

STRUCTURE DETERMINATION OF A SEVERE NONLINEAR PROCESS

Application of Techniques to a Fed-batch Bakers' Yeast Process

M. KEULERS

Measurement and Control Group
Faculty of Electrical Engineering
Eindhoven University of Technology
POBox 513, NL5600 MB Eindhoven, The Netherlands.

Received: January 26, 1993

Abstract

The fed-batch Bakers' yeast fermentation is a nonlinear and time-varying process. A simulation model of the process has been made. Three nonlinear structure determination techniques have been tested on the simulated process: an extended Wiener-Hammerstein cascade model described by Volterra kernels (HABER, 1989), a nonlinear model, which is linear in the parameters (BILLINGS et al., 1989) and the Group Method of Data Handling (FARLOW, 1984). The objective of the work was to determine the structure and parameters of a SISO model of the glucose flow (input)→ respiratory quotient (output) dynamic relation of the nonlinear process. The technique of Billings is found to be most flexible with respect to the tuning knobs, the choice of model structure and the estimation of the parameters. Note that all techniques can be applied to any other nonlinear process.

Keywords: structure determination, nonlinear systems, biotechnology.

1. Introduction

In the past, several methods have been elaborated for the modelling of (severe) nonlinear processes. A recent overview of these methods has been given by HABER and UNBEHAUEN (1990). It is noticed in this paper that most of the methods presented in literature assume that the structure of the process is given *a-priori*. The algorithms are therefore in reality only parameter estimation algorithms missing the possibility of determining the structure of the process. However, there are several methods known, according to HABER and UNBEHAUEN (1990), that can handle the structure determination problem. These methods range from well known model structure identification techniques such as block-oriented models using Volterra kernels (HABER, 1989) to less known techniques as orthogonal estimators (BILLINGS et al., 1989). Model structure determination is often vital for modelling of (severe) nonlinear processes. Simply increasing the order and the nonlinear degree of the model in order to achieve a desired accuracy will almost certainly result in an excessively complex model

and numerical ill-conditioning. A very complex model is not only computational expensive but has also limited practical use. In this paper we will present three techniques that try to determine the correct order of a process: Block-oriented models using Volterra kernels, group method of data handling (GMDH) and an orthogonal estimator. These techniques will be elaborated on the complexity of the obtained nonlinear model and applicability of the techniques to a severe nonlinear problem.

In this paper, as an example of a severe nonlinear process, a simulation model based on the widely accepted theory of SONNLEITNER and KÄPPELI (1986) of the fed-batch Bakers' yeast production is used. The process is highly nonlinear and time-varying. A detailed description of this process can be found in chapter 2. The structure identification algorithms selected from HABER and UNBEHAUEN (1990) are tested on this severe nonlinear process. A short description of the algorithms used is given in chapter 3. The stress of this paper is on the application of the algorithms presented in chapter 3. The results of these applications will be discussed throughout in chapter 4. Here a notification will be given of problems arising with several of the applied algorithms, how to avoid them and a classification of strong and weak points will be presented. This will be done using the process described in chapter 2 as test case. Chapter 5 concludes the found results.

2. The Severe Nonlinear Process

The fed-batch Bakers' yeast process is an inherently nonlinear, time-varying process. It is normally operated in a way such that a minimal amount of glucose (molasses), primary nutrient, is needed for an optimal (maximum) production of biomass with no ethanol production and a constraint on the available oxygen.

A simulation model has been made from this process. It is capable of describing both steady states and transients of the fermentation. It is based on the oxygen limited capacity of the yeast cell (SONNLEITNER and KÄPPELI, 1986) combined with a regulated enzyme production (SWEERE, 1988) of the cell and a self containing term. Glucose is the only limiting substrate. The hypothesis of SONNLEITNER and KÄPPELI, (1986) divides the growth of the yeast cells into 3 different pathways:

pathway 1: Under optimal conditions the yeast performs an oxidative reaction with the glucose.

pathway 2: If the glucose consumption is in excess with the oxidative capacity of the cells, a reductive reaction occurs.

pathway 3: Ethanol can be metabolized oxidatively if the glucose consumption is less than the oxidative capacity.

The biomass metabolizes according to one or a combination of the three above mentioned pathways. Switching between the pathways is regulated through specific enzyme production, causing the process to be time-varying and nonlinear.

The biochemical description has been transformed to an equivalent mathematical one. It is based on first order ordinary differential equations combined with empirical rules and algebraic expressions. These ODE's include oxygen, biomass, glucose and ethanol balances as well as some enzyme balances. The model has the following assumptions and limitations:

- the fermentation takes place in a 10 litre continuous stirred tank.
- the continuous stirred tank is perfectly mixed, no gradients occur within the broth.
- the temperature and pH are kept constant at 30 °C and 5, respectively.
- foam production is negligible.
- addition of anti-foam and NH_4OH does not influence the process in any other way than keeping foam negligible and pH at 5.
- there are no correction terms for time constants, time lags, etc. of actuators and sensors.

2.1. The Simulation Set-up

The process can be seen as a three input, four output MIMO system (see Fig. 1).

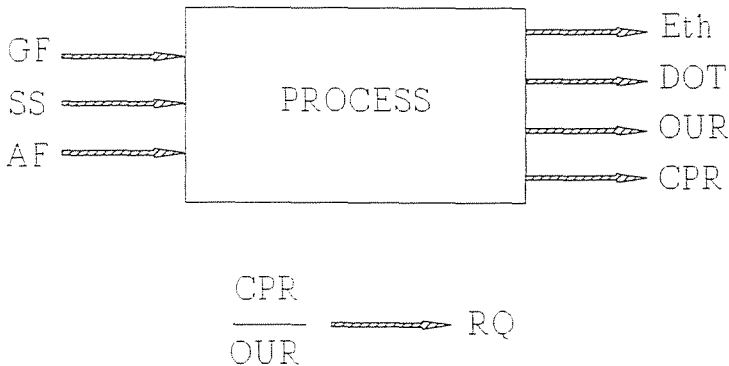


Fig. 1. The input-output representation of the process

The inputs are: stirrer speed (SS), air flow (AF) and glucose flow (GF), the outputs are: dissolved oxygen tension (DOT), ethanol concentration

(E), the oxygen uptake rate (OUR) and the carbon-dioxide production rate (CPR). In practice the requirements placed upon the controlled process are: a maximal biomass production with negligible ethanol production. Measuring the biomass concentration on-line is not always feasible; consequently an auxiliary signal has to be found for controlling the biomass concentration. This auxiliary signal is the respiratory quotient (RQ): the CPR divided by the OUR . In order to ensure maximum biomass concentration at the end of a production run the RQ has to be kept at a predefined value. RQ is influenced by GF more than by SS or AF . Although the process is MIMO, only a SISO loop will be identified: the $GF \rightarrow RQ$ loop. This loop is time-varying and highly nonlinear and is thus suitable to test the algorithms described before.

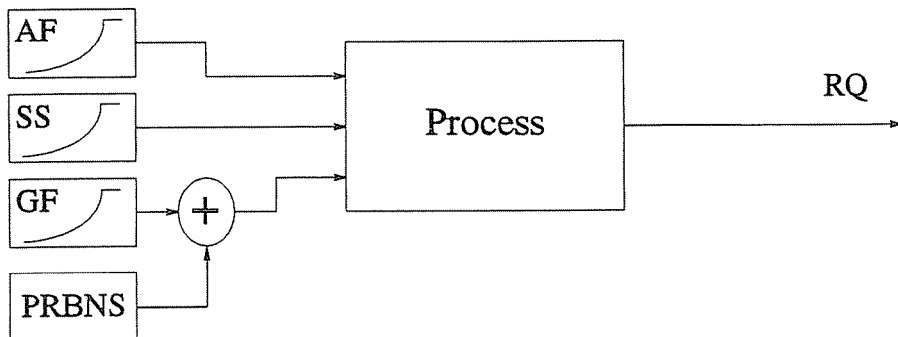


Fig. 2. The actual process with precompensated inputs.

2.2. The Simulation Data-set

The simulation set-up is chosen so that it resembles the actual process with its inputs and outputs (see Fig. 2). Care is taken to design such identification experiments that they can be executed on the real process in order to test the algorithms. Due to these requirements some attention is paid to the possible set of input signals. This set is limited due to the requirements placed upon the process, inputs and outputs. The process is operated in fed-batch mode, which requires a pre-setting of most inputs. This pre-setting is dominated by the initial value $inp(0)$ of the input and the growth rate $\mu(t)$ of the yeast and can be expressed as:

$$inp(t) = inp(0) \cdot e^{(\mu(t) \cdot t)}. \quad (1)$$

For the test, two simulation runs are selected that resemble the practical situation and are highly nonlinear. During the simulations ethanol is pro-

duced (pathway 2) as well as consumed (pathway 3). The input and output data of both runs are depicted in *Fig. 3* and *4*. The difference between the two sets is negligible. *Set 2* is regarded as being smoother than *set 1* but both data sets resemble a normal operation of a fed-batch process and are therefore good test example for the structure identification algorithms.

3. Applied Methods

In this chapter three structure identification methods are discussed: block-oriented models using Volterra kernels (HABER, 1989), a nonlinear model with linear parameters (BILLINGS et al., 1989) and Group Method of Data Handling (GMDH) (FARLOW, 1984). These are three among a vast variety of methods found in Haber and UNBEHAUEN (1990). The three algorithms are quite different in their approach to model the nonlinear process. The block-oriented method divides the process into a linear (dynamic) part and a nonlinear (static) part with corresponding kernels. The nonlinear model with linear parameters has both the static and dynamic part inside the nonlinear description. The GMDH algorithm selects inputs into a set of fixed polynomial that describe the process best at that stage. The order of the models is then increased by using the polynomials as inputs to a new set of polynomials, and so on.

3.1. Block-Oriented Models

The basic idea of block-oriented models is that the system can be divided into several blocks: blocks with static nonlinear and blocks with dynamic linear terms that are combined into a system through multipliers and summative elements. As the fermentation process is a exponential increasing process, only quadratic terms (almost exponential) will be discussed. The models can be divided into three groups (HABER, 1989):

1. **Hammerstein** models in which the static term is followed by a linear dynamic term.
2. **Wiener** models in which the linear dynamic term is followed by a static nonlinear one.
3. **Wiener-Hammerstein** cascade models which contain dynamic terms both at input and output. The nonlinear dynamic systems having polynomial static terms can generally be described by a discrete

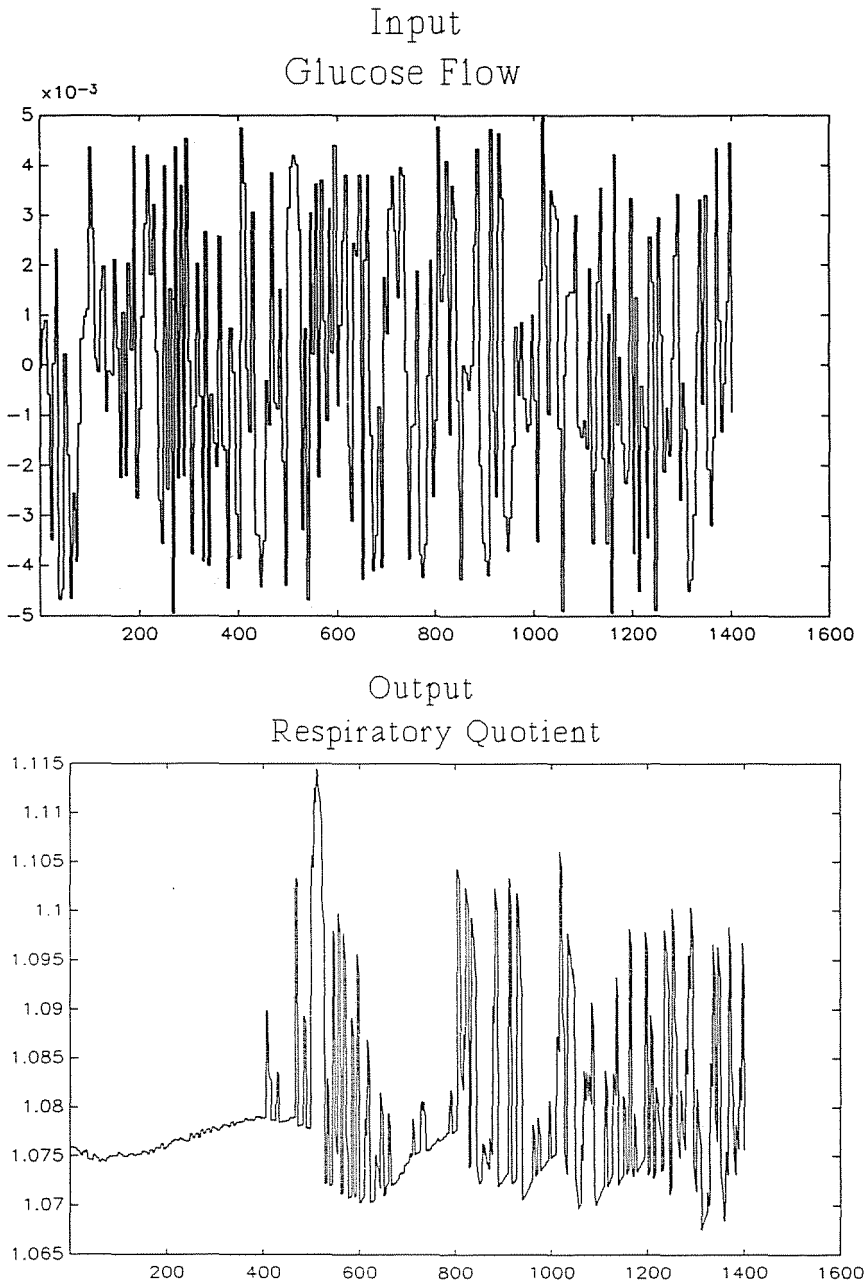


Fig. 3. Input (*GF*) and output (*RQ*) data set obtained from simulation run. This set will be denoted as *Set 1*.

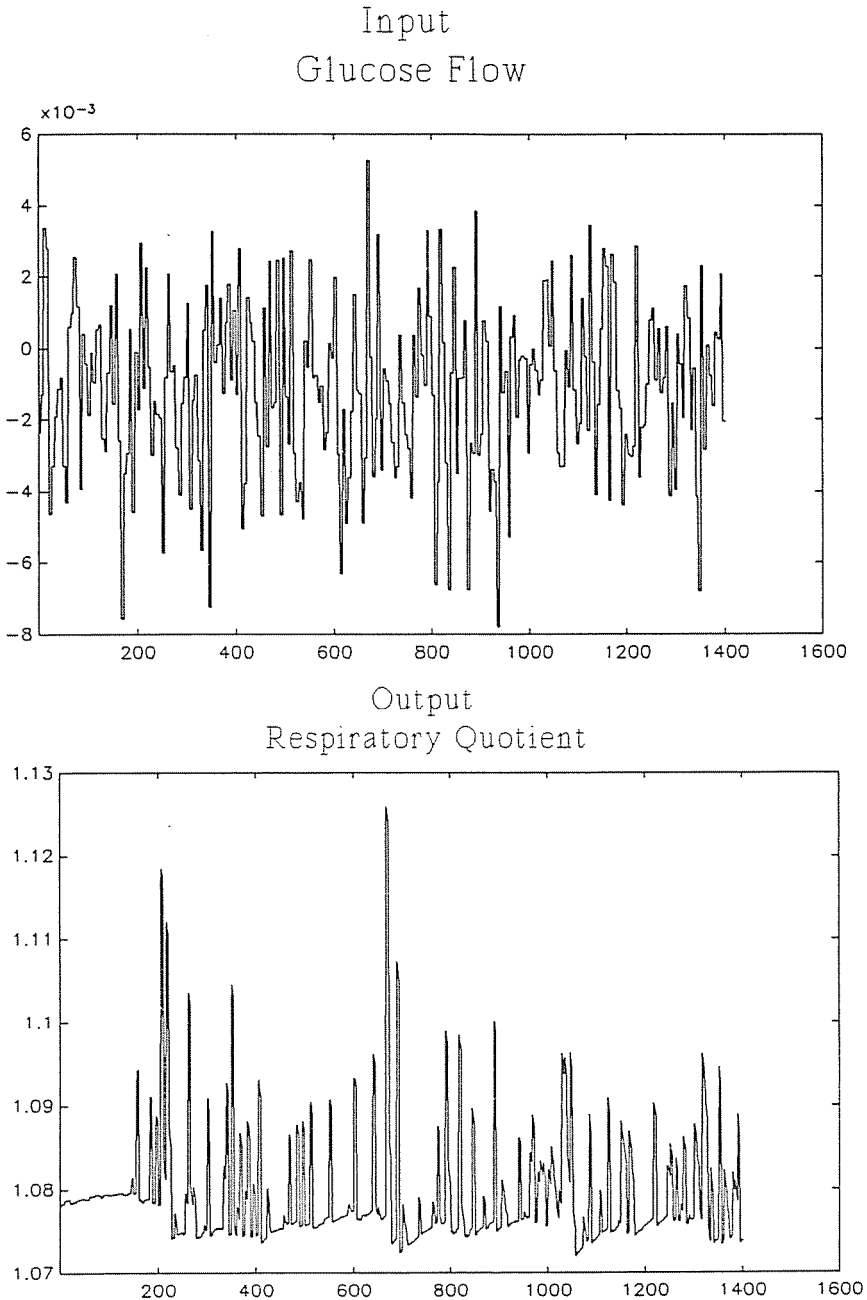


Fig. 4. Input (GF) and output (RQ) data set obtained from simulation run. This set will be denoted as *Set 2*.

Volterra series:

$$y(k) = h_0 + \sum_{\kappa_1=0}^m h_1(\kappa_1)u(k-\kappa_1) + \sum_{\kappa_1=0}^m \sum_{\kappa_2=0}^m h_2(\kappa_1, \kappa_2)(k-\kappa_1)u(k-\kappa_2) \quad (2)$$

where h_0 , $h_1(\kappa_1)$ and $h_2(\kappa_1, \kappa_2)$ are the kernels.

Parameter estimation of the Volterra series has the disadvantage that too many unknown parameters have to be estimated. On the other hand, it is possible to identify a structure that is valid for quite different models. The kernels can be estimated using an ordinary least squares method. Due to a high degree of non-linearity and a high memory value m the solution can be made numerically not feasible because of matrix inversion. A solution to this problem lies in applying a recursive least squares algorithm. Next the model structure is determined from the characteristics of the kernels. These characteristics are the graphical plots and analytical indices of the kernels (HABER, 1989). Finally, linear dynamic terms can be calculated either from the kernels or by means of direct estimation of the parameters of the parametric model of known structure. However, the linear dynamic terms cannot be reconstructed directly from the identified Volterra kernels.

The estimation of the Volterra kernels makes the one-step structural identification possible, independent from the model type as defined before. Only first- and second-degree kernels up to a sufficient high memory length m have to be estimated. The memory length is normally chosen between 10 and 20 in order to get sufficient clear graphical plots. Then the structure can be determined by applying the analytical measures (HABER, 1989) and, if not sufficient, by looking at the graphical representation of first- and second-degree Volterra kernels. Having determined the right structure, the parameters of the parametric model of known structure can be estimated. This is done using ordinary least squares estimators.

3.2. The Nonlinear Model Linear in the Parameters

The dynamic nonlinear model is assumed to be linear in the parameters. The structure identification is performed by selecting significant components from the set of possible. This can be done by using orthogonal regression methods. The model is divided into a process part (no noise elements) and a noise part. The structure of the process part is estimated and by convergence the noise part is determined. Next the actual estimation of the parameters is done over the combined process and noise parts of the model.

The model can be represented as an m outputs and r inputs nonlinear system:

$$y(t) = f(y(t-1), \dots, y(t-n_y), u(t-1), u(t-n_u), e(t-1), \dots, e(t-n_e)) + e(t), \quad (3)$$

where

$$y(t) = \begin{bmatrix} y_1(t) \\ \vdots \\ y_m(t) \end{bmatrix}, \quad u(t) = \begin{bmatrix} u_1(t) \\ \vdots \\ u_r(t) \end{bmatrix}, \quad e(t) = \begin{bmatrix} e_1(t) \\ \vdots \\ e_m(t) \end{bmatrix}, \quad (4)$$

are the system output, input and noise, respectively; n_y, n_u and n_e are the maximum lags in the output, input and noise; $\{e(t)\}$ is assumed to be a white sequence; and $f(\cdot)$ is some vector-valued nonlinear function. The nonlinear form of $f(\cdot)$ is generally unknown. One can write the model $f(\cdot)$ as a combination of a process part $f^p(\cdot)$ and a noise $f^n(\cdot)$ part:

$$y(t) = f^p(y(t-1), \dots, y(t-n_y), u(t-1), \dots, u(t-n_u)) \quad (5)$$

$$+ f^n(y(t-1), \dots, y(t-n_y), u(t-1), \dots, u(t-n_u), e(t-1), \dots, e(t-n_e)) + e(t)$$

where $f^p(\cdot)$ includes all terms $\Theta_j x_j(t)$ which do not contain noise elements. The remainder of the terms are included in $f^n(\cdot)$.

3.2.1. The Structure Identification Algorithm

For the general model, delayed noise terms $e(t-j)$ are included in the model and these have to be estimated using the prediction errors or residuals $\epsilon(t-j)$. The orthogonal property of the estimator ensures that the selection of the process and noise model parameters can be decoupled. Which terms will be included in the process model will not be affected by whatever noise model is produced later because of the orthogonal property. Initial residuals and a tolerance are then computed based on the process model and the model structure can then be re-selected with noise terms included. A revised residual sequence is calculated and so is a new tolerance and an improved process and noise model is determined. A few iterations are often enough to determine the final model. Finally it is determined whether the model is adequate by calculating some simple correlations. The original algorithm proposed by BILLINGS et al., (1989) has been improved so that convergence occurs if noise terms are identified. Tests have shown that less than 10 iterations, typically 5–6 iterations, are sufficient for the algorithm to converge with a tolerance of 1.10^3 .

3.3. Group Method of Data Handling

Ivakhnenko in 1966 introduced the Group Method of Data Handling (GMDH), which is an automatic, so called self-organizing method. GMDH can be used to build up a hierarchical multilevel model that can be transformed into an input/output one. The model itself has a cascade-like structure with more inputs and one output signal. Through the difference with the cascade structure, the model is called hierarchic and this is shown by the fact that each subsequent submodel can be estimated only in the knowledge of the output signals of the previous submodels. These submodels build a layer and the complexity of the model is characterized by the number of layers.

The system is represented as a regression equation of the form:

$$y = A + Bx_i + Cx_j + Dx_i^2 + Ex_j^2 + Fx_i x_j \tag{6}$$

for all possible combinations of inputs x_i and x_j and the output y . This will give $m(m - 1)/2$ higher-order variables for predicting the output y in place of the original m variables x_1, x_2, \dots, x_m . From the calculated $m(m - 1)/2$ variables m_1 quadratic regression models are selected that best predict y . These m_1 variables will be used to generate $m_1(m_1 - 1)/2$ regression equations for predicting y from the new variables. Essentially what is computed are a collection of polynomials of degree 4 in four variables. The process will continue until the regression equations begin to have a poorer predictability power than did the previous ones. This will happen since the regression equations are tested against a new independent set of observations. The model will start to become overparameterized. After stopping, the best of the polynomial representations in that generation is selected. Doing the necessary algebraic substitutions one arrives at the Ivakhnenko polynomial:

$$y = a + \sum_{i=1}^m b_i x_i + \sum_{i=1}^m \sum_{j=1}^m c_{ij} x_i x_j + \sum_{i=1}^m \sum_{j=1}^m \sum_{k=1}^m d_{ijk} x_i x_j x_k + \dots \tag{7}$$

3.3.1. The Algorithm

All relevant data are collected as regression type. Say n observations of m inputs and one output are collected. Next the observations are split up into a training set and a checking set. Take all the independent variables two at a time and construct the regression polynomial (Eq. 6) that best fits the dependent observation y in the training set. Now for each of the polynomials evaluate them at all data points, and store the new observations. One

should interpret these new observations as new improved variables which have better predictability powers than those of the original generations. The object is to keep only the best of these new observations, and this is where the observations in the checking set come into play. Select the observations that best estimate the dependent variable y in the checking set. One should note that the number of variables selected may be less than or greater than the original number. Also note that the test of goodness of fit was summed over the observations in the checking set. Find the best fit and compare it to the latest best fit. If it is less, repeat the construction of the regression polynomials, else the algorithm has reached its optimal structure.

4. Results

The results of the structure determination according to the three methods can be divided into two sections: numerical results (identification) and user results (handling the algorithm and its output). The stress will be on the last item, discussing the tuning knobs of the three algorithms, looking at their performance against test cases (defined by the authors), discussing the usability of different evaluation points suggested by the authors and finally applying the algorithms to the test data and evaluating it.

The determination of a nonlinear process roughly involves identification of time delays, the order of the nonlinear process, its nonlinearity and the identification of the parameters belonging to the nonlinear structure. The first item, time delay identification, can be done separately with signal processing of the input-output data. Doing so can speed up calculations of all algorithms. The order of the nonlinear process will be determined by the kind of nonlinearity and is usually restricted by the user to two. The nonlinearity is approximated by a series of products and summations. Ending the structure determination is the identification of the parameters of the obtained model. The results of the three algorithms will now be discussed separately.

4.1. Block-Oriented Models

Before doing any structure determination the order of the Volterra kernels has to be chosen in advance, in our case two (see *Eq.(2)*). Note that nonlinearities can cause a higher order and can introduce any algebraic function as input (e.g. $\sin(u)$). The set of input signals is limited by the number of input signals as the algorithm itself creates time series of these inputs. Note, however that the maximum memory length still has to

be chosen. If no delay correction has been done, the maximum memory length is rather high, incorporating the delay plus the length of the impulse response.

Second order Volterra kernels

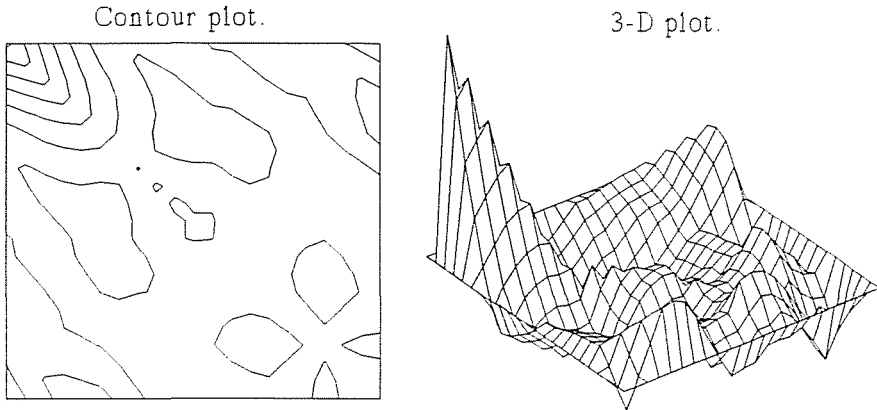


Fig. 5. Contour plot and 3-D plot of the data of *Set 1*. This data represents the Bakers' yeast process.

Second order Volterra kernels.

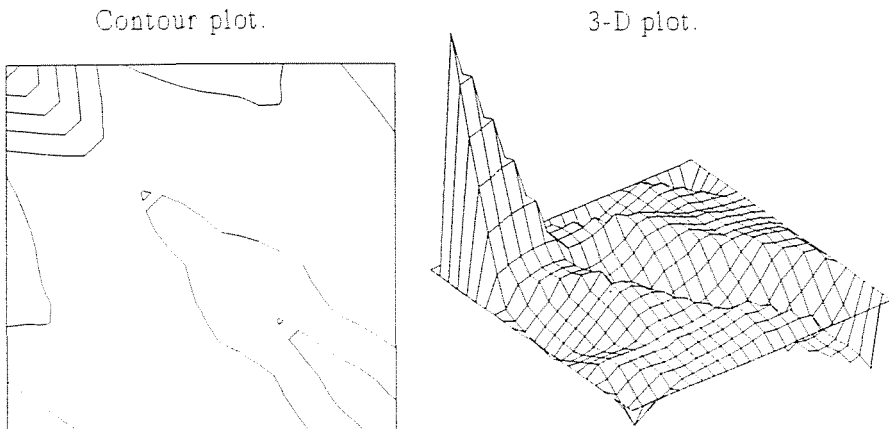


Fig. 6. Contour plot and 3-D plot of the data of *Set 2*. This data represents the Bakers' yeast process.

The algorithm was first tested on a known process described by HABER (1989). Results of these tests indicate that use of the graphical plots

gives the best results. Analytical indices formulated by HABER (1989) give no additional information. Only the information obtained from the contour plots of the graphical plots gives enough information to discriminate between model structures. The two data sets representing the yeast process are investigated with the structure identification tools based on the Volterra kernels. *Fig. 5* shows the graphical representation of *Set 1* and *Fig. 6* that of *Set 2*. It is seen that the 3-D plot and contour plot indicate a Wiener-Hammerstein model (see HABER, 1989). Due to the fact that the data of *Set 2* is smoother than that of *Set 1* the structure is better seen in the 3-D plot. The structure has been identified but leaves a great deal of freedom to the user. HABER (1989) presented 3 different Wiener Hammerstein models: simple, general and extended. The extended model was chosen. The choice was dictated by the presence of 'fast' and 'slow' dynamics in the system. The model consists of 4 different dynamic blocks, a multiplier and a constant. The dynamic block (B_1/A_1) represents the 'slow' dynamics of the system, associated with the enzyme dynamics. The other three blocks are related to the 'fast' dynamics.

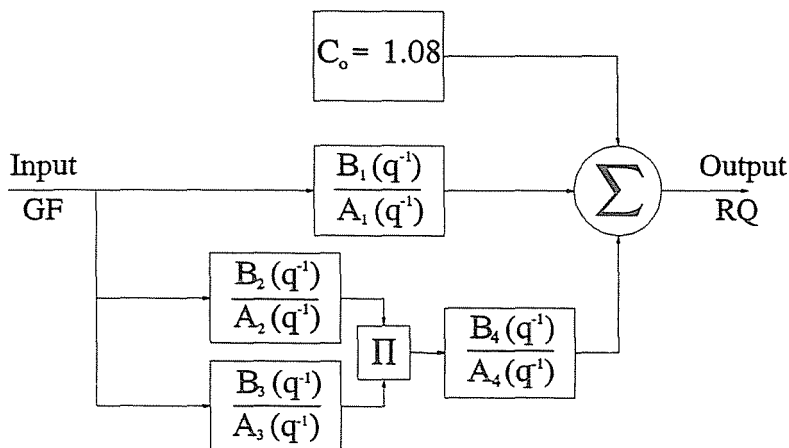


Fig. 7. The extended Wiener-Hammerstein model of the biological process

The blocks representing the system give a nice indication of the structure of the system but as already mentioned above, several different structures are applicable. Furthermore, the blocks themselves still have to be identified before any error criterion can be applied. Due to the freedom of a high order for the Volterra kernels almost any nonlinear system can be modelled this way, note, however that the computational burden becomes large. Care has to be taken when choosing the memory length, as choosing this one

not long enough will automatically lead to a wrong determination of the structure of the model.

4.2. The Algorithm of Billings

Again, before doing any structure determination the maximum order of the nonlinear model has to be chosen, in our case the order was two. The set of input signals is limited by the number of input signals and output signals, the expected delay and the memory length for input and output signals. The accuracy of the obtained process model has to be chosen in advance. This accuracy is set to a percentage of total recovery of the output signals by the (total) set of input and output signals. Next the influence of the error is incorporated by setting the stop criterion for the convergence.

The algorithm was tested on a known process described by BILLINGS et al., (1989). Results of these tests indicate the use of the selection criteria as described in BILLINGS et al., (1989). The best selection criterion is the convergence. In case of no convergence the model is too small (the order is too low, or not enough memory depth is applied). Note that this convergence criterion is only valid with noise corrupted models, as it depends on the evolution of the prediction error. Whether the model is correct can then be tested with several (auto/cross) correlation tests. Data

$$\begin{aligned}
 y(t) = & 1.08 + 0.9106 \cdot y(t-1) + 2.072 \cdot u(t) - 1.903 \cdot u(t-1) + \\
 & 120.7 \cdot y(t-1) \cdot u(t) - 107.3 \cdot y(t-1) \cdot u(t-1) + 299.6 \cdot u(t) \cdot u(t) - \\
 & 232.8 \cdot u(t) \cdot u(t-1) - 84.17 \cdot u(t-1)u(t-1) \quad (8)
 \end{aligned}$$

$$\begin{aligned}
 y(t) = & 1.08 + 0.8545 \cdot y(t-1) + 3.31 \cdot u(t) - 2.941 \cdot u(t-1) + \\
 & 19.21 \cdot y(t-1) \cdot u(t) + 422.4 \cdot u(t) \cdot u(t) - 309.9 \cdot u(t-1) \cdot u(t-1) \quad (9)
 \end{aligned}$$

from simulation runs, the same as with the Volterra kernels, is used to get an indication of the structure and parameters of the process. The two data sets are evaluated with the structure identification tool based on the algorithm of Billings. Formulas 8 and 9 give the selected terms of respectively data *Set 1* and data *Set 2*. It is seen that all the selected terms of *Set 2* reappear in *Set 1*. Furthermore, it is found that the contribution of the nonlinear terms to the output signal is small in comparison to the two major linear terms $y(t-1)$ and $u(t-1)$. The parameter values of the nonlinear terms are big in comparison with the linear terms. They also seem to compensate each other as can be seen with selected term $y(t-1) \cdot u(t)$ and $y(t-1) \cdot u(t-1)$ and also $u(t) \cdot u(t)$ and $u(t) \cdot u(t-1)$ of *Set 1*. The parameter

values have different signs and their values are comparable. This is due to the steep changes of the output signal. The original signal is reconstructed for about 90%. Note that the system has no noise incorporated such that only the process terms are selected and the user has to define the stop criterion for the process model so that this matches his needs. If the same stop criterion is chosen for *Set 1* as for *Set 2* the number of selected terms for *Set 1* is maximal.

The obtained model is less transparent than that of the Volterra kernels as there are no blocks representing it. If no noise signal is expected on the output signals the algorithm will skip the identification of any of the noise parameters. This means that an iterative procedure has to be done by the user to obtain a correct process model. This can be time consuming. If no delays are estimated, this algorithm will estimate them itself by excluding those signals with less delay. This can directly be seen from the obtained results. A measure of goodness of the obtained model, if noise is present, are the plotted correlation functions (BILLINGS, 1989), but care has to be taken e.g. when evaluating cross-correlation terms of the prediction error and the inputs.

4.3. Group Method of Data Handling

In contradiction to the other two algorithms the maximum order of the model is not chosen by the user but by the algorithm itself. The set of input signals is determined by the number of input signals, the expected delay and the memory length of the input signals. The algorithm normally divides the data set into a training and a checking set. This procedure is followed for the test, but for the Bakers' yeast data only a training set is assumed. The accuracy is determined by the algorithm itself as the total recovery percentage of the output signal by the (total) set of input signals (like in section 4.2).

The algorithm was tested on a known process described by FARLOW, (1984). Results of these tests indicate that the algorithm is self-learning and needs no presetting. This is a big contradiction to the other two algorithms. Note that with this algorithm inclusion of noise is difficult and special provisions have to be made to do so. The two data sets are evaluated with the algorithm yielding similar results (see *Eq. (10) and (11)*). It is seen that only the terms $y(t-1)$ and $y(t-3)$ are identified in a one-

$$y(t) = 1.08 + 0.8634 \cdot y(t-1) - 0.1067 \cdot y(t-3) + 6.7289 \cdot y(t-1) \cdot y(t-1) \\ + 2.4623 \cdot y(t-3) \cdot y(t-3) - 3.1006 \cdot y(t-1) \cdot y(t-3), \quad (10)$$

$$\begin{aligned}
 y(t) = & \\
 & 1.08 + 0.8767 \cdot y(t-1) - 0.1526 \cdot y(t-3) + 3.3274 \cdot y(t-1) \cdot y(t-1) - \\
 & 4.3437 \cdot y(t-3) \cdot y(t-3) - 6.5353 \cdot y(t-1) \cdot y(t-3) \qquad (11)
 \end{aligned}$$

layer model. The original signal is reconstructed for about 70% in both cases. It is interesting to observe that GMDH only selects the autoregressive terms whereas the Billings algorithm also selects inputs. This fact can be explained as an integral action.

For both results the obtained model is transparent due to the simple solution and the low reconstruction rate. If the reconstruction rate has to be higher, the model will become less transparent due to a more layer model. The algorithm can cope with delays as these are included into the first layer (by choosing the right delayed input signals). The algorithm can only be tested on goodness by its sum of squares of the output error.

5. Conclusions

The structure identification method based on Volterra kernels gives a nice indication of the structure in terms of blocks. These blocks have still to be identified using normal identification techniques. The structure identification technique proposed by BILLINGS et al., (1989), however, does identify structure and parameters at the same time. GMDH also identifies the structure and parameters at the same time but becomes less transparent when more than one layer is determined.

The Volterra kernels based algorithm is time consuming, specially if a recursive least squares algorithm has to be used. The technique has no flexibility in dealing with the memory, it has to be chosen beforehand and large enough to get a good impression of the structure and the blocks of the system. The memory restriction is also present in the other two algorithms but less stringent as these algorithms are not based on impulse responses. Note that for all algorithms it yields: when choosing the memory not high enough the algorithm does not yield practical results.

The algorithm of BILLINGS et al., (1989) is more flexible with respect to the choice of the model structure and the estimation of the parameters. The algorithm decides itself whether the process involved is a linear one or a nonlinear one by selecting relevant terms. The parameters are estimated directly with the resulting model and its structure whereas the linear dynamic terms in the Volterra kernels based algorithm have to be estimated after the model structure is chosen. Due to the fixed structure of the GMDH (the Ivakhnenko polynomial) there is less flexibility with respect to the model structure.

The algorithm of BILLINGS has only two tuning knobs: the stop criterion for the process model and the convergence criterion. The algorithm selects its own stop criterion for process and noise model together, which is sometimes very time consuming. Another advantage of the algorithm of BILLINGS is the fact of automatical estimation of delay terms in the NARMAX model and the fact that it may deal with MIMO models. The GMDH algorithm has no tuning knobs at all, which is a big advantage over the other algorithms, but makes the algorithm also less flexible in terms of interactive corrections by the user.

Comparing the outcome of the relative sum of squared output error for both data sets and for all structure determination algorithms indicates a preference for the Billings algorithm.

References

- BACKX, A. C. P. M., - DAMEN, A. A. H. (1989): Identification of Industrial MIMO Processes for Fixed Controllers. Part 1 General Theory and Practice. *Journal A*, (1) pp. 3-12.
- BILLINGS, S. A. - CHEN, S. - KORENBERG, M. (1989): Identification of MIMO Nonlinear Systems Using a Forward-Regression Orthogonal Estimator. *International Journal of Control*, Vol. 49, No. 6, pp. 2157-2189.
- FARLOW, S. J. (1984): Self-Organising Methods in Modelling: GMDH Type Algorithms. Ed. S. J. Farlow, Dekker, New York, Chapter 1, pp. 1-24.
- HABER, R. (1989): Structural Identification of Quadratic Block-Oriented Models Based on Estimated Volterra Kernels. *Int. J. Systems Sci.*, Vol. 20, No. 8, pp. 1355-1380.
- HABER, R. - UNBEHAUEN, H. (1990): Structure Identification of Nonlinear Dynamic Systems - a Survey on Input/Output Approaches *Automatica*, Vol. 26, No. 4, pp. 651-677.
- SONNLEITNER, B. - KÄPPELI, O. (1986): Growth of *Saccharomyces Cerevisiae* is Controlled by its Limited Respiratory Capacity: Formulation and Verification of a Hypothesis. *Biotechnology and Bioengineering*, Vol. 28., pp. 927-937.
- SWEERE, A. (1988): Response of Bakers' Yeast to Transient Environmental Conditions Relevant to Large-Scale Fermentation Processes. Ph.D. Thesis, Delft University of Technology.