doi:10.1007/s11633-005-0208-9

International Journal of Automation and Computing 2 (2005) 208-214

High-Order Volterra Model Predictive Control and Its Application to a Nonlinear Polymerisation Process

Yun Li*

Department of Electronics and Electrical Engineering, University of Glasgow, Rankine Building, Glasgow G12 8LT, UK

Hiroshi Kashiwagi

Kumamoto Study Center, The University of the Air, c/o Sojo University, Kumamoto 860-0082, Japan

Abstract: Model Predictive Control (MPC) has recently found wide acceptance in the process industry, but existing design and implementation methods are restricted to linear process models. A chemical process, however, involves severe nonlinearity which cannot be ignored in practice. This paper aims to solve this nonlinear control problem by extending MPC to accommodate nonlinear models. It develops an analytical framework for nonlinear model predictive control (NMPC). It also offers a third-order Volterra series based nonparametric nonlinear modelling technique for NMPC design, which relieves practising engineers from the need for deriving a physical-principles based model first. An on-line realisation technique for implementing NMPC is then developed and applied to a Mitsubishi Chemicals polymerisation reaction process. Results show that this nonlinear MPC technique is feasible and very effective. It considerably outperforms linear and low-order Volterra model based methods. The advantages of the developed approach lie not only in control performance superior to existing NMPC methods, but also in eliminating the need for converting an analytical model and then convert it to a Volterra model obtainable only up to the second order.

Keywords: Model predictive control, Volterra series, process control, nonlinear control.

1 Introduction

Model Predictive Control (MPC) has recently found wide acceptance in industrial applications, particularly in chemical processes where dynamics are relatively slow and hence can accommodate on-line optimisation easily. At any given time, MPC only needs to solve an on-line open-loop optimal control problem over a finite time horizon, where only the first control of the resulting finite control sequence is actually implemented on the plant.

MPC has also received worldwide attention in academia[1,2]. However, much of the work has been confined to linear control based on a linear model or linearised model of the plant. A chemical process for instance, is often severely nonlinear, and this system nonlinearity cannot be ignored in practice. This has stimulated work in synthesising MPC for use with a nonlinear analytical Volterra model and in Volterra series modelling.

Recently, progress in Volterra modelling with a high degree of accuracy has been made in the Kashiwagi Laboratory at Kumamoto University, in which Volterra kernels of up to a 3rd order can now be measured using process I/O data excited with a pseudo-random M-sequence. Based on this, the process can be modelled, and hence predicted more accurately than existing methods that can only obtain Volterra kernels of up to a 2nd order. In place of a Kalman estimator for the linear case, this highly accurate signal reconstruction leads naturally to an advanced nonlinear model predictive control (NMPC) technique[3].

In Section 2, an analytical framework for NMPC is developed based on Volterra models for nonlinear systems. Section 3 provides an on-line realisation technique. NMPC is applied to a Mitsubishi Chemicals' polymerisation reaction process in Section 4. Results show that NMPC is very efficient and effective and considerably outperforms existing methods. Conclusions are drawn in Section 5.

2 Analytical framework for NMPC

2.1 Design objective and exact solution

Fig.1 shows notation for a general model-based control structure. A pre-filter **F** outside the loop is for robust consideration in model-following, and is often a unity gain first-order low-pass with a relatively small time-constant, or a critically damped second-order filter with a relatively high natural frequency. A nonlinear process is represented by **P**. An overall controller **H** is composed of a feed-forward path controller **Q** and

Manuscript received September 4, 2005; revised December 8, 2005.

*Corresponding author. *E-mail address*: Y.Li@elec.gla.ac.uk

a feedback model **G**, all of which are determined in an MPC synthesis.

If we suppose that the nonlinear system **P** has a fading memory[4], as is found in many industrial processes, then its output can be represented with a Volterra series as a temporal extension of a Taylor series expansion as given by:

$$y(t)=\sum_{i=1}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty}p_i(\tau_1,\tau_2,\cdots,\tau_i)u(t-\tau_1)\\ u(t-\tau_2)\cdots u(t-\tau_i)d\tau_1 d\tau_2\cdots d\tau_i+d(t) \quad (1)$$

where p_i is an i-th order Volterra kernel, an i-dimensional impulse response of the nonlinear process. This equation can be compactly rewritten as:

$$y(t)=\sum_{i=1}^{\infty}y^{(i)}(t)+d(t) \quad (2)$$

where

$$y^{(i)}=p_i\,{}^*u \quad (3)$$

is the degree-i Volterra contribution to overall output, and $*$ denotes i-dimensional convolution in the sense of a Volterra series and is hence non-commutative for $i\geqslant 2$.

Similar to other control schemes, the objective $J:\Re\rightarrow\Re^+$ of controller synthesis is to find **Q** such that

$$J=\min_u\|e\| \quad (4)$$

where

$$e=y_R-y \quad (5)$$

is control error to be minimised, and M is the degree of freedom in u that needs to be optimised.

The norm in (4) can be the commonly used L_1, L_2, or L_∞ metric, or any other norm, provided that the metric used guarantees selectivity in minimisation. The metric L_2 is most commonly used in MPC and in optimal control in general, as the optimisation problem posed this way is most easily solved through quadratic programming. However, the L_1 metric and its time-weighted version (i.e. ITAE) offers better selectivity[5].

In realising MPC in practice, J is not measured over an infinite time span[1]. Neither is the control sequence to be optimised. Hence, for on-line realisations the objective is revised to:

$$J=\min_{u(t+1),\cdots,u(t+M)}\left\{\sum_{j=L}^{L+P-1}|y_R(t+j)-y(t+j)|^r\right\}^{1/r} \quad (6)$$

where $L\geqslant 1$ is the minimum number of predictive steps desired, P is the finite length of a "coincidence horizon", M is the finite length of the control sequence within a "control horizon", and $r=1,2,\cdots$, or ∞ is the order of the metric norm.

To formalise the structure of MPC based on an analytical Volterra model, please refer back to Fig.1. Denoting all identified Volterra kernels of **G** using g_i, $i=1,\cdots,N$, (2) can be rewritten as:

$$y=y_M+n=\sum_{i=1}^{\infty}g_i\,{}^*u+n=g\,{}^*u+n. \quad (7)$$

Hence:

$$e=(\xi+n)-(y_M+n)=\xi-y_M=\xi-(g\,{}^*q)^*\xi. \quad (8)$$

The objective of (4) is therefore strictly met if:

$$(g\,{}^*q)^*\xi=\xi,\ \forall\xi. \quad (9)$$

If we reuse capital letters to denote operators corresponding to respective Volterra representations and reuse $*$ to denote a composition of operators (which is usually non-commutative), then (9) becomes:

$$\mathbf{G}*\mathbf{Q}=I. \quad (10)$$

Hence, an optimal solution **Q** is found as $\mathbf{G}^{-1}$, if nonlinear Internal Model Control (IMC) can be "exactly" realised.

However, as observed in [5], obtaining a strictly zero J of (4), or strictly satisfying (9), is impossible under

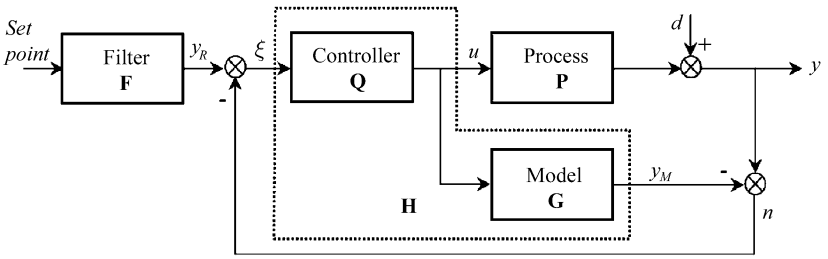

Fig.1 A canonical structure of an MPC scheme

practical constraints (such as voltage or current limits), as otherwise it would mean that controller "gain" has to be infinite for all frequencies and time. This is also because strictly satisfying (9) or (10) implies that MPC will reduce in effect to open-loop control, which in turn cannot always guarantee (9) or (4) if disturbance d exists.

2.2 Controller synthesis and second-order realisation

The usefulness of a Volterra model is that it decomposes a nonlinear system into an additive series of a linear model plus nonlinear models of different degrees of nonlinearity, as shown in (2). Now, we rewrite $\mathbf{G}$ and $\mathbf{Q}$ as a linear operator (based on a first-order Volterra kernel) and a nonlinear one based on higher-order Volterra kernels:

$$\mathbf{G} = \mathbf{G}_1 + (\mathbf{G}_2 + \mathbf{G}_3 + \cdots) \tag{11}$$

$$\mathbf{Q} = \mathbf{Q}_1 + (\mathbf{Q}_2 + \mathbf{Q}_3 + \cdots). \tag{12}$$

Then the controller is synthesised by solving (10) with the above equations [6], as derived in Chapter 7 of [7]:

$$\mathbf{Q}_1 = \mathbf{G}_1^{-1} \tag{13}$$

$$\mathbf{Q}_2 = -\mathbf{G}_1^{-1} * \mathbf{G}_2 * \mathbf{Q}_1 \tag{14}$$

$$\mathbf{Q}_3 = -\mathbf{G}_1^{-1} * \{\mathbf{G}_2 * (\mathbf{Q}_1 + \mathbf{Q}_2) - \mathbf{G}_2 * \mathbf{Q}_1 - \mathbf{G}_2 * \mathbf{Q}_2 + \mathbf{G}_3 * \mathbf{Q}_1\} \tag{15}$$

noting that because of nonlinearity,

$$\mathbf{G}_2 * (\mathbf{Q}_1 + \mathbf{Q}_2) \neq \mathbf{G}_2 * \mathbf{Q}_1 + \mathbf{G}_2 * \mathbf{Q}_2. \tag{16}$$

This result is useful in that (13) is an exact IMC controller only for the linear part, $\mathbf{G}_1$ (degree-1 contribution from the Volterra model), and in that the synthesis of all components of $\mathbf{Q}$ involves the inverse of only this linear component, and is dependent only upon known components in $\mathbf{Q}$.

However, a problem arises if the inverse of $\mathbf{G}_1$ either does not exist or is unrealisable. To resolve these issues, [6] proposed a "generalised inverse" synthesis method by first reverting back to (4) for the relaxed objective of a minimal J, as opposed to a strict zero J or (10), and then by revising objective (4) to:

$$J_i = \min_{Q} \|e^{(i)}\|, \ \forall i \tag{17}$$

where

$$e = \sum_{i=1}^{\infty} e^{(i)} \tag{18}$$

$$e^{(1)} = \mathbf{H}_1[\xi] - \xi \tag{19}$$

$$e^{(j)} = \mathbf{H}_j[\xi], \ \text{for } j = 2, 3, \cdots \tag{20}$$

and

$$\mathbf{H} = \mathbf{G} * \mathbf{Q} = \mathbf{H}_1 + \mathbf{H}_2 + \mathbf{H}_3 + \cdots. \tag{21}$$

If the Euclidian metric (i.e. the L_2 norm) is used to measure revised objective (17), the optimisation problem for the case $i = 1$ will have the following solution[6]:

$$u^{(1)} = \mathbf{G}_1^{-L}[\xi] \tag{22}$$

i.e.

$$\mathbf{Q}_1 = \mathbf{G}_1^{-L} \tag{23}$$

where the operator $\mathbf{G}_1^{-L}$ is the "left inverse" of $\mathbf{G}_1$ defined by:

$$\mathbf{G}_1^{-L} * \mathbf{G}_1 = \mathbf{I}. \tag{24}$$

This is now a much relaxed inverse linear controller compared with $\mathbf{Q}_1 = \mathbf{G}_1^{-1}$ for the exact inverse.

Similarly, higher-degree solutions can be derived as given by [6]:

$$u^{(j)} = -\mathbf{G}_1^{-L}[\xi^{(j)}], \ j = 2, 3, \cdots \tag{25}$$

where $\xi^{(j)}$ is a quantity dependent only upon known quantities up to Step j.

To realise the algorithm, [6] derived a second-order Volterra controller structure as shown in Fig.2. They reported successful case studies in the process control

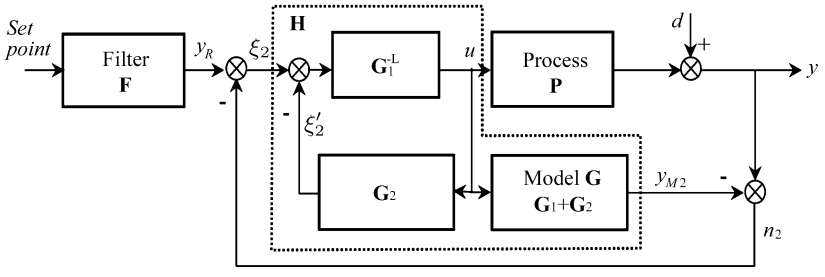

Fig.2 A second-order Volterra controller

of the benchmark van de Vusse reactor and in a continuous stirred tank reactor using this second-order realisation. However, their Volterra model can only be identified up to a second-order via bilinear Carlemann "linearization" applied to a first-principles model. This means that a first-principles model is needed and the NMPC controller can offer only one nonlinear kernel, which consequently does not unleash the full potential of a Volterra controller.

2.3 Third-order structure

Volterra kernels up to a 3rd order can now be obtained directly from I/O data by exciting a process with a pseudo-random M-sequence[8], which will be shown in Section 3. This leads to improved modelling accuracy and hence to the ability to unleash greater potential in NMPC.

Here we show that the structure of a second-order Volterra controller based on the "generalised inverse" of [6] can be extended to a third-order.

Theorem. The structure of a third-order Volterra controller based on the "generalised inverse" takes the same underlying form as a second-order one and is given in Fig.3.

Proof. The theorem is proved if we can show that the control signal u in Fig.3 is the same as that in Fig.2.

Third-order structure:

$$y_{M3} = (\mathbf{G}_1 + \mathbf{G}_2 + \mathbf{G}_3)[u] \tag{26}$$

$$\begin{aligned}\xi_3 &= y_R - (y - y_{M3}) \\ &= y_R - y + (\mathbf{G}_1 + \mathbf{G}_2 + \mathbf{G}_3)[u]\end{aligned} \tag{27}$$

$$\xi_3' = (\mathbf{G}_2 + \mathbf{G}_3)[u]. \tag{28}$$

By definition of the "addition" of operators,

$$\begin{aligned}\xi_3 - \xi_3' &= y_R - y + \mathbf{G}_1[u] + (\mathbf{G}_2 + \mathbf{G}_3)[u] \\ &\quad - (\mathbf{G}_2 + \mathbf{G}_3)[u] = y_R - y + \mathbf{G}_1[u]\end{aligned} \tag{29}$$

$$\begin{aligned}\therefore \quad u &= \mathbf{G}_1^{-L}[\xi_3 - \xi_3'] \\ &= \mathbf{G}_1^{-L}[y_R - y + \mathbf{G}_1[u]].\end{aligned} \tag{30}$$

Second-order structure:

$$y_{M2} = (\mathbf{G}_1 + \mathbf{G}_2)[u] \tag{31}$$

$$\begin{aligned}\xi_2 &= y_R - (y - y_{M2}) \\ &= y_R - y + (\mathbf{G}_1 + \mathbf{G}_2)[u]\end{aligned} \tag{32}$$

$$\xi_2' = \mathbf{G}_2[u] \tag{33}$$

$$\begin{aligned}\xi_2 - \xi_2' &= y_R - y + \mathbf{G}_1[u] + \mathbf{G}_2[u] - \mathbf{G}_2[u] \\ &= y_R - y + \mathbf{G}_1[u]\end{aligned} \tag{34}$$

$$\begin{aligned}\therefore \quad u &= \mathbf{G}_1^{-L}[\xi_2 - \xi_2'] \\ &= \mathbf{G}_1^{-L}[y_R - y + \mathbf{G}_1[u]].\end{aligned} \tag{35}$$

So both control signals are indeed the same. □

Corollary. The essence of (30) and (35) is to force $y = y_R$ through IMC, so that objective (4) or (6) is met.

Proof. This yields by noting that $\mathbf{G}_1^{-L}$ is the "left inverse" of $\mathbf{G}_1$ and both operators are linear. □

3 On-line predictive realisation

To relieve application engineers from needing to obtain and hence linearise a first-principle based nonlinear model, an NMPC system could be realised directly in discrete-time using (7) and process I/O data. Also, using this method, a third-order Volterra controller does not lead to more complication in its realisation than a second-order one, as long as the third-order kernel can be measured.

Realising a Volterra controller directly in discrete-time is similar to a special category of MPC: Dynamic Matrix Control (DMC)[1], where a linear Kalman estimator is replaced by nonlinear Volterra kernel based signal reconstruction. Such an on-line realization of MPC requires only modest computational power and the process dynamics are often relatively slow. At any given time, only the first control of the resulting solution to the on-line open-loop optimal control problem is actually implemented in plant[9].

Direct discrete-time realisation of MPC has recently gained increasing popularity in process control, as it

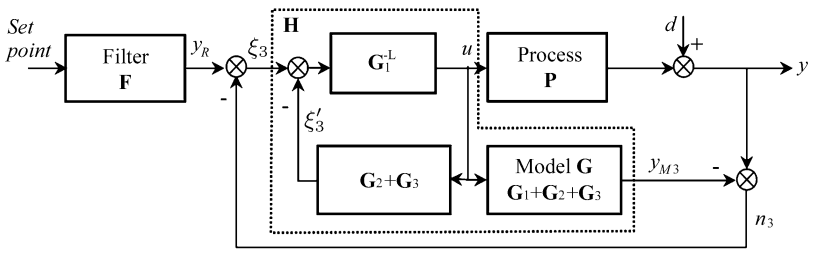

Fig.3 A third-order Volterra controller

requires that a simple quadratic program be solved online only over finite horizons P and M, as depicted in Fig.4. Here the set-point is R and the reference trajectory for the process to follow is y_R. Referring back to Fig.1, y_R is the response of pre-filter $\mathbf{F}$ to the step command of amplitude R. Given a typical first-order low-pass pre-filter:

$$F(s) = \frac{1}{1+\tau s} \tag{36}$$

the reference signal in the continuous domain is represented by:

$$y_R(t) = R(1 - e^{-t/\tau}). \tag{37}$$

It is not difficult to derive the discrete-time difference equation version, as given by:

$$y_R(t+1) = \alpha y_R(t) + (1-\alpha)R \tag{38}$$

where with a given sampling interval T,

$$\alpha = 1 - \frac{T}{\tau}. \tag{39}$$

Iterating the first-order difference equation yields:

$$\begin{aligned} y_R(t+2) &= \alpha y_R(t+1) + (1-\alpha)R \\ &= \alpha^2 y_R(t) + (1-\alpha^2)R \end{aligned} \tag{40}$$

and, in general,

$$y_R(t+j) = \alpha^j y_R(t) + (1-\alpha^j)R. \tag{41}$$

Hence, the reference trajectory can be calculated j steps ahead.

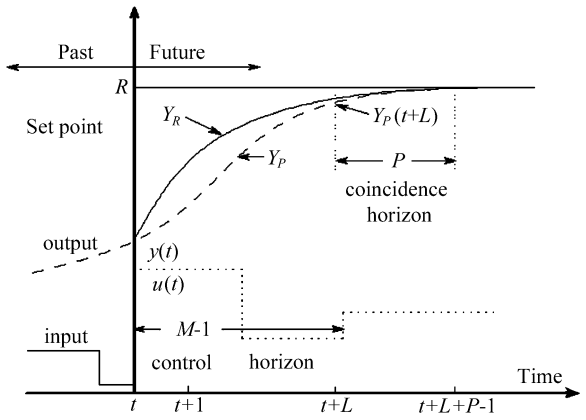

Fig.4 Direct discrete-time realisation of MPC

At time t, process output can also be predicted j steps ahead using:

$$y_P(t+j) = y_P(t) + [y_M(t+j) - y_M(t)] \tag{42}$$

where y_M is calculated using Volterra kernels up to a third-order, as given by (7) for $i = 1, 2$ and 3.

The revised objective of a linear MPC case[1] is also applicable to a nonlinear one, given the assumption that the nonlinear system $\mathbf{P}$ has a fading memory, and is hence suitable for Volterra representation[4]. Under this assumption, the objective of (4) is enforced within a "coincidence horizon" of a finite length P. Hence, the task is to find a control sequence within a "control horizon" of a finite length M, such that:

$$J = \min_{u(t+1),\cdots,u(t+M)} \sum_{j=L}^{L+P-1} \{y_R(t+j) - y_P(t+j)\}^2 \tag{43}$$

where $L \geqslant 1$ is the minimum prediction step needed.

At the end of the optimisation, only the first control signal $u(t+1)$ is implemented. Similar to [6], the Euclidian metric is used here. However, the objective is not limited to the use of this norm, as other norms can also be used as long as the optimisation algorithm to be used can accommodate these norms. Note that in the optimisation process we need to be able to search for the control sequence under constraints such as actuator saturation. This can limit the choice of a suitable optimisation algorithm.

However, since the control signal will go through a D/A converter the resolution of u will be finite. Hence we can search for a discrete value of Δu for each of the elements in the M-element control sequence $\boldsymbol{u} \in \Re^M$. The easiest optimisation algorithm that can accommodate the constraints is an a-posteriori nondeterministic hill-climbing algorithm, i.e. hill-climbing guided by trail-and-error, not by a gradient that may not exist or exist accurately due to sampled noisy data. Such an algorithm is outlined in Fig.5.

Now the only task left in devising an accurate Volterra model based predictive controller is to obtain Volterra kernels in (7) for $i = 1, 2$ and 3. For details of this, please refer to [3].

4 Application to a polymerisation process

The NMPC method developed here is applied to a process control problem of Mitsubishi Chemical Corp. Their chemical reactor is described by the differential equations:

$$\begin{cases} \dfrac{dx_1}{dt} = \dfrac{1}{Tp_1}(-x_1 + Kp_1 u_1) \\ \dfrac{dx_2}{dt} = \dfrac{1}{Tp_2}(Kp_2 x_1 x_2 - x_2 + Kp_3 u_2) \\ y = x_2 \end{cases} \tag{44}$$

with initial conditions:

$$\begin{cases} x_1 = 0.02\text{kgh}^{-1}, & x_2 = 5.0\text{kgcm}^{-2} \\ u_1 = 0.05\text{kgh}^{-1}, & u_2 = 3195\text{kgh}^{-1} \end{cases} \tag{45}$$

where x_1 is the consumption velocity of a catalyst, x_2 is gas density, u_1 is the supply quantity of the catalyst, u_2 is the supply quantity of polyethylene, and Tp_1, Tp_2, Kp_1, Kp_2 and Kp_3 are constants. Here, the control input to optimise is u_1, and the output to control is x_2, which are required to follow a step change with reference to $R = 10$ kg cm^{-2} from 5.

For this, an M-sequence with amplitude ± 0.025 and characteristic polynomial $f(x) = 260577$ in octal notation is applied to the reactor with a sampling period of 0.3 h. Taking cross-correlations between this excitation signal and corresponding plant output, Volterra kernels can be measured. The search for a control sequence was carried out within a range of ± 0.05 with a 0.001 increment. Figs.6 and 7 present control performance and control signal results. From these results, it can be seen that the NMPC formulated from a third-order Volterra model offers the best closed-loop performance. In addition the control signal predicts well, and hence results in minimal chatter.

5 Conclusion

MPC has recently found wide acceptance in the process industry, but existing design and implementation methods have been restricted to linear process models. As severe nonlinearity inherent in, for example, a chemical process, cannot be ignored in practice, this paper has developed an analytical framework for NMPC. It is based on a Volterra model for nonlinear systems, which has been successfully extended to a third-order kernel.

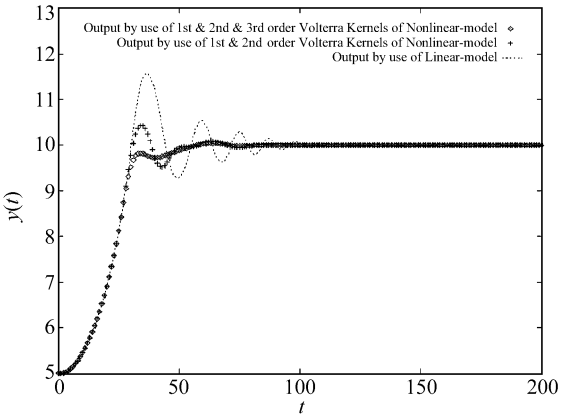

Fig.6 Performance of the NMPC controllers with first, second, and third-order Volterra models

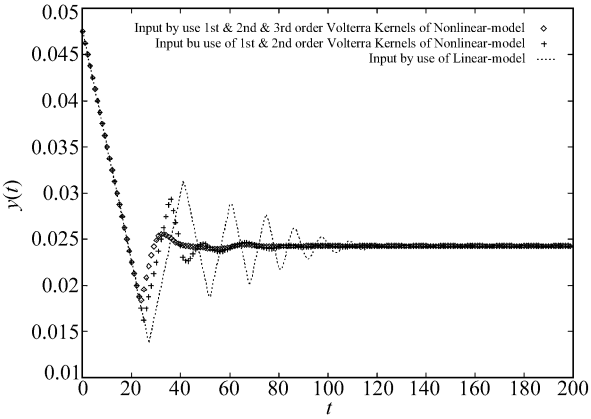

Fig.7 Control signals resulting from models with three different levels of nonlinearity

```
public hillclimber() {
  declaration and initialisation;
  for (i=0; i<Number_of_repetitions_required; i++) {
    local= FALSE;
    use a known or existing u_c as current (or generate it at random);
    evaluate u_c by J(u_c);   // calculate constrained J
    do {
      perturb u_c to generate N new u sequences;
      // with constraints on u
      evaluate all N new sets;
      select the sequence u_i that has the smallest J;
      if (J(u_i) < J(u_c))
        u_c = u_i; // this means that improvement is possible
      else
        local = TRUE; // no improvements found
      } while (!local ) // continue if local not found
      output u_c;
    }
    return; // completed a prescribed number of trials
  }
```

Fig.5 An a-posteriori nondeterministic hill-climbing algorithm

With an on-line realisation technique, NMPC has been applied to a Mitsubishi Chemical' polymerisation reaction process. Results show that this nonlinear MPC technique is feasible and very effective. It considerably outperforms linear and low-order Volterra model based methods. The advantages of the approach developed lie not only in control performance superior to existing NMPC methods, but also in relieving practising engineers from the need to derive an analytical model, and then convert it to a Volterra model in which a model can only be obtained up to a second order.

References

[1] C. E. Garcia, D. M. Prett, M. Morari. Model predictive control: Theory and practice - A survey. *Automatica*, vol. 25, no. 3, pp. 335–348, 1989.

[2] D. Q. Mayne, J. B. Rawlings, C. V. Rao, P. O. M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[3] H. Kashiwagi, Y. Li. Nonparametric nonlinear model predictive control. *Korean Journal of Chemical Engineering*, vol. 21, no. 2, pp. 329–337, 2004.

[4] S. Boyd, L. O. Chua. Fading memory and the problem of approximating nonlinear operators with Volterra series. *IEEE Transactions on Circuits and Systems*, vol. 32, no. 11, pp. 1150-1161, 1985.

[5] Y. Li, K. H. Ang, G. Chong, W. Feng, K. C. Tan, H. Kashiwagi. CAutoCSD - Evolutionary Search and Optimisation Enabled Computer Automated Control System Design. *International Journal of Automation and Computing*, vol. 1, no. 1, pp. 76–88, 2004.

[6] F. J. Doyle III, B. A. Ogunnaike, R. K. Pearson. Nonlinear Model-based Control Using Second-order Volterra Models. *Automatica*, vol. 31, no. 5, pp. 697–714, 1995.

[7] M. Schetzen. *The Volterra and Wiener Theories of Nonlinear Systems*, Wiley, New York, 1980.

[8] H. Kashiwagi, Y. Sun. Identification of Nonlinear System by Use of Volterra Kernel. *Transactions Society of Instrumentation and Control Engineers*, vol. 31, no. 8, pp. 1054–1060, 1995.

[9] Y. H. Kim, I. J. Sohn. Minimum deviation control of a binary distillation column using reduced order dynamic matrix. *Journal of Chemical Engineering of Japan*, vol. 19, no. 5, pp. 501–509, 1993.

Yun Li is a senior lecturer at University of Glasgow, UK, where has taught and researched in evolutionary computation and control engineering since 1991. He worked in the UK National Engineering Laboratory and Industrial Systems and Control Ltd, Glasgow in 1989 and 1990. In 1998, he established the IEEE CACSD Evolutionary Computation Working Group and the European Network of Excellence in Evolutionary Computing (EvoNet) Workgroup on Systems, Control, and Drives. In summer 2002, he served as a visiting professor to Kumamoto University, Japan. He is also a visiting professor at University of Electronic Science and Technology of China. His research interests are in parallel processing, design automation and discovery of engineering systems using evolutionary learning and intelligent search techniques. Applications include control, system modelling and prediction, circuit design, microwave engineering, and operations management. He has advised 12 Ph.D.s in evolutionary computation and has 140 publications.

Hiroshi Kashiwagi received B.E, M.E. and Ph.D. degrees in measurement and control engineering from the University of Tokyo, Japan, in 1962, 1964 and 1967 respectively. In 1967 he became an Associate Professor and in 1976 a Professor at Kumamoto University. From 1973 to 1974, he served as a visiting Associate Professor at Purdue University, Indiana, USA. From 1990 to 1994, he was the Director at Computer Center of Kumamoto University. He has also served as a member of Board of Trustees of Society of Instrument and Control Engineers (SICE), Japan, Chairman of Kyushu Branch of SICE and General Chair of many international conferences held in Japan, Korea, Chin and India. In 1994, he was awarded SICE Fellow for his contributions to the field of measurement and control engineering through his various academic activities. He also received the Gold Medal Prize at ICAUTO'95 held in India. In 1997, he received the "Best Book Award" from SICE for his new book entitled "M-sequence and its application" written in Japanese and published in 1996 by Shoukoudou Publishing Co. in Japan. In 1999, he received the "Best Paper Award" from SICE for his paper "M-transform and its application to system identification". His research interests include signal processing and applications, especially pseudorandom sequence and its applications to measurement and control engineering.