A GMDH-PTSV Algorithm and Its Application to Modelling for Fermentation Processes*

ZHAO Jiang

(Department of Automation, Hebei Institute of Mechanism and Electronics·Shijiazhuang, 050054, PRC)

WANG Junshe

(The 54th Research Institute of Ministry of Mechano-Electric Industry·Shijiazhuang, 050081, PRC)

TIAN Shubao

(Department of Automation, Tianjin University·Tianjin, 300072, PRC)

Abstract: In this paper, a modelling technique, GMDH-PTSV (Group Method of Data Handling with the Principal-Component Two-way Selecting Variables) algorithm, is presented, in which selecting variables is based on the data taken from the system but not on the relative matrix constructed according to one's knowledge and experience. The model accuracy obtained using GMDH-PTSV is compared with that obtained using other three algorithms through simulation. At last, the GMDH-PTSV algorithm is applied to modelling for fermentation processes.

Key words: modelling; identification; complex system; multivariable; GMDH; biotechnical process; nonlinear system

1 Introduction

self-organizing nonlinear mapping algorithm, known as the Group Method of Data Handling (GMDH), was introduced by A. G. Ivakhnenko in 1968[1]. It is considered to be a new idea[2] for the identification of many complex systems (nonlinear, multivariate) which are relatively known little, such as economical, social and environmental systems. Up to now, the GMDH algorithm has had some improved forms and applications of these algorithms have shown that the GMDH algorithm are effective for modelling and prediction for complex systems for example Duffy et. al[3].

On the basis of GMDH and ISM (Interpretive Structural Modelling), Minoru Ryobe, et. al. presented Group Method of Structural Modelling (GMSM)[4]. In the algorithm, one has to know a technological process as much as possible to construct a relative matrix for selecting input variables. In fact, the matrix is usually inaccurate because of the differences in one's knowledge; besides, it is very difficult to express exactly the dependence between variables by 0-1 matrix. Consequently, considerable "man-made disturbance" is led to the modelling when either GMSM or its improved forms are applied. Moreover, the relative matrix usually has to be modified again and again in computation.

* The subject subsidized by the National Natural Science Foundation of P. R. China.
Manuscript received Nov. 14, 1990, revised June 22, 1991.

According to the principle that the essence of a system is implicated in and only in its data, we present a modelling technique, Group Method of Data Handling with the Principal-component Two-way Selecting Variables (GMDH-PTSV) algorithm. In the algorithm, selection of variables is based on the data. The variables and the dependence between them are decreased by the PTSV. Therefore, the accuracy of model and the stability of computation are greatly improved.

In this paper, these are discussed: GMDH-PTSV algorithm, simulation of the algorithm and its application to modelling of the fermentation processes.

2 The Algorithm of GMDH with the Principal-Component Two-Way Selecting Variables for Modelling

2.1 Analysis and Determination of Principal Component

The principal component approach is a technique used in multivariate analysis for reducing the dimension of the observable vector variables[5].

Let $X=(x_1,x_2,\cdots,x_m)^{\mathrm{T}}$ be a random vector with

$$E(X)=\mu,\ \mathrm{cov}(X)=S=(s_{ij}),$$

where μ is a real m-vector and S (of order $m\times m$) is a real positive semidefinite matrix.

Assume that $v_i(i=1,2,\cdots,m)$ is a eigenvalue of S and the eigenvector of S corresponding to v_i is $d_i(i=1,2,\cdots m)$. Obviously it may true that

$$v_1\geqslant v_2\geqslant\cdots\geqslant v_m\geqslant 0 \text{ and } d_i^{\mathrm{T}}d_i=1.$$

From [5], the following definition is given.

Definition The normalized linear function $d_kX=\sum_{i=1}^{m}d_{ik}x_i$, where d_k is the normalized eigenvector of S corresponding to its K^{th} largest eigenvalue v_k, is called the K^{th} principal component of X.

For avoiding the problem of units used in the analysis we shall use in practice the sample correlation matrix instead of the sample covariance matrix to estimate the principal component.

Let

$$y=f(X),$$

where $X=(x_1,\cdots,x_m)^{\mathrm{T}}$ is the random input variables vector, and y is the output variable.

The data from the observation are

$$\left[\begin{array}{cccc:c} x_{11} & x_{12} & \cdots & x_{1m} & y_1 \\ x_{21} & x_{22} & \cdots & x_{2m} & y_2 \\ \cdots & \cdots & \cdots & \cdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} & y_n \end{array}\right].$$

The sample correlated matrix of X is

$$R=R(X)=\begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1m} \\ r_{21} & r_{22} & \cdots & r_{2m} \\ \multicolumn{4}{c}{\cdots\cdots\cdots\cdots} \\ r_{m1} & r_{m2} & \cdots & r_{mm} \end{bmatrix}.$$

Where

$$r_{ij} = r(x_i, x_j) = \frac{s_{ij}}{\sqrt{s_{ii}s_{jj}}}, \tag{2.1}$$

$$s_{ij} = \frac{1}{(n-1)} \sum_{k=1}^{n} (x_{ki} - \bar{x}_i)(x_{kj} - \bar{x}_j), \tag{2.2}$$

$$\bar{x}_j = E(x_j) = \frac{1}{n} \sum_{k=1}^{n} x_{kj}, \qquad i, j = 1, 2, \cdots, m. \tag{2.3}$$

Taking note of that the matrix R is real symmetric and positive semidifinite, we have the following theorems.

Theorem 1 For the correlative matrix R (of order $m \times m$), with eigenvalues $v_1 \geqslant v_2 \geqslant \cdots \geqslant v_m \geqslant 0$, it has m mutually orthogonal unit eigenvectors $a_1, a_2, \cdots, a_m$, with

$$Ra_j = v_j a_j, \qquad (j = 1, 2, \cdots, m),$$

$$A^{\mathrm{T}}RA = V.$$

Where A is the orthogonal matrix with colums $a_1, a_2, \cdots, a_m$ and $V = \mathrm{diag}(v_1, v_2, \cdots, v_m)$.

The proof can be obtained from any textbook on matrix theory.

Making a canonical transformation, we let

$$\hat{x}_j = \frac{x_j - \bar{x}_j}{\sqrt{s_{jj}}}, \quad j = 1, 2, \cdots, m, \tag{2.4}$$

$$\hat{y} = \frac{y - \bar{y}}{\sqrt{s_{yy}}}, \tag{2.5}$$

where

$$\bar{y} = E(y) = \frac{1}{n} \sum_{k=1}^{n} y_k,$$

$$s_{yy} = \frac{1}{(n-1)} \sum_{k=1}^{m} (y_k - \bar{y})^2.$$

Thus we have the following theorem.

Theorem 2 For the vector $\hat{X}$ with n samples, there exists an orthogonal transformation

$$Z = A^{\mathrm{T}}\hat{X}. \tag{2.6}$$

such that $\mathrm{cov}(Z) = (n-1)V$, where $\hat{X} = (\hat{x}_1, \hat{x}_2, \cdots, \hat{x}_m)^{\mathrm{T}}$, $Z = (z_1, z_2, \cdots, z_m)^{\mathrm{T}}$ and V is a diagonal matrix with diagonal elements $v_1 \geqslant v_2 \geqslant \cdots \geqslant v_m \geqslant 0$, the ordered eigenvalues of the correlation matrix R. The i^{th} column a_i of A satisfies $(R - v_i I)a_i = 0$. The components of Z are uncorrelated and z_i has maximum variance among all normalized linear combinations uncorrelated with $z_1, z_2, \cdots, z_{i-1}$.

Proof

$$\mathrm{cov}(Z) = ZZ^{\mathrm{T}} = A^{\mathrm{T}}\hat{X}\hat{X}^{\mathrm{T}}A,$$

Let

$$Q = \hat{X}\hat{X}^{\mathrm{T}},$$

then

$$q_{ij} = \sum_{k=1}^{n} \hat{x}_{ki}\hat{x}_{kj} = \sum_{k=1}^{n} \frac{x_{ki} - \bar{x}_i}{\sqrt{s_{ii}}} \frac{x_{kj} - \bar{x}_j}{\sqrt{s_{jj}}} . \tag{2.7}$$

Thus from (2.1) and (2.2) we have

$$q_{ij} = (n-1)\frac{s_{ij}}{\sqrt{s_{ii}s_{jj}}} = (n-1)\, r_{ij} . \tag{2.8}$$

So that $Q=(n-1)R$,

$$\mathrm{cov}(Z) = (n-1)A^{\mathrm{T}}RA .$$

Since A is the orthogonal matrix whose i^{th} column is the eigenvector corresponding to the eigenvalue v_i of the matrix R, we obtain from theorem 1

$$\mathrm{cov}(Z) = (n-1)\begin{pmatrix} v_1 & & 0 \\ & v_2 & \\ & & \ddots \\ 0 & & v_m \end{pmatrix} = (n-1)V , \tag{2.9}$$

that is

$$\mathrm{cov}(z_i, z_j) = \begin{cases} (n-1)v_i, & (i=j), \\ 0, & (i \neq j), \end{cases} \qquad i,j = 1,2,\cdots,m. \tag{2.10}$$

The proof is complete.

From Theorem 2 we can conclude that corresponding to the eigenvalue $v_i (v_1 \geqslant v_2 \geqslant \cdots \geqslant v_m \geqslant 0)$, z_i is the i^{th} principal component of X.

The estimate of the total system variance is may given by $\sum_{i=1}^{m} v_i$ and is called the total sample variance. The importance of the i^{th} principal component is measured by

$$v_i \Big/ \Big(\sum_{i=1}^{m} v_i \Big)$$

which, when expressed in percentage, will be called the percentage of contribution of the i^{th} principal component to the total variance.

If the first P principal components contain a large amount of total information of the vector X, they may be used in further investigation in place of the original vector X. So the dimension of the problem is considerably reduced. However, because the principal components are the linear combinations of X, the variables in a final model can not be reduced. For this reason it is natural to look for methods for selecting the principal variables among the original ones.

2.2 The Algorithm of the Principal-Component Two-Way Selecting Variables (PTSV)

For the selection of a "good" subset of variables, the properties of the principal components should be investigated.

From Theorem 2 it follows that

$$z_i = a_i^{\mathrm{T}}\hat{X} = \sum_{k=1}^{m} a_{ki}\hat{x}_k .$$

Obviously, weight a_{ki} is a measure of the importance to be palced on the component $\hat{x}_k$ in the i^{th}

principal component z_i. If

$$|a_{qi}| = \max_k \{|a_{ki}|\},$$

the ignoring of the i^{th} principal component z_i is nearly equated to the eliminating of the q^{th} original variable x_q which, in this paper, is called the first principal variable of the i^{th} principal component z_i.

In the algorithm of backward selection of variables based on the principal component[6], the variable eliminated is the first principal variable of the least component. As a matter of fact, the variable is sometimes the same as the first principal variable of the largest component and in this case the loss of the information of the variables is too much.

In order to overcome the shortcoming of the algorithm we modify it as PTSV in which the variables omitted are selected on the basis of the investigating the first two principal variables of the largest component and the same variables of the least component respectively. We'll briefly discuss it here.

Let the first and the second principal variables of the largest component be respectively $X_m^{(1)}$ with weight $a_m^{(1)}$ and $X_m^{(2)}$ with weight $a_m^{(2)}$ and the same of the least component be respectively $X_l^{(1)}$ with weight $a_l^{(1)}$ and $X_l^{(2)}$ with weight $a_l^{(2)}$. Further, we define

$$b_m = |a_m^{(2)}/a_m^{(1)}| \quad \text{and} \quad b_l = |a_l^{(2)}/a_l^{(1)}|. \tag{2.11}$$

Then the variables eliminated should be selected in the following principles.

a) If $X_m^{(1)}$ is not the same as $X_l^{(1)}$ the variable omitted should be $X_l^{(1)}$.

b) If $X_m^{(1)}$ is the same as $X_l^{(1)}$ several situations are to be considered.

1° $b_l \geqslant 0.8$ and $b_m \leqslant 0.8$.

This indicates that the weights of $X_l^{(1)}$ and $X_l^{(2)}$ are nearly equal in the least principal component and conversely $X_m^{(1)}$ is much more important than $X_m^{(2)}$ in the first principal component. Therefore the variable dropped should be $X_l^{(2)}$.

2° $b_l \geqslant 0.8$ and $b_m > 0.8$.

Based on the similar reason $X_l^{(1)}$ is eliminated.

3° $b_l < 0.8$ and $b_m > 0.8$.

The variable eliminated should also be $X_l^{(1)}$.

4° $b_l < 0.8$ and $b_m \leqslant 0.8$.

In this case $X_l^{(2)}$ should be omitted if $b_l/b_m > 5$, and otherwise $X_l^{(1)}$ is omitted. The reason is that the information loss of the variables should be as little as possible in the elimination. In case that $X_m^{(1)}$ and $X_l^{(1)}$ are all very important in the principal components of themselves $X_l^{(2)}$ should be omitted only when the importance of $X_l^{(2)}$ in the least component is much greater than that of $X_m^{(2)}$ in the largest component.

In summary we can obtain the principle of eliminating variables. If

$$(b_l \geqslant 0.8 \text{ and } b_m \leqslant 0.8) \text{ or } (b_l < 0.8 \text{ and } b_l/b_m > 5), \tag{2.12}$$

the variable eliminated should be $X_l^{(2)}$. Otherwise $X_l^{(1)}$ is dropped.

The selection process should stop when the loss of information exceeds a certain value. In

order to avoid lossing too much information the calculation of the loss [6] is revised as

$$J = l^0\mathrm{DEC}(k) + l^1\mathrm{DEC}(k-1) + \cdots + l^k\mathrm{DEC}(0), \tag{2.13}$$

where $\mathrm{DEC}(k) = v_{m_k} / \left(\sum_{i=1}^{m_k} v_i \right)_k$ is the percentage of contribution of the least component after the K^{th} variable eliminated has been selected and $l(0<l<1)$ is the weight efficient designated.

Summarizing the above we can give in Fig. 1 an abbreviated flowchart of the algorithm of PTSV.

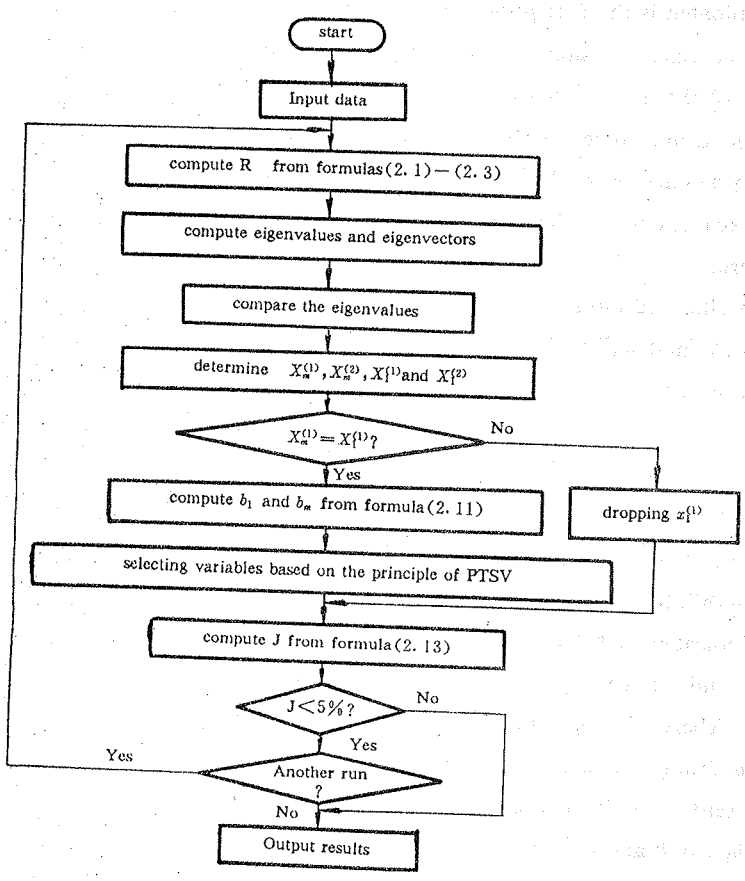

Fig. 1 Flowchart of PTSV algorithm

2.3 The Algorithm of the GMDH with the Principal-Component Two-Way Selecting Variables (GMDH-PTSV) for Modelling

The algorithm of the GMDH-PTSV may be considered as a special combination of the improved GMDH algorithm, the GMDH with Initial Varialble Keeping algorithm (GMDH-IVK)[7], with the algorithm of the PTSV. The selection of the input variables using the PTSV is based on the data taken from the system to be identified instead of the knowledge about the system. Moreover, the variables kept in each layer of GMDH-IVK are selected using the PTSV but not only according to the sequence of the error on the test set data. In order to obtain better convergence one to three variables are selected based on the minimum error and entered the next

GMDH-layer directly. So the dependences among the input variables in each layer of the GMDH are decreased and as such the algorithm has better computation stability and model accuracy. The network of GMDH-PTSV is shown in Fig. 2 where S, G, C and R express respectively selector for selecting variables, basic-GMDH network, comparer for ordering partial polynomials with respect to the smallest MSE (mean-square error) and recorder for keeping initial variables.

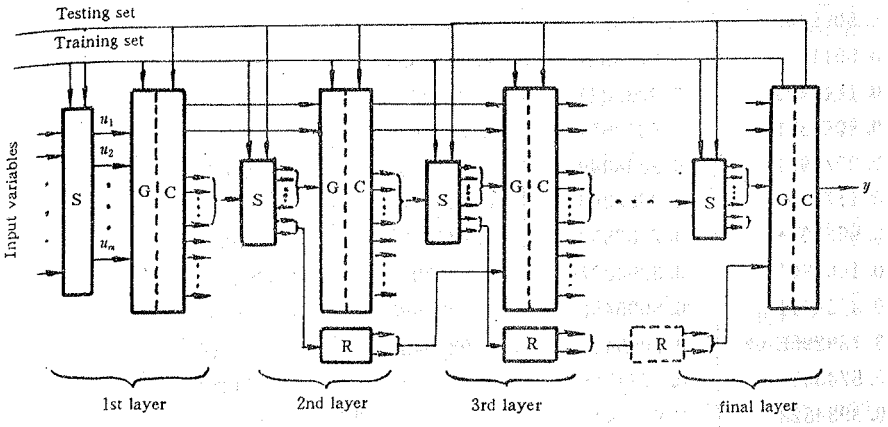

Fig. 2 Abbreviated network of PTSV—GMDH

3 Simulation Study of the GMDH-PTSV Algorithm

Three GMDH algorithms, Basic-GMDH, GMDH-IVK and GMDH-PTSV are used to modelling for three simulation examples which are respectively

1) $y_1 = 2 + u_1 + u_3 + 2u_1u_2 - u_2^2 + \sin(u_2u_4)$.

2) $y_2 = 0.3 + u_1 + u_3 + 2u_1u_2 - u_2^2 + u_1u_4 + u_4^3$.

3) $y_3 = 2 + u_1 + e^{u_3} + 2u_1u_2 - u_2^2 + u_4^3 + u_3u_5$.

30 measurements of input variables are generated by random function of Great Wall 0520C-H computer and the data are separated into two sets, the "training set" with 20 measurement times and the "testing set" with 10. Sum of square error of the testing set data is

$$Rss_i = \sum_{t \in \{T_2\}} [\hat{y}_i(t) - y_i(t)]^2, \quad (i = 1,2,3),$$

where $\{T_2\}$ is the set of times in the testing set and $\hat{y}_i(t)$ is the estimate of $y_i(t)$. For showing conveniently the computation, we define

$$V(w,z) = (1\ w\ z\ w^2\ z^2\ wz)^T$$

and

$$G_{kp} = (g_0\ g_1\ g_2\ g_3\ g_4\ g_5)^T.$$

Because of limitation of space, we give only the data used in example 1 in Table 1, the coefficients vector of example 1 using GMDH-PTSV algorithm in Table 2 and the results of Rss for all examples in Table 3. The estimate of y_1 is expressed in the coefficients vector here

$$y_1 = G_{32}^T V(y_{22}, y_{24}), \quad y_{22} = G_{22}^T V(y_{12}, y_{14}), \quad y_{24} = G_{24}^T V(y_{11}, y_{14}).$$

$$y_{11} = G_{11}^T V(u_1, u_2), \quad y_{12} = G_{12}^T V(u_1, u_3), \quad y_{14} = G_{14}^T V(u_2, u_3).$$

Table 1 Data used in simulation example 1

u_1	u_2	u_3	u_4	y_1
1.100712E-02	0.4984514	0.9502588	0.7933141	3.108989
0.6762106	0.701513	0.8037676	0.915644	4.535667
0.8700599	0.5386053	0.9545389	0.3951538	4.68297
0.9413699	0.0130919	0.4658356	6.872218E-02	3.432582
0.5095435	0.257972	0.9415805	9.235416E-02	3.671293
0.991137	0.9796342	0.1952339	0.9817234	4.988774
0.1841499	0.8338841	0.3443373	0.4926043	2.539564
0.8054391	0.6613618	0.7995875	0.3138043	4.439052
0.9076961	0.2834849	0.1113796	0.808787	3.680624
0.1178653	0.8692248	0.7991675	0.4338387	2.734613
0.9629044	0.7468883	0.8458033	0.2433331	4.869973
0.1667961	0.8294881	0.526898	0.378211	2.590955
0.4024711	6.600546E-02	2.187459E-02	0.3148507	2.4939
8.189295E-02	0.2000422	0.0325452	0.9976216	2.30543
0.5743653	0.1734918	0.8299087	0.5344253	3.666056
0.3934828	0.413344	1.873849E-02	0.1373083	2.623381
0.7834805	0.1546669	0.9905094	0.6699325	4.095856
0.1797593	0.1644999	0.4235958	7.666711E-02	2.648047
0.7366418	0.9359299	0.7713543	0.9534967	4.789506
0.1592663	0.3447531	0.4796381	0.0243849	2.638272
0.5237618	0.2123873	8.712356E-02	0.6410765	2.923994
0.2651788	7.457916E-02	0.9426256	0.7717748	3.299323
0.2464716	0.667427	0.8407791	0.9886303	3.583785
7.520019E-02	0.8491488	0.2332966	9.747193E-03	1.723432
0.5894254	0.2588048	0.1707713	0.6394001	3.163034
0.1881254	0.6437961	0.4725236	0.3991221	2.742539
4.906027E-02	0.4854909	0.6270657	0.8966162	2.909749
0.6858248	0.2694724	0.2040152	0.3322105	3.276249
0.3543726	0.1956527	0.6182844	0.5578479	3.181973
0.4666672	4.278986E-02	0.5130912	0.5296165	3.040525

Table 2 Coefficents vector of example 1 using GMDH-PTSV algorithm

G_{11}	G_{12}	G_{14}	G_{22}	G_{24}	G_{32}
1.927509	1.746397	3.001389	4.937378	−3.278809	0.210083
2.279015	1.869362	−2.181816	0.4691468	1.379028	2.388184
2.092461	1.356396	−0.7390823	−2.481049	1.185913	−1.479004
−0.8128302	0.5574417	3.130039	−0.2834625	−0.140564	−0.2597656
−2.087898	4.594422E-02	2.030788	2.827454E-02	−0.1694908	0.303711
1.702858	−0.643364	7.228088E-02	0.7122803	0.1428833	−3.613281E-02

Besides, the example for modelling using GMSM in [4] is modelled by GMDH-PTSV and

the estimate of X_1 is

$$X_1 = -109.4375 + 15.32813X_2 + 14.59204X_4 - 0.50337231X_2^2 - 0.486084X_4^2 - 0.2998047X_2X_4.$$

The data of variables can be obtained from [4]. The mean-square errors of total data by GMDH-PTSV and GMSM are respectively 2.344556 and 2.511073.

Table 3 Comparision of simulation Results

algorithm		PTSV-GMDH	IVK-GMDH	BASIC-GMDH
example 1	final layer	3	2	2
	Rss1	6.839304E-02	0.164412	0.1730471
example 2	final layer	3	3	2
	Rss2	9.969799E-02	0.1664206	0.1660074
example 3	final layer	3	3	3
	Rss3	0.4765467	0.5130102	0.8109914

Comparing the results of GMDH-PTSV with any one of the other algorithms we can reasonably conclude that the models using the GMDH-PTSV algorithm have less order (if the accuracy of the models are the same), better accuracy and stability. In particular, the algorithm can be more effectively adapted to nonlinear natures of the processes.

4 Application to Modelling the Fermentation of Cephalosporin C

The antibiotic fermentations are of very complex processes with nonlinear and multivariate. Because of the difficulties of measuring on-line main states it is important to extract pertinent information in terms of descriptive mathematical models for prediction and control of the processes. Unforturnately, the modelling for these processes is almost impossible by means of mechanism analysis if more than, say, four variables are to be taken into account. Therefore, it is effective in a sense to apply the GMDH-PTSV algorithm to modelling for the processes based on the data obtained from the systems. In the following we present the results of modelling for the fermentation of Cephalosporin C.

By the PTSV algorithm, five principal input variables are selected from twelve process variables and they are pH(u_1), glucose concentration (u_2), NH_3(u_3), DO(u_4) and temperature (u_5). Output variables to be estimated are CPC concentration (y_1), cell mass (y_2) and purity (y_3). As the report data from the real process are not permitted to be announced, a part of the results can be published here. The stable models of y_1, y_2 and y_3 are respectively

$$y_1 = G_1^{\mathrm{T}}V(u_3, u_4),$$

$$y_2 = G_{21}^{\mathrm{T}}V(z_1, z_2), \quad \text{where} \quad z_1 = G_{22}^{\mathrm{T}}V(u_2, u_3), \quad z_2 = G_{23}^{\mathrm{T}}V(u_3, u_5)$$

and

$$y_3 = G_{31}^{\mathrm{T}}V(f_1, f_2), \quad \text{where} \quad f_1 = G_{32}^{\mathrm{T}}V(u_2, u_3), \quad f_2 = G_{33}^{\mathrm{T}}V(u_1, u_5).$$

The coefficient vectors are shown in Table 4 and comparisons between measurements and estimates of the outputs y_1, y_2 and y_3 respectively in (a), (b) and (c) of Fig. 3.

Table 4 Coefficients vectors of stable models

G_1	G_{21}	G_{22}	G_{23}	G_{31}	G_{32}	G_{33}
7.089138	0.1625977	0.6906738	−3.220703	−1.828125	46.96826	−60
−9.270508	0.7617188	1.411053	−10.78906	1.359375	−13.96484	−232
−0.8937588	−0.1621094	−0.2223969	18.35156	−0.2890625	0.5019379	76.5
3.062851	4.453125	0.1051025	−0.99002344	−7.080078E-03	4.077149E-02	24
7.443458E-02	5.046875	0.1881714	−13.98438	5.371094E-03	−1.624376E-03	−1.417969
0.492445	−9.3125	−1.019043	11.42969	1.953125E-03	5.236817E-02	−2.5625

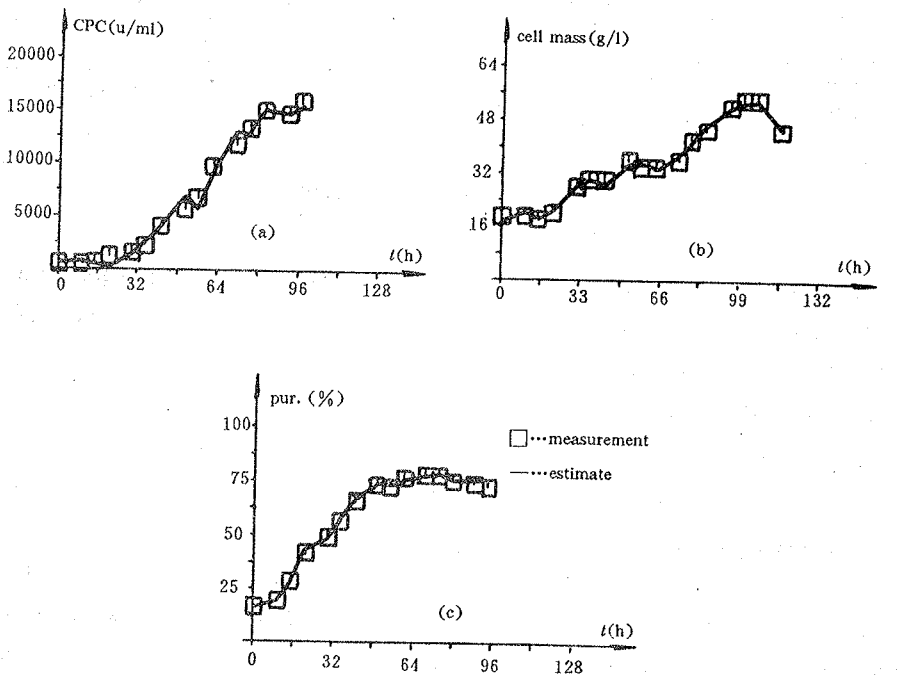

Fig. 3 The comparisons between measurements and estimates of the outputs

5 Conclusion

On the basis of the forgoing results and theoretical arguments, the following general conclusions could be made about the GMDH-PTSV algorithm. 1) Both selecting variables and modelling processes are based on the data therefore the "man-made disturbance" is decreased. 2) The model accuracy and computation stability by the algorithm is better than that by any one of the other three algorithms mentioned in this paper. 3) It is effective for modelling of multivariable, nonlinear, complex systems which is relatively known little. 4) Its application to modelling for the fermentation process of Cephalosporin C, has shown that the models are useful and practical.

References

[1] Ivakhnenko, A. G.. Polynomial Theory of Complex Systems. IEEE Trans. Syst., Man, Cybern., 1971, SMC-1:364—378

[2] Bekey, G. A. and Saridis, G. N.. On Identification and System Parameter Estimation. Proc. of 6th IFAC Symp. Wach DC, U. S. A., June, 1982, 7—11

[3] Duffy, J. and Franklin, M.. A Learning Identification Algorithm and Its Application to an Environmental System. IEEE Trans. Syst., Man, Cybern., 1975, SMC-5(2):226—239

[4] Ryobe, M. et. al.. System Analysis of A Cement Plant by the GMSM. System and Control, 1979, 23(10):587—593 (in Japanese)

[5] Narayanc, G.. Multivariate Statistical Inference. New York:Academic Press, 1977

[6] Lu Jiyu and Xi Ying. The Selection for the Optimal Variable Subset of Regression Model. Mathematics in Practice and Theory, 1987, (3):22—28 (in Chinese)

[7] Yang Zihou, et. al.. An Improved GMDH Algorithm—The Initial Variable-Keeping Algorithm. Acta Automatica Sinica, 1986, 12(4):379—400 (in Chinese)

GMDH-PTSV 算法及其在发酵过程建模中的应用

赵 江

(河北机电学院自动化系·石家庄,050054)

王俊社

(机电工业部第54研究所·石家庄,050081)

田树苞

(天津大学自动化系,300072)

摘要：本文提出了 GMDH-PTSV(具有主成份双向变量选择器的 GMDH)建模算法. 在此算法中,变量的选择是根据对系统的测量数据,而不是根据人的经验构造出的(0-1)相关矩阵. 通过仿真,对 GMDH-PTSV 算法和其它三种算法进行了比较. 最后,应用 GMDH-PTSV 算法对一种生物发酵过程建立了模型.

关键词：建模;辨识;复杂系统;多变量;GMDH 算法;生化过程;非线性系统

本文作者简介

赵 江 1958年生. 1982年毕业于河北机电学院自动化系,并获学士学位. 1988年毕业于天津大学研究生院,并获工学硕士学位. 现为河北机电学院自动化系讲师. 研究兴趣是多变量系统的自组织建模,神经元网络和模糊控制. 目前研究领域主要是神经网络在系统辨识与控制中的应用.

王俊社 女. 1960年生. 1982年毕业于河北机电学院自动化系,获学士学位. 后一直在机电部第54研究所从事计算机软件研制开发工作. 研究兴趣是计算机仿真,计算机通信. 目前主要研究领域是计算机网络通信控制软件的开发与研制.

田树苞 1937年生. 1960年毕业于天津大学工业电气自动化专业. 后留校任教. 现任电力及自动化系教授,从事自动控制理论教学及研究工作. 主要研究工作为工业过程的建模及优化控制方法,机器人的控制算法. 对生物发酵(谷氨酸、头孢菌素)过程的建模及优化控制进行过多年探讨.