On-line Multivariable Identification and Control of Chemical Processes Using Canonical Variate Analysis

Andreas H. Kemna, Wallace E. Larimore¹, Dale. E. Seborg*, and Duncan A. Mellichamp*

Weyerhaeuser Company, Tacoma, WA 98447; akemna@wdi.com
¹ Adaptec, Inc, 40 Fairchild Drive, Reading, MA 01867; adaptec@mcmiall.com
*Department of Chemical and Nuclear Engineering
University of California, Santa Barbara, CA 93106
seborg@engrhub.ucsb.edu, dmellich@halbut.ucsb.edu

1 Abstract

In this paper, an on-line identification and control strategy is developed for chemical processes. The procedure used for system identification is canonical variate analysis (CVA), a method that optimizes on-line identification and control strategies by minimizing the error between the actual and predicted values of the output variables. The resulting models are then used to control the process by minimizing the error between the actual and predicted values of the output variables.

2 System Identification and Adaptation to Change

The procedure used for system identification is the canonical variate analysis (CVA) method (Larimore, 1990). The CVA method involves computation of the nonnegative decomposition (SNVD) to determine the linear combinations of the past data with predictive information for the future, i.e., states of the system. The computations are entirely automatic and have been shown to approach the theoretical optimum accuracy of maximum likelihood (ML) while avoiding a nonlinear optimization scheme.

A fundamental problem in on-line identification and adaptive control is the determination of the model to the changes in the actual process variables. Previous approaches to the adaptation problem have been largely heuristic using such procedures as exponential averaging. Only recently has more fundamental approach been taken using concepts of information. A recent study (Larimore, 1986) approaches the general problem of decisions involving model state order, structure, rate of adaptation, and detection of abrupt changes using an entropy or information measure in a predictive inference framework. The entropy measure has been given a fundamental statistical inference justification in terms of the basic statistical principles of sufficiency and repeated sampling in a predictive setting (Larimore, 1985).

This approach provides a sound theoretical basis for the development of adaptive identification procedures.

Consider a piece-wise constant model with constant dynamics governing y1 at the j+1 time interval T_j = (t_j ≤ t < t_{j+1}) where y1 is the first point of the j+1 interval and T_j is the corresponding set of observations Y_j = {y1, ..., y(t_{j+1})}. Thus consider the model of the form (1) and (2) for t ∈ T_j with piece-wise constant parameters θ:

\[ x_{t+1} = \Phi^t x_t + G^t u_t + w_t \]  
\[ y_t = H^t x_t + D^t u_t + B^t w_t + v_t \]

\[ \theta^T = \theta^T \]  
\[ \theta^T \]  
\[ \theta^T \]  

which are constant over an interval of time T_j and change from one time interval to another.

The problem of adaptation to slow variations is primarily that of determining the lengths of time intervals to use in the time varying model (1) and (2). The predictive inference and entropy methods provide a means of objectively comparing the vast multitude of such hypotheses entailed in the selection of interval length for the adaptive time series analysis problem.

Consider the division of a section of data into h subintervals Y_j for 1 ≤ j ≤ h. The set S = {Y_1, ..., Y_h} of subintervals can be associated with the set of states (t_1, ..., t_h) where t_i specifies the first point of the interval Y_i. Then the various hypotheses can be considered such as H_k, divide the interval into subintervals Y_j.

For each subinterval Y_j, for j = 1, 2, ..., h, suppose a state space model with parameters denoted by \( \theta^j \) is fitted using the CVA method with the Akaike information criterion (AIC) used to select the best model state order.

By successive application of the Markov property, the joint probability density of the observations conditioned on the initial state x_0 is given by

\[ \log p(Y_1, ..., Y_h | x_0, \theta^j) = \sum_{j=1}^{h} \log p(Y_j | Y_{j-1}, \theta^j) \]  

where \( \theta^j = (\theta^T, ..., \theta^T) \) is the parameter vector for the composite model consisting of all of the CVA models fitted over each of the h subintervals. This gives the log likelihood as the sum of the conditional log likelihoods on each subinterval.

To evaluate the model fit of using a set of time intervals (Y_1, ..., Y_h) and associated models \( \theta_j \), the AIC is used which is defined for a particular time interval and model as

\[ AIC(Y_j, \hat{\theta}_j) = -2 \log p(Y_j | Y_{j-1}, \hat{\theta}_j) + 2M_j \]  

where M_j is the number of estimated parameters in fitting \( \hat{\theta}_j \). Thus the AIC corresponding to (3) is given by

\[ AIC(Y_1, ..., Y_h, \hat{\theta}) = \sum_{j=1}^{h} AIC(Y_j, \hat{\theta}) \]

The optimal selection of data subintervals is chosen as the S minimizing the above AIC. An efficient procedure for selecting optimal interval lengths can be developed using an interval doubling procedure. Consider the following interval sets for model fitting: S_0 consists of intervals of some shortest length L_0; S_1 is the set of intervals of length 2L_0; ..., S_n is the set of intervals of length 2^nL. Now if the AIC's are saved for each of the subintervals, then comparisons can be made for composite models over some total data interval of say 2^nL. The interval length with the minimum AIC gives the optimum choice data length for identification of the process dynamics.

3 Model Predictive Control

Modern control systems usually utilize a model of the process to be controlled. If a reasonably accurate physical model of the process is not available, an empirical model can be identified from input-output data. The CVA identification technique has
shown considerable promise for estimating a reliable, accurate low-order model of the process from input-output data. The identified process model can then be used as a basis for the implementation of advanced control algorithms.

Model predictive control (MPC) has found widespread acceptance in the chemical industry (Garcia et al, 1989; Richalet, 1993). Several companies have specialized in the development and installation of MPC algorithms for plants and processes, and various industrial applications have been reported in recent years in the open literature. It is the ability of MPC to achieve high performance control on rather complex systems without constant expert intervention which has made it so successful. Using MPC, problems such as inverse response, large dead times, and inequity constraints on inputs and outputs can be handled in a rather straightforward manner.

Common features of MPC algorithms are, among others, the following:

(i) A performance index is employed to calculate the control action which optimizes the process output over some finite future time interval.

(ii) A reference trajectory and a vector of predicted outputs is used.

(iii) A process model is used for system representation and prediction.

The two best known MPC algorithms are Model Algorithmic Control (MAC; see Richalet et al, 1993) and Dynamic Matrix Control (DMC; see Garcia et al, 1989).

As suggested by Morari and Lee (1991), the optimal estimation of the open-loop output predictions ̂y_{f-l} (not related to the previous notation ̂y_{f}) can be represented in a two-step form:

(i) Model Prediction

In the model prediction step the predicted future (open loop) output vector ̂y_{f-l} is updated, taking into account the input at t-1.

\[ ̂y_{f-l} = M ̂y_{f-l} + S u_{t-1} \]

M is a n x n shift matrix which has the form

\[ M = \begin{pmatrix}
0 & 1 & 0 & \cdots & 0 \\
0 & 0 & 1 & \cdots & 0 \\
\vdots & \vdots & \ddots & \cdots & \vdots \\
0 & 0 & \cdots & 0 & 1 \\
0 & 0 & \cdots & 0 & 0
\end{pmatrix} \]  

(7)

and n is the model horizon. It is chosen in such a way that nΔt covers 90% of the open-loop response time of the process to be controlled. Δt is the sampling period of the system. S is the first column of the step response matrix A.

\[ A = \begin{pmatrix}
a_1 & 0 & \cdots & 0 \\
a_2 & a_1 & 0 & \cdots \\
\vdots & \vdots & \ddots & \vdots \\
a_n & a_{n-1} & \cdots & a_2 & a_1
\end{pmatrix} \]  

(8)

with

\[ a_k = R(\sum_{i=1}^{k} \Phi^{i-1})G + D \]

(9)

where a_k is the kth step response coefficient. The matrices H, Φ, G and D are evaluated from input/output data using the CVA method.

The change in input Δu_{t-1} is defined as

\[ \Delta u_{t-1} = u_t - u_{t-1} \]  

(10)

(ii) Prediction Correction

In the prediction correction step the model prediction of step (i) is corrected for disturbances and model errors, according to

\[ ̂y_{t} = ̂y_{f-l} + K[u_t - ̂y_{f-l}] \]  

(11)

y_t is the current, measured output and ̂y_{f-l} is the predicted output, based on measured data up to and including t-1 (see (14)). K is a gain vector and has the form

\[ K = [f \ldots f \ldots f]^T \quad 0 < f < 1 \]  

(12)

As pointed out by Morari and Lee (1991), the value of f lies in (0,1) and is related to the disturbance-to-noise ratio. When only small modeling errors and insignificant noise are expected, f can be chosen close to one, which results in a deadbeat observer.

The evaluation of the future open-loop outputs ̂y using a simple Kalman filter is completely decoupled from the control part of the algorithm. Thus, as this ratio goes to infinity, f goes to 0. As the ratio goes to 0, f tends to 0.

The predictive control problem, as formulated and solved in Morari and Lee (1991), involves minimization of a performance index which includes the (closed-loop) prediction error ̂e_t:

\[ E_t = R_t - ̂y_t = \begin{pmatrix} r_t \\ r_{t+1} \\ \vdots \\ r_{t+p} \end{pmatrix} - \begin{pmatrix} ̂y_{t-1} \\ ̂y_{t-1} \\ \vdots \\ ̂y_{t-1} \end{pmatrix} \]  

(13)

R_t is the vector of future set points (reference trajectory); ̂y_t is a vector of predicted future outputs based on data up to t-1. Parameter p is referred to as the prediction horizon. The vector of predicted future outputs ̂y_t is evaluated from

\[ ̂y_t = A_p^m \Delta U_t + ̂y_{pt} \]  

(14)

where A_p^m is a submatrix of the step response coefficient matrix A, comprised of the first p rows and m columns. The control horizon m is the number of control actions that are calculated in order to affect the predicted outputs over the prediction horizon p. ΔU_t is the vector of future input changes which will be the subject of the optimisation:

\[ ΔU_t = col[Δu_t, Δu_{t+1}, \ldots, Δu_{t+m}] \]  

(15)

\[ ̂y_{pt} \] includes the first p elements of the predicted output vector ̂y_t from (11).

Now consider the general quadratic measure of performance J to be minimized by appropriate selection of the future incremental control inputs ΔU_t to solve the optimal deterministic predictive control problem,

\[ J = (R_t - ̂y_t - ΔU_t)^T W_1 (R_t - ̂y_t - ΔU_t) + ΔU_t^T W_2 ΔU_t \]  

(16)

where W_1 and W_2 are weighting matrices for the output and input, respectively. This optimization problem can be solved over p time steps into the future. The minimization of the objective function J is then performed by differentiating J with respect to ΔU_t and setting this derivative to zero. The explicit control law can then be easily evaluated as

\[ ΔU_t = (A_p^T W_1 A + W_2)^{-1} A_p^T W_1 (R_t - ̂y_t) \]  

(17)

under the assumption that the weighting matrices W_1 and W_2 are diagonal. These concepts can be readily extended to the multi-input/multi-output (MIMO) case and when the process exhibits integrating behavior, as shown by Morari and Lee (1991).

4 Controller Design

In the general MPC formulation, four design parameters have to be specified: the control horizon p, the prediction horizon m, the input weighting W_2, and the output weighting W_1. Ricker (1991) proposes a controller design methodology which allows m, p, and W_1 and W_2 to be specified in a rather straightforward manner. He then suggests using the input weighting matrix W_2 as the main tuning parameter. These
tuning rules have one significant disadvantage: there is no available method to specify \( W_k \) a priori. Trial and error simulations have to be employed to evaluate a suitable \( W_k \).

As pointed out by Maurath et al. (1988), using a control horizon \( m > 1 \) immediately necessitates the specification of a nonzero weighting matrix \( W_k \), with all its related problems. One way to avoid these is to use a control horizon of \( m = 1 \). In this case the prediction horizon is used as the tuning parameter, and \( W_k \) can be set to zero. Maurath et al. (1988) show that this design method produces an effective controller that can be easily tuned, or even automated in a straightforward manner. In their paper, Maurath et al. (1988) propose a design method where the prediction horizon \( p \) is chosen such that \( q_p > 0.5k_p \), where the process gain \( K_p \) is assumed for simplicity to be positive. In the following examples, this method is used.

5 CSTR Simulations

A high fidelity simulation of a continuous stirred tank reactor (CSTR) with a single irreversible reaction, \( A \rightarrow B \), is used to represent a common nonlinear process. This is used to show how the model obtained from CVA and the MPC algorithm interact. Closed-loop simulations are used to evaluate the CVA model under closed-loop conditions. The CSTR is also used to show how the controller design works in a MIMO control configuration.

The CSTR is a rather common process in the chemical industry. Typically the reactor concentration \( C_A \) is controlled using the jacket temperature \( T_j \) as the manipulated variable. The CSTR can be described using energy and component balances. By assuming i) perfect mixing, ii) negligible ambient heat loss, iii) first-order irreversible reaction, and iv) constant physical properties of the inlet and outlet streams, the model equations are:

\[
\begin{align*}
\frac{dC_A}{dt} &= \frac{Q_f}{V} (C_{Af} - C_A) - r \\
\frac{dT}{dt} &= \frac{Q_f}{V} (T_p - T) + \frac{-\Delta H}{c_p \rho} (T - T_j) - \frac{U_A}{c_p \rho V} (T - T_j) \\
r &= C_A k_A e^{\frac{-E_A}{RT}}
\end{align*}
\]

The notation and nominal conditions are tabulated in Seborg et al. (1989). Typically the reactor concentration is measured using a gas chromatograph. Here we assume an analysis time of 4 minutes which determines the sampling period and introduces a time delay of the same magnitude in the measured response. Figure 3 shows a comparison between the step response of the process and a step response generated from a CVA model. This CVA model was evaluated from input and output data vectors containing 480 data points, where the output data were corrupted by noise \( n = 3 \times 10^{-5} \text{kg mol/m}^3 \). The input was a PRBS signal \( \pm 5.56^\circ \text{K} \) imposed on the jacket temperature. The CVA model describes the process well rather despite its nonlinearity. Figure 2 shows a comparison of the performance of a well-tuned PI controller \( (K_c = \pm 150 \text{ kg mol/m}^3, \tau = 8.25 \text{ min}) \) and the MPC algorithm \( m = 1, p = 4, f_i = 0.5 \). At \( t = 12 \text{ minutes} \) a setpoint change from 0.73 to 0.80 occurs. At \( t = 70 \text{ minutes} \) an inlet flow disturbance \( (+10\% \text{ change}) \) is introduced. At \( t = 155 \text{ minutes} \) another disturbance, this time in the inlet concentration \( (+10\% \text{ change}) \) is made. For the setpoint change and the inlet flow disturbance, the MPC algorithm performs markedly better than the PI controller; for the inlet concentration disturbance, they perform equally well. Overall, these simulations show that a CVA-based SISO MPC can perform very well when applied to processes typically encountered in industry.

Closed-loop data generated through several setpoint changes were then used to evaluate the CVA (setpoints were changed regularly from 0.8 kg mol/m^3 to 0.6 kg mol/m^3 and back – altogether 7 changes within 500 minutes). The closed-loop CVA model is shown in Figure 3. The overall quality of the identification is about the same as in the open-loop case. Figure 4 shows a comparison between the cases where the open-loop and the closed-loop CVA identified model were used to determine the \( A \) matrix and the MPC controller.

The suggested SISO controller design method can be readily extended to the MIMO case. A slightly modified CSTR model (\( \Delta t = 1 \text{ min} \); no delay in the output) is used to show the controller performance for a MIMO control configuration. The two controlled CSTR outputs are now the reactor concentration \( C_A \) and the reactor temperature \( T \). The two manipulated inputs are the jacket temperature \( T_j \) and the reactor inlet concentration \( C_{Af} \). This control configuration includes a rather challenging feature: a change in the inlet concentration results in an inverse response in the reactor concentration \( C_A \). As can be seen in Figure 5, the CVA identified model describes the process quite well. The MIMO CVA model was consequently used for control purposes. The performance of the MIMO controller with a prediction horizon of 13 is shown in Figure 6. At \( t = 10 \text{ min} \), the setpoint of the reactor concentration was changed from 0.729 (kg mol/m^3) to 0.8 (kg mol/m^3). At \( t = 100 \text{ min} \), the reactor temperature setpoint was changed from 317.11° K to 318° K. At \( t = 200 \text{ min} \), an unmeasured disturbance, namely a 10% increase in the feed flow rate \( Q_f \) was introduced. Over all the implementation of the suggested controller design method was straightforward; subsequently the controller showed very satisfactory performance.

6 Summary

Implementation of an MPC is quite straightforward using the controller design method outlined here. It works well for both SISO and MIMO systems and requires the specification of relatively few design parameters and no weighting functions. The parameters are chosen using simple ad hoc rules.

CVA models obtained from both open-loop and closed-loop data characterize the CSTR process studied here quite well, even for the cases of inverse dynamics or for a nonlinear underlying process model. The MPC controller obtained using a CVA model worked as well as one obtained from the active process model itself (no modeling error) and a comparison of MPC controllers — one designed on the basis of a CVA model obtained from open-loop data, the second using a model obtained with closed-loop data — shows few differences.

Thus we conclude that MPC techniques, when applied with CVA models obtained under either open- or closed-loop conditions, provide excellent properties for the controlled system. The advantages obtained from the use of the CVA technique to obtain process models under operating conditions (discussed above) are quite significant compared to alternative approaches.

7 Acknowledgement

The authors wish to acknowledge support of this research work by the National Science Foundation through a Small Business Innovations Research Contract to Computational Engineering, Inc.

8 References


**Figure 1:** Comparison of actual and estimated CSTR models.

**Figure 2:** Comparison of PI and MPC performance for setpoint and load changes (Δt = 4 min.)

**Figure 3:** Comparison of true and estimated model of the CSTR from closed loop data.

**Figure 4:** Comparison of MPC performance for setpoint and load changes using open-loop and closed-loop CVA models.
Figure 5: Comparison of true and estimated MIMO CSTR models.

Figure 6: MPC performance with set point and load responses for the MIMO CSTR model.