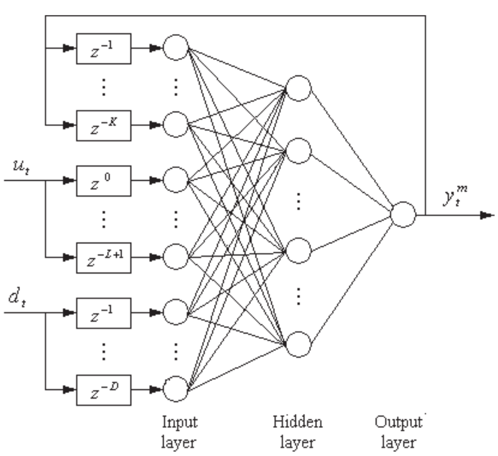


Figure 8. Structure of an external recurrent neural networks (There may be more than one hidden layers).

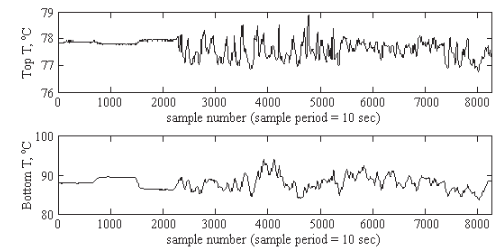


Figure 9. Training data: top and bottom temperatures (CVs) of the ethanol-water column.

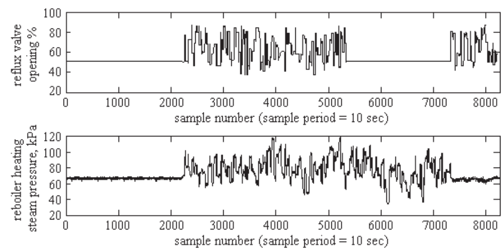


Figure 10. Training data: reflux and reboiler heating steam (MVs) of the ethanol-water column.

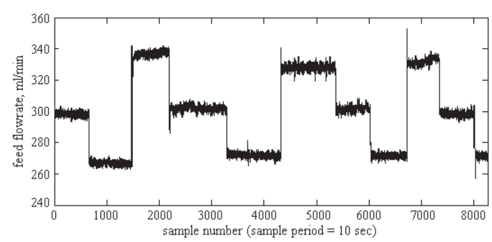


Figure 11. Training data: feed flowrate (the measured disturbance) of the ethanol-water column.

measured DV as their partial input against PI control and linear model predictive control (LMPC), through a series of experiments on this ethanol-water column. The tests of this paper compare the performances of CMPC and MPC using ERNs with and without the feed rate as their partial input. In implementing the ERN-based CMPC and MPC, no penalty on the MVs (u_1 and u_2) is used, namely, all q's in Equation 2 are zero. Control horizon (C) for both MVs is one, whereas the prediction horizon (P) is chosen to be 20. Constraints on minimization includes $35\% \le u_1 \le 95\%$, $|\Delta u_i| \le 6\%$, $55\ \text{kPa} \le u_2 \le 145\ \text{kPa}$, and $|u_2| \le 12$ kPa. Figures 13 to 15 compare the capability of CMPC and MPC to reject the disturbance caused by 7, 10 and 15% decreases in the feed rate of the column, respectively. It is clear from these figures that CMPC performs better than MPC.

Table 2. Structural parameters of the two ERNs with the feedrate as partial input

number of hidden layer	1
number of hidden nodes	3
activation function for hidden nodes	hyperbolic tangent
number of output node	1
transfer function for output node	Linear
entries in input data set 1	$u_1(t)$, …, $u_1(t\text{-}17)$
entries in input data set 2	$u_2(t)$, …, $u_2(t\text{-}17)$

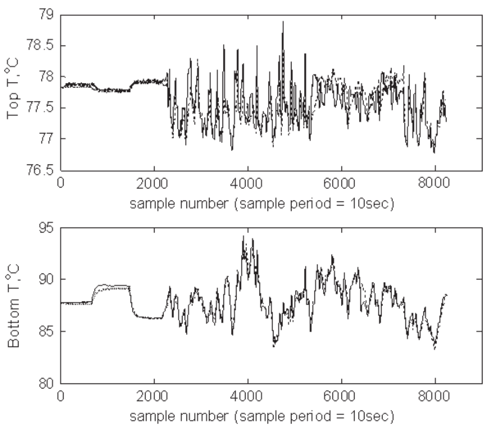


Figure 12. Testing data: top and bottom temperatures of the ethanol-water column, where solid line = measured and dot line = predicted by the two ERNs with the feed rate as their partial input.

Concluding Remarks

The performance of CMPC has been studied in contrast with MPC through simulation and experimentation. In the phase of simulation, the two control schemes are used to a pH process, a highly nonlinear single-input and single-output system. The models used for CMPC and MPC have the same analytical form as the plant. By taking the flowrate of the acid stream as partial input of the model used, CMPC performs much better than MPC that uses a model without the flowrate as one input, for rejecting the disturbance caused by the fluctuation of the flowrate. The simulation results also demonstrate that the attempt to separate the effect of the DV from that of the MV is of little use. In the experiment with a bench-scale distillation column for ethanol and water, training and testing datasets are collected and used to build ERN models correlating CVs, the top and bottom temperatures with MVs, the top reflux and the bottom heating steam as well as a measured DV, the feed rate of the column. The two ERN models used in CMPC have an input window of the feed rate, whereas those two in MPC do not include such a window. The superiority of CMPC over MPC is evident in this dual temperature control problem. The experimental process, especially the part for data collection, is also detailed to enrich practical experience in MPC based on artificial neural networks.

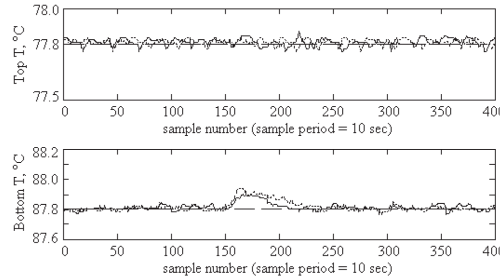


Figure 13. Transient curves of the ethanol-water column in response to a decrease of the feed rate from 300 to 279 mL/min: dash line = setpoint, dot line = MPC, and solid line = CMPC.

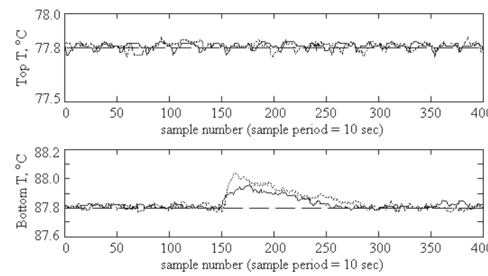


Figure 14. Transient curves of the ethanol-water column in response to a decrease of the feed rate from 300 to 270 mL/min: dash line = setpoint, dot line = MPC, and solid line = CMPC.

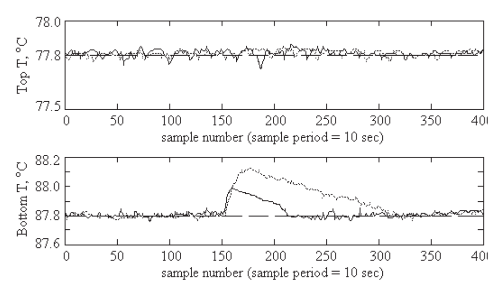


Figure 15. Transient curves of the ethanol-water column in response to a decrease of the feed rate from 300 to 255 mL/min: dash line = setpoint, dot line = MPC, and solid line = CMPC.

Acknowledgements

The authors thank the financial support for this work from National Science Council, Republic of China through the grant NSC90-2622-E007-003.

Nomenclature

C	length of control horizon, in time steps
C	concentration, (mol/L)
d	a measured DV or a vector of measured DVs
d'	an unmeasured DV or a vector of unmeasured DVs
D	order of a model with respect to a DV
G	a transfer function of real process
h	difference between the measured and predicted values of a CV
K	dissociation equilibrium constant
K	order of a model with respect to a CV
L	order of a model with respect to a MV
M	number of CVs
M	a model
N	number of MVs
P	length of prediction horizon, in time steps
q	penalty weight for a MV
Q	flowrate of the outlet flow, L/s
t	time in s or current instant of time
u	a MV or a vector of MVs
V	volume, L
y	a CV or a vector of CVs

Greek Symbol

Δ	increment

Superscripts

m	predicted by a model
s	setpoint

Suberscripts

d	of a measured DV or of a vector of measured DVs
d'	of an unmeasured DV or of a vector of unmeasured DVs
u	of a MV or of a vector of MVs
max	maximum
min	minimum

References

Bhat, N. and T. McAvoy, "Use of Neural Nets for Dynamic Modelling and Control of Chemical Process", Comp. Chem. Eng. **14**, 573–583 (1990).

Bristol, E., "On a New Measure of Interaction for Multivariable Process Control", IEEE Trans. Auto. Control **AC-11**, 133–134 (1966).

Chu . J.-Z, P.-F. Tsai, W.-Y. Tsai, S.-S. Jang, S.-S. Shieh, P.-H. Lin and S.-J. Jiang, "Multistep Model Predictive Control Based on Artificial Neural Networks", Ind. Eng. Chem. Res. **42**, 5215–5228 (2003).

Doherty, S.K., "Control of pH in Chemical Processes Using Artificial Neural Networks", Thesis of Doctor Degree, School of Engineering, Liverpool John Moores University, UK (1999).

Economou, C.G., M. Morari and B.O. Palsson, "Internal Model Control. 5. Extension to Nonlinear Systems", Ind. Eng. Chem. Process Des. Dev. **25**, 403–411 (1986).

Findeisen, R. and F. Allgöwer, "An Introduction to Nonlinear Model Predictive Control", In C.W. Scherer and J.M. Schumacher (editors): Summer school on "The Impact of Optimization in Control", Dutch Institute of Systems and Control (DISC), Delft, The Netherlands (2001), pp. 3.1–3.45.

Hussain, M.A., "Review of the application of neural networks in chemical process control: simulation and on-line implementation", Artificial Intelligence in Engineering **13**, 55–68 (1999).

MacMurray, J.C. and D.M. Himmelblau, "Modelling and Control of a Packed Distillation Column Using Artificial Neural Networks", Comp. Chem. Eng. **19**, 1077–1088 (1995).

Marquardt, D.W., "An Algorithm for Least-squares Estimation of Nonlinear Parameters", J. Soc. Indust. Appl. Math. **11**, 431–441 (1963).

Nikolaou, M., "Model Predictive Controllers: A Critical Synthesis of Theory and Industrial Needs", In K.B. Bischoff, A. Chakraborty, M.M. Denn, N. Peppas, J.H. Seinfeld, G. Stephanopoulos, J. Wei and J. Ying (editors): Advances in Chemical Engineering Series, Academic Press, New York, NY (2001), pp. 132–199 .

Palancar, M.C., J.M. Aragón, J.A. Miguéns and J.S. Torrecilla, "Application of a Model Reference Adaptive Control System to the pH-control. Effects of Lag and Delay Time", Ind. Eng. Chem. Res. **35**, 4100–4110 (1996).

Palancar, M.C., J.M. Aragón and J.S. Torrecilla, "pH-control System Based on Artificial Neural Networks", Ind. Eng. Chem. Res. **37**, 2729–2740 (1998).

Psichogios, D.C. and L.H. Ungar, "Direct and Indirect Model Based Control Using Artificial Neural Networks", Ind. Eng. Chem. Res. **30**, 2564–2573 (1991).

Qin, S.J. and T.A. Badgwell, "An Overview of Nonlinear Model Predictive Control Applications", In F. Allgöwer and A. Zheng (editors): Nonlinear Model Predictive Control, Birkhäuser, Cambridge, MA (1999).

Ramchandran, S. and R.R. Rhinehart, "A very Simple Structure for Neural Network Control of Distillation", J. of Process Control **5**, 115–128 (1995).

Shaw, A.M., F.J. Doyle III and J.S. Schwaber, "A Dynamic Neural Network Approach to Nonlinear Process Modelling", Comp. Chem. Eng. **21**, 371–385 (1997).

Stephanopoulos, G., "Chemical Process Control: An Introduction to Theory and Practice", Prentice-Hall, Inc., Englewood Cliffs NJ (1984).

Tsai, P.-F., J.-Z. Chu, S.-S. Jang and S.-S. Shieh, "Developing a robust model predictive control architecture through regional knowledge analysis of artificial neural networks", J. of Process Control **13**, 423–435 (2002).

Manuscript received October 30, 2003; revised manuscript received June 23, 2004; accepted for publication September 24, 2004.