METAHEURISTIC APPROACH IN NONLINEAR SYSTEMS IDENTIFICATION

Mihai CORNOIU[1], Cătălin BÂRA[2], Dumitru POPESCU[3]

In this paper is presented a Hammerstein systems identification algorithm. The algorithm uses Particle Swarm Optimization firstly to approximate the nonlinear component, also using sigmoid type functions, and, secondly, to estimate the linear component's parameters and the nonlinear functions connection coefficients, by solving a standard least squares problem. Due to the nature of Hammerstein systems, Particle Swarm Optimization was adapted with respect to specific constraints, which are detailed in this article. Numerical results confirm the accuracy of this proposed identification method.

Keywords: nonlinear, modeling, identification, optimization, parameter estimation

1. Introduction

It is a widely known fact that nonlinearities are generic in nature; hence almost all processes are nonlinear if they are considered not merely in a small vicinity of their working points. Also, the development of the industrial equipment and the desire to use it at its full potential has generated a need to create mathematical models, which can describe the global nonlinear behavior of the process. Identification is a powerful tool which allows such breakthroughs, by experimentally determining the structure and the parameters of the mathematical models, if the process is unknown, or if the describing equations are too complex [1].

The two main directions in describing nonlinear systems behavior consist in either using nonparametric models, which theoretically need an infinite numbers of parameters, therefore they are restricted suitably only for

[1] PhD Student, Automatic Control and Systems Engineering Department, Automatic Control and Computers Faculty, University POLITEHNICA of Bucharest, Romania, e-mail: mihai.cornoiu@acse.pub.ro

[2] PhD Student, Automatic Control and Systems Engineering Department, Automatic Control and Computers Faculty, University POLITEHNICA of Bucharest, Romania, Romania, e-mail: bara.catalin@gmail.com

[3] Prof. Automatic Control and Systems Engineering Department, Automatic Control and Computers Faculty, University POLITEHNICA of Bucharest, Romania, e-mail: dpopescu@indinf.pub.ro

identification and control purposes, either by parametric models, which can be described by a finite number of parameters [2]. A widely used category of parametric models is represented by block oriented models, such as Hammerstein models, which are separable models consisting of a static nonlinear element followed by a linear dynamic component. Literature proposes several identification methods suitable for Hammerstein systems: [3] use instrumental variables methods; [4] estimates the parameters for systems having piecewise-linear nonlinearities with asymmetric dead-zones; [5], [6] use correlation analysis to identify the block components, [7] use subspace methods, [8] use Bezier curves and Bernstein polynomials, and examples could continue. An important aspect in Hammerstein systems identification consists in the nonlinear function approximation. Many methods have been proposed in the literature; for example, [9] uses single input rule modules connected fuzzy inference model (SIRMs model) and [10] tackles the problematic of sparse function approximation. However, [11] uses sigmoid type functions to approximate the static nonlinear component, thus defining the concept of Automatic Choosing Function. The idea behind this concept consists in separating the input signal's corresponding data region into subdomains and approximating each one with a linear function. The ACF will consist in the junction of these local linear functions, which are smoothly connected using appropriate coefficients. Connection coefficients and linear component parameters are estimated by using linear least-squares techniques. Still, it is important to note that the approximation error is strongly connected with the ACF parameters, namely the choice of the partitioning intervals and the shape of the ACF. A way for determining these parameters is given by nature, namely by using a stylized representation of the movement of organisms in a bird flock or fish school. [12] developed the Particle Swarm Optimization method, a metaheuristic, as it makes few or no assumptions about the problem being optimized and can search very large spaces of candidate solutions. Hence, by using PSO and adding some problem specific restrictions to it, the systems parameters will be successfully estimated. The measure of quality will be the evaluation criterion, which is given by the mean square errors between the outputs of the real systems and of the estimated model.

This paper is organized as follows. The general statement of the problem and the ACF concept are described in section 2. Section 3 consists in the description of the identification method. Section 4 handles the optimization problem, namely adjusting the parameters of ACF by using PSO. Numerical results that confirm the accuracy of the algorithm are presented in section 5. Conclusions are drawn in section 6.

2. Statement of the problem

A Hammerstein system, as depicted in Fig. 1, consists of a static nonlinear part, represented by $f(\cdot)$ function, and of a dynamical linear component, described by known degrees polynomials $A(q^{-1})$ and $B(q^{-1})$. The problem is finding the system parameters a_i and b_j, $i=1:n_A, j=0:n_B$, where n_A, n_B are the corresponding degrees of polynomials $A(q^{-1})$ and $B(q^{-1})$, and approximating the nonlinear component $f(\cdot)$, using only input data, $u(k)$, and output data, $y(k)$. The intermediate signal, $x(k)$, is not accessible to measurements, and $e(k)$ represents the measurement noise.

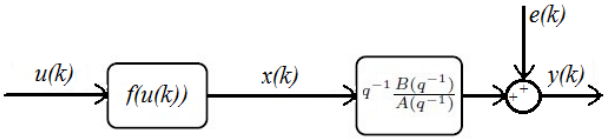

Fig. 1. Hammerstein System

The characteristic equations describing the Hammerstein system are:

$$\begin{cases} A(q^{-1})y(k) = B(q^{-1})x(k-1) + A(q^{-1})e(k) \\ x(k) = f(u(k)) \\ A(q^{-1}) = 1 + a_1 q^{-1} + a_2 q^{-2} + \cdots + a_{n_A} q^{-n_A} \\ B(q^{-1}) = b_0 + b_1 q^{-1} + b_2 q^{-2} + \cdots + b_{n_B} q^{-n_B} \end{cases} \tag{1}$$

The nonlinear function $f(\cdot)$ is approximated using sigmoid type Automatic Choosing Functions, which are defined as follows:

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

where $H \in \mathbb{R}_+$. Considering the data region corresponding to the input signal $u(k)$ to be $D=[u_{\min}, u_{\max}]$, this domain is divided into M partitions such that $D = \bigcup_{i=1}^{M} D_i$. Each D_i can be expressed as $[\alpha_i, \beta_i]$, where $\alpha_1 = u_{\min}, \beta_1 = u_{\max}, \alpha_{i+1} = \beta_i, i = 1, \cdots, M$. Due to its nature, $I_i(u(k))$ is almost unity only for input signals values $u(k) \in [\alpha_i, \beta_i]$, and is almost zero for values that do not fit the specific partition.

Fig. 2 presents the shape of the ACF obtained using the following parameters: $u(k)$ is a 1000 element uniformly distributed signal with amplitude range of $[0,10]$, $\alpha = 2, \beta = 8$ and $H = \{10, 50, 100, 500\}$. It can be noticed that by

increasing the value of H, the shape of the ACF approximates a block-pulse function.

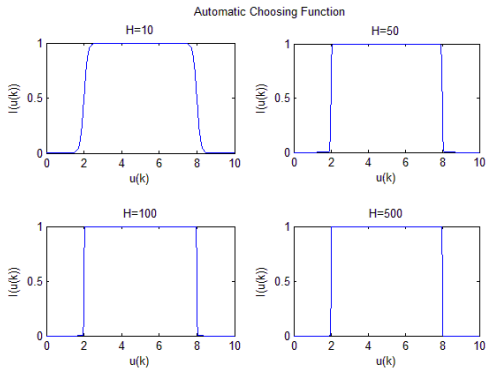

Fig. 2. Shape of Automatic Choosing Function (ACF)

3. Identification Procedure

Assuming that the nonlinear function $f(u(k))$ is approximated linearly on each partition D_i as follows:

$$f_i(u(k)) \cong c_i + d_i u(k), \tag{3}$$

we obtain the representation on the entire domain D:

$$f(u(k)) = \sum_{i=1}^{M} f_i(u(k)) I_i(u(k)) + \varepsilon(k), \tag{4}$$

where, $\varepsilon(k)$ is the approximation error. By substituting equations (3) and (4) in the Hammerstein model recurrent equations (1) we obtain:

$$\begin{aligned} A(q^{-1})y(k) &= B(q^{-1})\sum_{i=1}^{M} f_i(u(k-1)) I_i(u(k-1)) + \\ &+ A(q^{-1})e(k) + B(q^{-1})\varepsilon(k-1), \end{aligned} \tag{5}$$

or

$$\begin{aligned} y(k) &= a_1 y(k-1) + a_2 y(k-2) + \cdots + a_{n_A} y(k-n_A) + \\ &+ b_0 \sum_{i=1}^{M} f_i(u(k-1)) I_i(u(k-1)) + \cdots + \\ &+ b_{n_B} \sum_{i=1}^{M} f_i(u(k-n_B-1)) I_i(u(k-n_B-1)) + \upsilon(k), \end{aligned} \tag{6}$$

where

$$\upsilon(k) = A(q^{-1})e(k) + B(q^{-1})\varepsilon(k-1) \tag{7}$$

represents the total approximation error. Equation (6) reveals the classic form:

$$y(k)=\varphi^T(k)\theta+\upsilon(k), \tag{8}$$

with:

$$\begin{cases}\theta=[\theta_a^T,\theta_{c_1}^T,\theta_{c_2}^T,\cdots,\theta_{c_M}^T,\theta_{d_1}^T,\theta_{d_2}^T,\cdots,\theta_{d_M}^T]^T\\ \theta_a^T=[a_1,a_2,\cdots,a_{n_A}]^T\\ \theta_{c_i}^T=[b_0c_i,b_1c_i,\cdots,b_{n_B}c_i]^T\\ \theta_{d_i}^T=[b_0d_i,b_1d_i,\cdots,b_{n_B}d_i]^T\end{cases} \tag{9}$$

$$\begin{cases}\varphi(k)=[\varphi_a^T(k),\varphi_{c_1}^T(k),\varphi_{c_2}^T(k),\cdots,\varphi_{c_M}^T(k),\varphi_{d_1}^T(k),\varphi_{d_2}^T,(k)\cdots,\varphi_{d_M}^T(k)]^T\\ \varphi_a(k)=[-y(k-1),-y(k-2),\cdots,-y(k-n_A)]^T\\ \varphi_{c_i}(k)=\begin{bmatrix}I_i(u(k-1))\\ I_i(u(k-2))\\ \vdots\\ I_i(u(k-n_B-1))\end{bmatrix}\\ \varphi_{d_i}(k)=\begin{bmatrix}u(k-1)I_i(u(k-1))\\ u(k-2)I_i(u(k-2))\\ \vdots\\ u(k-n_B-1)I_i(u(k-n_B-1))\end{bmatrix}\end{cases} \tag{10}$$

where $i=1,\cdots,M$.

Equation (8) also permits evaluating the unknown parameter vector θ, by using the linear least-squares algorithm, for example:

$$\hat{\theta}=[\sum\varphi(k)\varphi^T(k)]^{-1}[\sum\varphi(k)y(k)] \tag{11}$$

The unknown parameter vector can be rewritten in the form:

$$\hat{\theta}=[\hat{\theta}_a^T,(\hat{\theta}_b\otimes\hat{\theta}_c)^T,(\hat{\theta}_b\otimes\hat{\theta}_d)^T]^T \tag{12}$$

where "$\otimes$" denotes the Kronecker tensor product, and:

$$\begin{cases}\hat{\theta}_b=[\hat{b}_0,\hat{b}_1,\cdots,\hat{b}_{n_B}]^T\\ \hat{\theta}_c=[\hat{c}_1,\hat{c}_2,\cdots,\hat{c}_M]^T\\ \hat{\theta}_d=[\hat{d}_1,\hat{d}_2,\cdots,\hat{d}_M]^T\end{cases} \tag{13}$$

Upon analyzing equation (12), it becomes obvious that the first n_A elements of $\hat{\theta}$ provide exactly the estimation for the A polynomial parameters (linear dynamic component parameters), or $\hat{\theta}_a$. However, determining the exact

values of the ACF approximation parameters and of the *B* polynomial of the linear dynamic part is impossible, since vector $\hat{\theta}$ only contains information about the products of these parameters. Still, a procedure which permits the evaluation of vectors $\hat{\theta}_b, \hat{\theta}_c$ and $\hat{\theta}_d$ has been developed, and it is presented in the following.

Keeping in mind that the unknown parameter vector $\hat{\theta}$ contains information from all *M* partitions, one idea is to count the number of appearances of the input signal corresponding to each partition and storing them in a weight vector *w*. Upon having this information, the next step is to repeatedly assign each element of $\hat{\theta}_b$ the value *1*, thus permitting the evaluation of $\hat{\theta}_c$ and $\hat{\theta}_d$ using again the least-squares technique, but with the imposed *w* weights, and estimating the remaining parameters of $\hat{\theta}_b$. At the end of this procedure, the values for the *B* polynomial estimated coefficients are obtained by averaging the estimations previously provided for each coefficient.

Since we now have $\hat{\theta}_b$, by applying again the least-squares technique we estimate the ACF connection coefficients, namely vectors $\hat{\theta}_c$ and $\hat{\theta}_d$, which allows us to write the approximation equation for the nonlinear function as:

$$\hat{f}(u(k)) = \sum_{i=1}^{M} (\hat{c}_i + \hat{d}_i u(k)) I_i(u(k)) \tag{14}$$

4. Particle swarm optimization of model

As stated, there exists an important interdependence between the accuracy of the algorithm and the partitioning intervals chosen to approximate the nonlinear function. [13] use the metaheuristic Particle Swarm Optimization algorithm to determine these partitioning intervals. A variation of their method, that considers an additional constrain (such that the partition intervals should be disjoint and the partitioning points, α_i, should satisfy $\alpha_{i-1} \le \alpha_i, i = 1, \cdots M$) is presented in this current section.

The proposed algorithm, an iterative procedure, follows the next steps:

Step 1. Generation of an initial population of Q particles with random positions and velocities.

Each particle now has an initial position X_i^0 and velocity V_i^0, where $X = [\{\alpha_j\}, H], j = 1, ..., M$ and $i = 1, \cdots, Q$.

Step 2. Construction of candidates for ACF

By using current particles positions, which contain information concerning the partitioning intervals ($\{\alpha_j\}, j=1,...,M$) and the shape of the ACF (*H*), by using equation (2) we construct $I_i(u(k))$ for each partition.

Step 3. Running the identification procedure

Once the candidates for ACF are constructed, the next step is to estimate the unknown parameters vectors, $\hat{\theta}_a, \hat{\theta}_b, \hat{\theta}_c$ and $\hat{\theta}_d$, as described in Section 4.

Step 4. Evaluation of performance criterion

The imposed performance criterion (or cost function) relies on the quadratic approximation error,

$$J(X_i)=\frac{1}{N}\sum_{k=1}^{N}(y(k)-\hat{y}_i(k))^2, \tag{15}$$

where $\hat{y}_i(k)=\varphi_i^T(k)\hat{\theta}_i$ is the output signal corresponding to each candidate of the estimated model, and *N* represents the size of input and output data vectors.

Step 5. Updating each particle's personal best position (pbest), and the global best position among all particles (gbest)

For the first iteration, it is considered that each particle is in its personal best position. The global best, *gbest*, is chosen as the personal best position of the particle which has provided the smallest value for the cost function, as computed in Step 4. For the other iterations, the following formulas are used:

$$pbest_i^l=\begin{cases}X_i^l, & (J(X_i^l)<J(pbest_i^{l-1}))\\ pbest_i^{l-1}, & (otherwise)\end{cases} \tag{16}$$

$$gbest^l=pbest_{i_{best}}^l, i_{best}=\arg\min_i J(pbest_i^l) \tag{17}$$

Step 6. Updating particles positions and velocities

$$\begin{cases}V_i^{l+1}=w\cdot V_i^l+c_1\cdot rand_1\cdot(pbest_i^l-X_i^l)+c_2\cdot rand_2\cdot(gbest^l-X_i^l)\\ X_i^{l+1}=X_i^l+V_i^{l+1}\end{cases} \tag{18}$$

The above formula represents the standard expression for computing the velocities of the particles and their future positions. However, due to the particular nature the particle vector has (the thresholds must be in increasing order), the standard PSO procedure is modified, as follows. Each particle component should not pass its neighbor as it updates its position, so the corresponding velocity is set as the modulus between the primary velocity (as without the partitions limit constraint) and the subdomains length. Also, if the velocity is negative, it is treated the same as before, keeping in mind that the minimal value for the first threshold should not be smaller than the minimum of the input signal. An example of how this procedure works is shown in the next set of figures.

Fig. 3. Initial particle positioning

In Fig. 3 is represented a particle which contains 5 elements, for example the first 5 thresholds that determine the partitions.

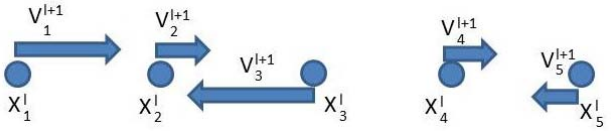

Fig. 4. Particle velocities

Each element has its own unique velocity, either positive or negative.

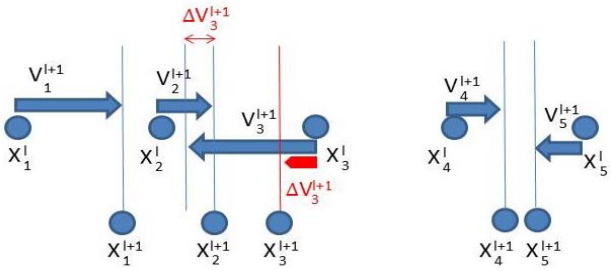

Fig. 5. Future particle positioning (using standard PSO)

Considering the velocities in this example, element 2 should overpass the third one, thus violating the boundary constraint. Hence, the velocity of element 3 is recomputed, as well as its future position.

Fig. 6. Future particle positioning (using modified PSO)

Final particle positioning, no overpasses between any two neighboring elements had occurred.

Step 7. Incrementing the iteration counter and returning to Step 2 until termination criterion is satisfied.

The termination criterion could consist in either reaching preset maximum number of iteration, either obtaining smaller value for the cost function than a pre-specified. Once the termination criterion is satisfied, the algorithm will return the suboptimal solution for choosing the partitioning intervals and the shape of the

ACF. These parameters allow the estimation of the static nonlinear and linear dynamic components with enhanced accuracy.

5. Numerical examples

The algorithm was implemented in Matlab and simulated using Simulink environment; this section contains the obtained numerical results, which confirms its accuracy. Note that numerical results that are presented in Section 6 are normalized with b_0.

Let us consider the discrete time Hammerstein system

$$\begin{cases} A(q^{-1})y(k) = B(q^{-1})x(k-1) + A(q^{-1})e(k) \\ x(k) = 2 \cdot u(k) + u^2(k) + 0.4 \cdot u^3(k) \\ A(q^{-1}) = 1 + 0.4 \cdot q^{-1} + 0.2 \cdot q^{-2} \\ B(q^{-1}) = 0.5 + 0.1q^{-1} \end{cases} \tag{19}$$

The considered input signal is uniformly distributed with amplitude range [*-2.0, 2.0*]; *e(k)* is considered white Gaussian noise with signal-to-noise ratio of 1%, since simulation has revealed that increased noise power greatly affects the accuracy of the results. The input and output data vectors length is $N = 300$, the number of partitions is set $M = 10$. The input parameters of the PSO are: particle size of *20*; maximum iteration limit of *50*; inertia factor and acceleration coefficients of *0.8*. Results (also shown in Figures 7-9):

$$\begin{cases} A(q^{-1}) = 1 + 0.4018 \cdot q^{-1} + 0.1965 \cdot q^{-2} \\ B(q^{-1}) = 0.5 + 0.1012q^{-1} \end{cases}$$

The adjusting parameters of the ACF are: $\{\alpha_i\}$ = *{-1.7332, -1.7244, -1.5611, -0.5762, -0.4458, 0.4023, 1.1326, 1.1791, 1.2239, 1.2245}*; $H = 3.7825$. The obtained quadratic approximation error was $2.8575 \cdot 10^{-3}$.

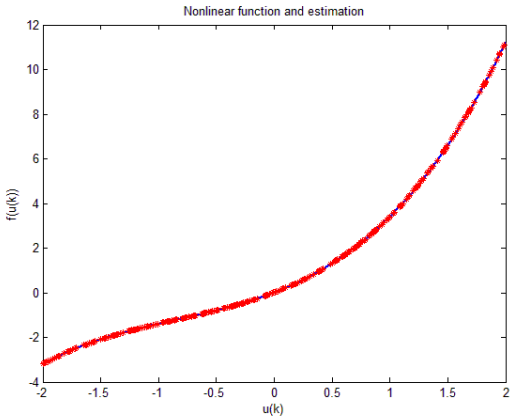

Fig. 7. Estimated nonlinear function

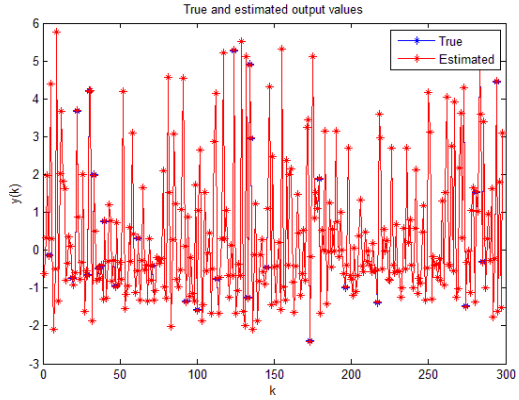

Fig. 8. True and estimated outputs

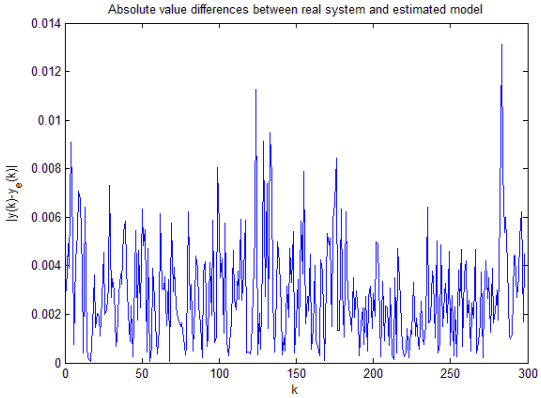

Fig. 9. Differences between true and estimated output values

By replacing the previously chosen nonlinear function with a sinusoid, $x(k) = \sin(2 \cdot \pi \cdot u(k))$, the obtained results (see Figures 10-12) are the following:

$$\begin{cases} A(q^{-1}) = 1 + 0.3982 \cdot q^{-1} + 0.1987 \cdot q^{-2} \\ B(q^{-1}) = 0.5 + 0.0981 q^{-1} \end{cases}$$

The adjusting parameters of the ACF determined by the PSO are: $\{\alpha_i\}$ = *{-1.2951, -1.2944, -1.2925, -0.4884, -0.3263, -0.0869, 0.4355, 0.8598, 1.0142, 1.0166}*; $H = 1.2321$. The quadratic approximation error value was $2.2973 \cdot 10^{-3}$.

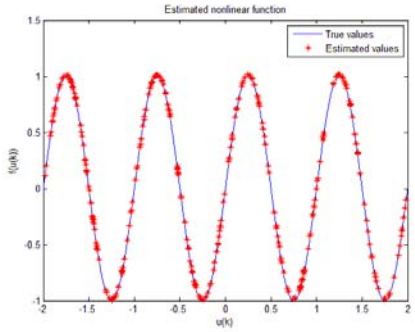

Fig. 10. Estimated nonlinear function (dead-zone)

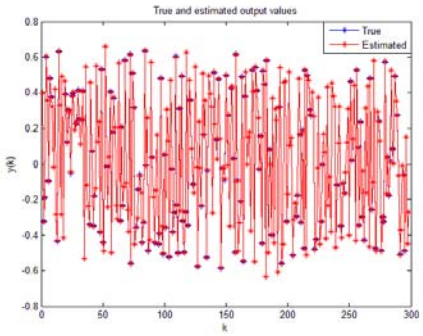

Fig. 11. True and estimated outputs

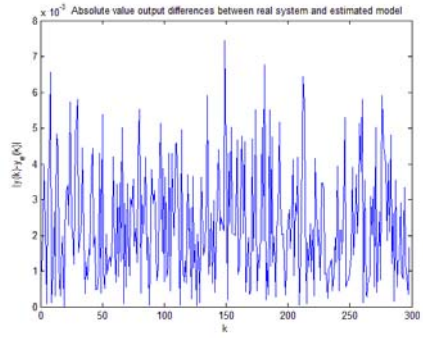

Fig. 12. Differences between true and estimated output values

The algorithm has also been tested on a linear analog system connected to a PC. The discrete model corresponding to the association of a zero-order hold CAN and the simulated analog system (sampling time was set to *0.1* seconds) is:

$$\frac{B(z^{-1})}{A(z^{-1})}=\frac{-0.0411\cdot z^{-1}-0.0914\cdot z^{-2}}{1-1.662\cdot z^{-1}+0.9571\cdot z^{-2}-0.2191\cdot z^{-3}} \tag{20}$$

The considered input signal is an uniform random signal with amplitude range of [*-2V, 2V*]. The static nonlinearity was generated as $f(u(k))=u(k)+0.3\cdot u^3(k)$. The results (also see Figures 13-15) are the following:

$$\begin{cases} A(q^{-1})=1-1.6445\cdot q^{-1}+0.9044\cdot q^{-2}-0.1826\cdot q^{-3} \\ B(q^{-1})=-0.0411-0.0891 q^{-1} \end{cases}$$

$\{\alpha_i\}$= {*-1.9192, -1.8942, -0.8418, -0.4675, -0.3279, -0.1203, 0.0581, 0.6942, 0.7568, 1.7065*}; *H = 8.3068*. The quadratic approximation error characteristic to the estimation model was $2.5029\cdot 10^{-5}$.

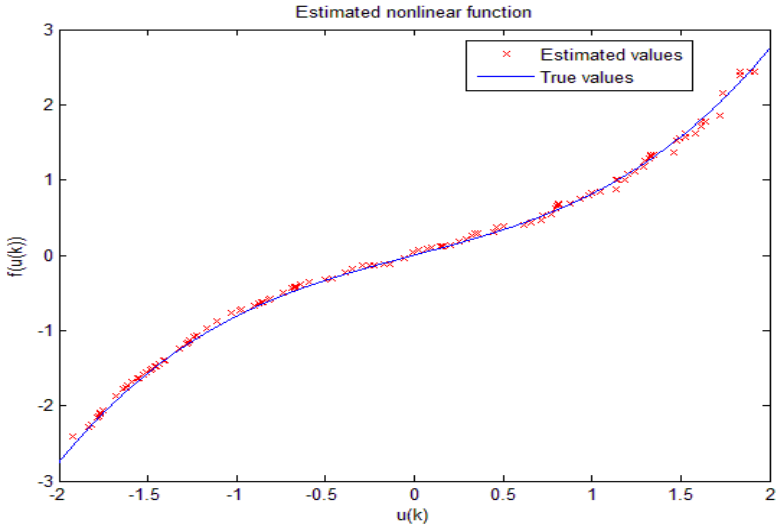

Fig. 13. Estimated nonlinear function (analog system case)

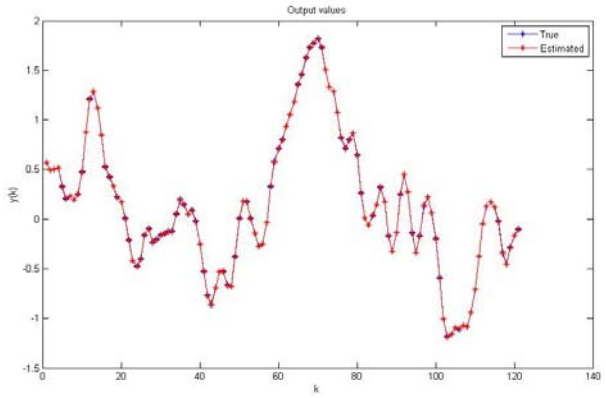

Fig. 14. True and estimated outputs

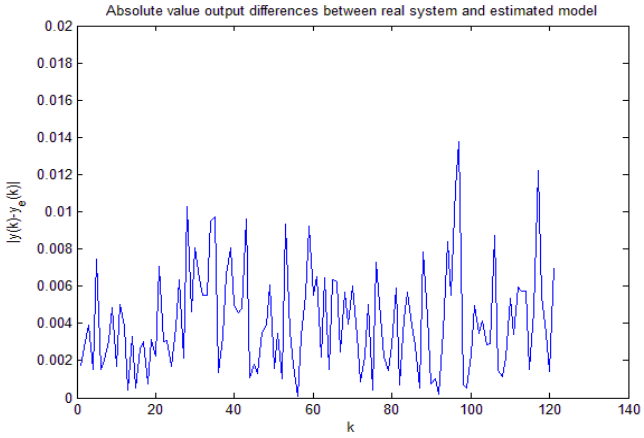

Fig. 15. Differences between true and estimated output values

6. Conclusions

This paper describes an identification method used for Hammerstein systems, by providing estimations for both static nonlinear and dynamic linear components. The main challenge in implementing this algorithm was solving the non-convex equation (12) and evaluating the unknown coefficients; however, optimizing the ACF coefficients by using PSO has greatly reduced its significance. Also, the PSO procedure was modified, in order to adapt this method to the specific restrictions generated by our identification problem. Still, for greater dimensional problems, PSO proves to have some disadvantages, because of the heavy increase of numeric effort.

Acknowledgement

The work has been funded by the Sectoral Operational Programme Human Resources Development 2007-2013 of the Romanian Ministry of Labour, Family and Social Protection through the Financial Agreement POSDRU/88/1.5/S/61178 and by FP7-REGPOT-2010-1 ERRIC project, No. 264207.

REFERENCES

[1] G. *Dauphin-Tanguy, L. Foulloy, D. Popescu*, Modélisation, identification et commande des systèmes, Romanian Academy Press, Bucharest, Romania, 2004.

[2] *R. Haber, L. Keviczky*, Nonlinear system identification - Input-Output modeling approach, Kluwer Academic Publishers, Dorchrecht/ Boston/ London, 1999.

[3] *P. Stoica, T. Söderström*, Instrumental variable methods for identification of Hammerstein systems, International Journal of Control, **vol. 35**, pp. 459-476, 1982.

[4] *J. Vörös*, Parameter identification of Hammerstein systems with asymmetric dead-zones, Journal of Electrical Engineering, **Vol. 55**, No. 1-2, 2004, pp. 46-49.

[5] *S. A. Billings, S. Y. Fakhouri*, Identification of non-linear systems using correlation analysis and pseudorandom inputs, International Journal of Systems Science, **Vol. 11**, No. 3, 1980, pp. 261–279.

[6] *S. A. Billings, S. Y. Fakhouri*, Identification of systems containing linear dynamic and static nonlinear elements, Automatica, **Vol. 18**, No. 1, 1982, pp. 15–26.

[7] *X. Chen, H.T. Fang*, Recursive Identification for Hammerstein Systems with State-space Model, Acta Automatica Sinica, **Vol. 36**, No. 10, 2010, pp. 1460-1467.

[8] *M. Ahmadi, H. Mojallali*, Identification of multiple-input single-output Hammerstein models using Bezier curves and Bernstein polynomials, Applied Mathematical Modeling, **Vol. 35**, No. 4, 2011, pp. 1969-1982.

[9] *H. Seki,* Nonlinear Function Approximation Using Fuzzy Functional SIRMs Inference Model, Proceeding of 11th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases, 2012, pp.201--206.

[10] *C. Novara*, Sparse Identification of Nonlinear Functions and Parametric Set Membership Optimality Analysis, Proceedings of the American Control Conference, 2011, pp. 663-668.

[11] *H. Takata*, An automatic choosing control for nonlinear systems, Proceedings of the 35th IEE CDC, 1996, pp. 3453-3458.

[12] *J. Kennedy, R. C. Eberhart*, Particle Swarm Optimization, Proceedings of IEEE International Conference on Neural Networks, 1995, pp. 1942-1948.

[13] *T. Hachino, K. Shimoda, H. Takata*, Identification of discrete-time Hammerstein systems by using Automatic Choosing Function model and Particle Swarm Optimization, Proceedings of the International Conference on Electrical Engineering, 2008, pp.O-095-1-O-095-6.