2004 International Symposium on Nonlinear
Theory and its Applications (NOLTA2004)
Fukuoka, Japan, Nov. 29 - Dec. 3, 2004

Application of Automatic Choosing Function Model and Genetic Algorithm for Hammerstein System Identification

Tomohiro Hachino and Hitoshi Takata

Department of Electrical and Electronics Engineering, Kagoshima University
1-21-40 Korimoto, Kagoshima, 890-0065 Japan
Email: hachino@eee.kagoshima-u.ac.jp, takata@eee.kagoshima-u.ac.jp

Abstract—This paper deals with an identification method of Hammerstein type nonlinear systems by using an automatic choosing function (ACF) model and genetic algorithm (GA). An unknown nonlinear static part to be estimated is approximately represented by the ACF model. The connection coefficients of the ACF and the system parameters of the linear dynamic part are estimated by the linear least-squares method. The adjusting parameters for the ACF model, i.e. the number and widths of the subdomains and the shape of the ACF are properly determined by using the GA, in which the Akaike information criterion is utilized as the fitness value function. Simulation results are shown to illustrate the effectiveness of the proposed method.

1. Introduction

Since most practical systems have inherent nonlinear characteristics such as saturation and dead-zone, the problem of identifying such kind of systems is of great importance for precise analysis, synthesis and prediction. One of approaches for nonlinear system identification is use of the block oriented models such as Hammerstein model or Wiener model [1, 2, 3]. The Hammerstein model is expressed by a nonlinear static part followed by a linear dynamic part. The model has many advantages for control design or stability analysis due to the model structure [3]. Several identification methods have been proposed for the Hammerstein model by using correlation theory [4], neural networks [5], polynomials [6], piecewise linear model [7], and so on. However in many cases the model structure for representation of nonlinear static part is assumed to be known.

In this paper an application of automatic choosing function (ACF) model [8] and genetic algorithm [9] is proposed for identification of Hammerstein type nonlinear systems. The data region of the input signals is divided into some subdomains. Unknown nonlinear static part to be estimated is approximately represented by a local linear equation on each subdomain. These local linear equations are united into a single one by the ACF smoothly. The connection coefficients of the ACF and the system parameters of the linear dynamic part are estimated by the linear least-squares method. The accuracy of this identification method depends strongly on the ACF model structure, i.e. the number and widths of the subdomains and the shape of the ACF. These adjusting parameters are properly determined by using the GA, which is a probabilistic search procedure based on the mechanics of natural selection and natural genetics [9]. The fitness value in this GA is calculated by the Akaike information criterion (AIC) [10].

This paper is organized as follows. In section 2 the problem is formulated. In section 3 the identification method is proposed in case of fixed ACF model structure. In section 4 the GA is applied to determine the ACF model structure. In section 5 simulation results are shown to illustrate the effectiveness of the proposed method. Finally some conclusions are remarked in section 6.

2. Statement of the Problem

Consider a discrete-time nonlinear system described by the Hammerstein model shown in Figure 1:

$$\begin{cases} A(q^{-1})y(k) = B(q^{-1})x(k-1) + e(k) \\ x(k) = f(u(k)) \\ A(q^{-1}) = 1 + a_1 q^{-1} + \cdots + a_n q^{-n} \\ B(q^{-1}) = b_0 + b_1 q^{-1} + \cdots + b_r q^{-r} \end{cases} \tag{1}$$

where $u(k)$ and $y(k)$ are input and output signals, respectively. $x(k)$ is intermediate signal that is not accessible for measurement. $e(k)$ is measurement noise. q^{-1} denotes backward shift operator. n and r are known degrees of polynomials $A(q^{-1})$ and $B(q^{-1})$, respectively. $f(\cdot)$ is unknown nonlinear function. The problem is to identify the system

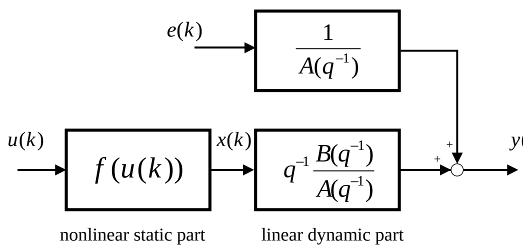

Figure 1: Hammerstein model

parameters $\{a_i\}$ and $\{b_j\}$ of the linear dynamic part, and nonlinear static function $f(\cdot)$ from input and output data.

3. Identification

In order to represent the nonlinear function $f(\cdot)$, the sigmoid type ACF [8] is introduced.

Let a domain being a data region of $u(k)$ be $D = [u_{min}, u_{max}]$. The domain D is divided into some subdomains of $D = \cup_{i=1}^{M} D_i$ where $D_i = [\alpha_i, \beta_i]$, $\alpha_1 = u_{min}$, $\beta_M = u_{max}$, $\alpha_k = \beta_{k-1}$ $(k = 2, 3, \cdots, M)$. Then the ACF is defined by

$$I_i(u(k)) = 1 - \frac{1}{1 + \exp(H(u(k) - \alpha_i))} - \frac{1}{1 + \exp(-H(u(k) - \beta_i))} \tag{2}$$

where H is positive real value. $I_i(u(k))$ is almost unity only on a subdomain $D_i = [\alpha_i, \beta_i]$ and nearly equals to zero on $D - D_i$, so it chooses D_i automatically (see Figure 2).

Assume that $f(u(k))$ is well approximated linearly on each subdomain D_i:

$$f(u(k)) \simeq c_i + d_i u(k) \quad \text{on } D_i. \tag{3}$$

Then $f(u(k))$ is represented by using the ACF on the whole domain D as

$$f(u(k)) = \sum_{i=1}^{M} (c_i + d_i u(k)) I_i(u(k)) + \epsilon(k) \quad \text{on } D, \tag{4}$$

where $\epsilon(k)$ is an approximation error.

Substituting Eq.(4) into Eq.(1) yields

$$A(q^{-1})y(k) = \sum_{i=1}^{M} c_i B(q^{-1}) I_i(u(k-1)) + \sum_{i=1}^{M} d_i B(q^{-1}) u(k-1) I_i(u(k-1)) + v(k) \tag{5}$$

or in vector form,

$$y(k) = \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta} + v(k) \tag{6}$$

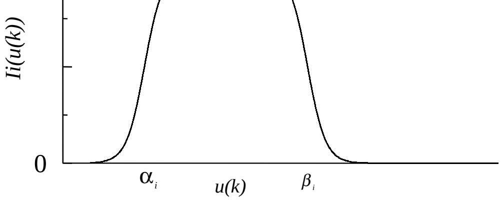

Figure 2: Automatic choosing function (ACF)

where $v(k) = e(k) + B(q^{-1})\epsilon(k-1)$ is an equation error, and

$$\begin{cases} \boldsymbol{\theta} = [\boldsymbol{\theta}_a^{\mathrm{T}}, \boldsymbol{\theta}_{c_1}^{\mathrm{T}}, \boldsymbol{\theta}_{c_2}^{\mathrm{T}}, \cdots, \boldsymbol{\theta}_{c_M}^{\mathrm{T}}, \boldsymbol{\theta}_{d_1}^{\mathrm{T}}, \boldsymbol{\theta}_{d_2}^{\mathrm{T}}, \cdots, \boldsymbol{\theta}_{d_M}^{\mathrm{T}}]^{\mathrm{T}} \\ \boldsymbol{\theta}_a = [a_1, a_2, \cdots, a_n]^{\mathrm{T}} \\ \boldsymbol{\theta}_{c_i} = [\theta_{c_i}(1), \theta_{c_i}(2), \cdots, \theta_{c_i}(r+1)]^{\mathrm{T}} \\ \quad = [b_0 c_i, b_1 c_i, \cdots, b_r c_i]^{\mathrm{T}} \\ \boldsymbol{\theta}_{d_i} = [\theta_{d_i}(1), \theta_{d_i}(2), \cdots, \theta_{d_i}(r+1)]^{\mathrm{T}} \\ \quad = [b_0 d_i, b_1 d_i, \cdots, b_r d_i]^{\mathrm{T}} \end{cases}$$

$$\begin{cases} \boldsymbol{\varphi}(k) = [\boldsymbol{\varphi}_a^{\mathrm{T}}(k), \boldsymbol{\varphi}_{c_1}^{\mathrm{T}}(k), \boldsymbol{\varphi}_{c_2}^{\mathrm{T}}(k), \cdots, \boldsymbol{\varphi}_{c_M}^{\mathrm{T}}(k), \\ \qquad \boldsymbol{\varphi}_{d_1}^{\mathrm{T}}(k), \cdots, \boldsymbol{\varphi}_{d_M}^{\mathrm{T}}(k)]^{\mathrm{T}} \\ \boldsymbol{\varphi}_a(k) = [-y(k-1), -y(k-2), \cdots, -y(k-n)]^{\mathrm{T}} \\ \boldsymbol{\varphi}_{c_i}(k) = [I_i(u(k-1)), I_i(u(k-2)), \cdots, \\ \qquad I_i(u(k-r-1))]^{\mathrm{T}} \\ \boldsymbol{\varphi}_{d_i}(k) = [u(k-1)I_i(u(k-1)), u(k-2)I_i(u(k-2)), \\ \qquad \cdots, u(k-r-1)I_i(u(k-r-1))]^{\mathrm{T}} \end{cases}$$

$$(i = 1, 2, \cdots, M). \tag{7}$$

Each parameter will be estimated as follows.

First, the unknown parameter vector $\boldsymbol{\theta}$ is easily evaluated by applying the linear least-squares method to Eq.(6):

$$\hat{\boldsymbol{\theta}} = \left[\sum_{k=N_s+1}^{N_s+N} \boldsymbol{\varphi}(k)\boldsymbol{\varphi}^T(k)\right]^{-1} \left[\sum_{k=N_s+1}^{N_s+N} \boldsymbol{\varphi}(k)y(k)\right] \tag{8}$$

where N is the number of input and output data. Thus the parameters of the linear dynamic part are estimated by

$$[\hat{a}_1, \cdots, \hat{a}_n, \hat{b}_0, \cdots, \hat{b}_r]^{\mathrm{T}} = \left[\mathbf{I}_{(n+r+1)\times(n+r+1)} \,\vdots\, \mathbf{0}\right]\hat{\boldsymbol{\theta}}, \tag{9}$$

putting $c_1 = 1$ without loss of generality.

Next, the parameters of the nonlinear static part are obtained by using the linear least-squares technique again as

$$\begin{aligned} \hat{c}_i &= \sum_{j=1}^{r+1} \hat{\theta}_{c1}(j)\hat{\theta}_{ci}(j) / \sum_{j=1}^{r+1} \hat{\theta}_{c1}^2(j) \quad (i = 2, 3, \cdots, M) \\ \hat{d}_i &= \sum_{j=1}^{r+1} \hat{\theta}_{c1}(j)\hat{\theta}_{di}(j) / \sum_{j=1}^{r+1} \hat{\theta}_{c1}^2(j) \quad (i = 1, 2, \cdots, M). \end{aligned} \tag{10}$$

Thus the nonlinear static function is composed by Eq.(10) as

$$\hat{f}(u(k)) = \sum_{i=1}^{M} (\hat{c}_i + \hat{d}_i u(k)) I_i(u(k)). \tag{11}$$

4. Optimization of ACF Model Structure by GA

The accuracy of the above identification algorithm greatly depends on the ACF model structure, i.e. the number and widths of the subdomains and the shape of the ACF. The number M of the subdomains should be determined properly in order to avoid overparametrization and reduce the complexity of the estimated model. In this section the AIC is utilized as an objective function, and the ACF model structure is determined by the GA. Note that if candidates of M, α_i $(i = 2, 3, \cdots, M)$ and H are given, then the unknown parameter vector $\boldsymbol{\theta}$ is estimated by the algorithm of section 3. Therefore only $\boldsymbol{\Omega} = (M, \{\alpha_i\}, H)$ is coded into binary bit strings and searched by the GA.

First, an initial population which consists of binary bit strings as candidates of $\boldsymbol{\Omega}$ is generated. Then, candidates of the ACF are constructed by using the decoded values from the strings. The candidates of unknown parameter vector $\boldsymbol{\theta}$ are estimated by the identification method described in section 3. The fitness values are calculated by using the AIC. The genetic operations, which are reproduction based on the fitness values, crossover and mutation, are repeated so that the fitness value of the population increases.

In more detail the algorithm is as follows:

step 1: Initialization

Generate an initial population of Q binary bit strings for $\boldsymbol{\Omega}$ randomly.

step 2: Decoding

Decode Q strings into real values $\hat{\boldsymbol{\Omega}}_i$ $(i = 1, 2, \cdots, Q)$.

step 3: Construction of ACF

Construct Q candidates of the ACF using $\hat{\boldsymbol{\Omega}}_i$ $(i = 1, 2, \cdots, Q)$.

step 4: Identification

Identify $\hat{\boldsymbol{\theta}}_i$ and $\hat{f}_i(u(k))$ $(i = 1, 2, \cdots, Q)$ from Eqs.(8)~(11), using each candidates of the ACF.

step 5: Fitness value calculation

Calculate the AIC:

$$AIC_i = N \log \left\{ \frac{1}{N} \sum_{k=N_s+1}^{N_s+N} (y(k) - \hat{y}_i(k))^2 \right\} + 2P_i \qquad (i = 1, 2, \cdots, Q) \tag{12}$$

and the fitness values $F_i = -AIC_i$, using $\hat{\boldsymbol{\Omega}}_i$, $\hat{\boldsymbol{\theta}}_i$ and $\hat{f}_i(u(k))$. $P_i = n + 2M_i(r + 1)$ is the number of the parameters in the identification model Eq.(6). $\hat{y}_i(k)$ is the output of the estimated model.

step 6: Reproduction

Reproduce each of individual strings with the probability of $F_i / \sum_{j=1}^{Q} F_j$. Practically, the linear fitness scaling [9] is utilized to avoid undesirable premature convergence.

step 7: Crossover

Pick up two strings randomly and decide whether or not to cross them over according to the crossover probability P_c. Exchange strings at a crossing position if the crossover is required. The crossing position is chosen randomly.

step 8: Mutation

Alter a bit of string (0 or 1) according to the mutation probability P_m.

step 9: Repetition

Repeat step 2 ∼ step 8 from generation to generation so that the fitness value of the population increases. In simulations, the genetic operations will be repeated until prespecified G-th generation.

Finally, at the termination of this algorithm, the suboptimal parameters of the ACF model $\hat{\boldsymbol{\Omega}}_{best}$ is determined by the string with the best fitness value over all the past generations. So the final estimated model is constructed by $\hat{\boldsymbol{\Omega}}_{best}$, and the corresponding $\hat{\boldsymbol{\theta}}_{best}$ and $\hat{f}_{best}(u(k))$.

5. Numerical Simulation

Consider a system described by

$$\begin{cases} A(q^{-1})y(k) = B(q^{-1})x(k-1) + e(k) \\ x(k) = f(u(k)) \\ \quad = \begin{cases} -2.0 & (-3.0 \le u(k) < -1.8) \\ u(k)/0.6 + 1.0 & (-1.8 \le u(k) < -0.6) \\ 0.0 & (-0.6 \le u(k) < 0.6) \\ u(k)/0.6 - 1.0 & (0.6 \le u(k) < 1.8) \\ 2.0 & (1.8 \le u(k) \le 3.0) \end{cases} \\ A(q^{-1}) = 1 + 0.8q^{-1} + 0.6q^{-2} \\ B(q^{-1}) = 0.4 + 0.2q^{-1} \end{cases} \tag{13}$$

This system has saturation and dead-zone nonlinearity.

The output signal is generated by uniformly distributed input with amplitude range $[-3.0, 3.0]$. $e(k)$ is white Gaussian noise $N(0, 0.01)$. The number of input and output data is $N = 300$. The maximum number of the subdomain is $M_{max} = 11$. The design parameters of the GA are empirically chosen as follows:

(1) population size: $Q = 30$
(2) all bit number of string: $L = 180$
(3) search range of H: $[h_{min}, h_{max}] = [0.1, 100.0]$
(4) crossover probability: $P_c = 0.8$
(5) mutation probability: $P_m = 0.03$

The genetic operations are repeated until $G = 150$-th generation.

The adjusting parameters of the ACF model have been determined by the GA as $\hat{\boldsymbol{\Omega}} = (\hat{M}, \{\hat{\alpha}_i\}, \hat{H}) = (8, \{-2.06, -1.89, -1.03, -0.52, 0.00, 0.74, 1.61\}, 99.30)$.

The estimated nonlinear static function $\hat{f}_1(u(k))$ by the proposed method is shown in Figure 3. For comparison, the estimated nonlinear static function $\hat{f}_2(u(k))$ is also shown in Figure 3, where the conventional polynomial model (8th order) is used to represent the nonlinear static part [3, 6]. Clearly $\hat{f}_1(u(k))$ by the proposed method is very close to the true nonlinear function $f(u(k))$ on the given data region, while $\hat{f}_2(u(k))$ by the polynomial model has larger error to the true nonlinear function.

Figure 4 shows the true output $y(k)$, the output $\hat{y}_1(k)$ of the estimated model by the proposed method and $|y(k) - \hat{y}_1(k)|$. Similarly, the output $\hat{y}_2(k)$ by the polynomial model and $|y(k) - \hat{y}_2(k)|$ are also shown in Figure 4. The output errors are $(\sum_{k=N_s+1}^{N_s+N} |y(k) - \hat{y}_1(k)|)/N = 0.124$ for the proposed method and $(\sum_{k=N_s+1}^{N_s+N} |y(k) - \hat{y}_2(k)|)/N = 0.138$ for the polynomial model, respectively. From these results, we confirm that the accuracy of the proposed method is superior to that of the conventional polynomial model.

Estimates of the system parameters of the linear dynamic part are shown in Table 1.

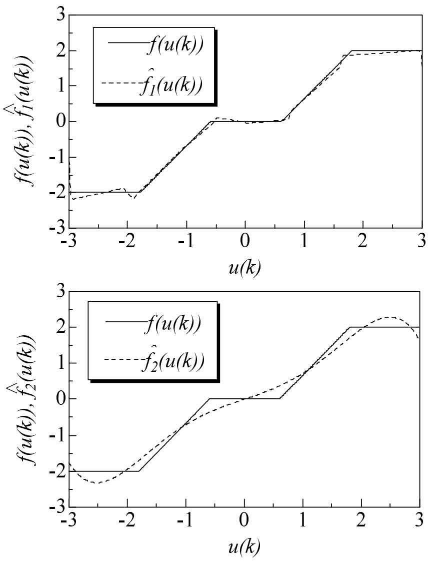

Figure 3: True nonlinear function and estimated nonlinear functions

Table 1: System parameters of the linear dynamic part

	true values	estimates (proposed)	estimates (polynomial model)
a_1	0.8	0.823	0.801
a_2	0.6	0.596	0.590
b_0	0.4	0.400	0.400
b_1	0.2	0.237	0.186

6. Conclusions

In this paper an identification method of the Hammerstein systems using the ACF model and GA has been proposed. The nonlinear static part to be estimated is approximately represented by the ACF model. Then the linear least squares method has been applied to estimate the connection coefficients of the ACF and the system parameters of the linear dynamic part. The adjusting parameters for the ACF model, i.e. the number and widths of the subdomains and the shape of the ACF have been properly determined by the GA. The GA would be one of suitable methods for such complicated optimization problems. Simulation results show that the identification by this method is easy in computation and superior in accuracy even in the presence of low measurement noises.

Acknowledgments

The authors would like to thank Mr. Katsuhisa Deguchi, who was a graduate student of Kagoshima University, for his help in numerical simulations.

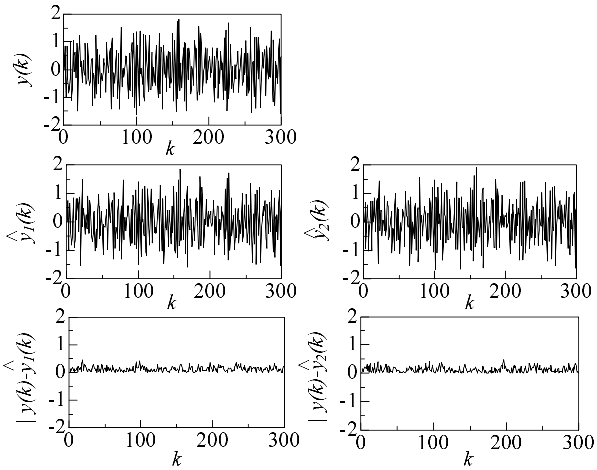

Figure 4: True output, outputs by the estimated models, and differences between them

References

[1] S. A. Billings, “Identification of Nonlinear Systems - A Survey”, *IEE Proc. Pt. D*, vol.127, no.6, pp.272–285, 1980.

[2] R. Haber and H. Unbehauen, “Structure Identification of Nonlinear Dynamic Systems - A Survey on Input/Output Approaches”, *Automatica*, vol.26, no.4, pp.651–677, 1990.

[3] O. Nelles, “Nonlinear System Identification”, Springer, 2000.

[4] S. A. Billings and S. Y. Fakhouri, “Identification of Systems Containing Linear Dynamic and Static Nonlinear Elements”, *Automatica*, vol.18, no.1, pp.15–26, 1982.

[5] H. Al-duwaish and M. N. Karim, “A New Method for the Identification of Hammerstein Model”, *Automatica*, vol.33, no.10, pp.1871–1875, 1997.

[6] S. Adachi and H. Murakami, “Generalized Predictive Control System Design Based on Non-linear Identification by Using Hammerstein Model (in Japanese)”, *Trans. of the Institute of Systems, Control and Information Engineers*, vol.8, no.3, pp.115–121, 1995.

[7] T. Hatanaka, K. Uosaki and M. Koga, “Evolutionary Computation Approach to Hammerstein Model Identification”, *Proc. of the 4th Asian Control Conference*, pp.1730–1735, 2002.

[8] H. Takata, “An Automatic Choosing Control for Nonlinear Systems”, *Proc. of the 35th IEEE CDC*, pp.3453–3458, 1996.

[9] D. E. Goldberg, “Genetic Algorithms in Search, Optimization, and Machine Learning”, Addison-Wesley Publishing Company, Inc., 1989.

[10] H. Akaike, “A New Look on the Statistical Model Identification”, *IEEE Trans. on Automatic Control*, vol.19, pp.716–723, 1974.