

Using Geno-Iterative Approach to Identify Weiner and Hammerstein Models

Ali H. Hasan

Thi-Qar University

Abstract

This paper presents a novel method for identification of the Weiner and Hammerstein models and the model's parameters by the application of genetic algorithm optimization method and an iterative search through a lock up Table. The coefficient values of both linear and nonlinear parts are estimated by the GA while the type of nonlinearity and degree of delay of the linear part are determined by the iterative search through the lock up Table. The simulation results show the effectiveness and ability of the proposed algorithm for identification and realization of the Weiner and Hammerstein models that describe the real system.

Keywords: Hammerstein model identification, Weiner model identification, linear and nonlinear system identification , genetic algorithms, iterative method.

المستخلص

قمنا في هذا البحث بعمل طريقة جديدة لإيجاد نماذج Weiner و نماذج Hammerstein المستخدمان في تمثيل الأنظمة الحقيقية المحتوية على المكونات الخطية وغير الخطية وذلك باستخدام مبدأ الخوارزميات الجينية وطريقة البحث التكرارية. استخدمنا طريقة البحث التكرارية لاختيار النماذج الخطية وغير الخطية المكونة لنماذج Weiner و نماذج Hammerstein المستعملة في تمثيل النظام الحقيقي. بينما استخدمنا الخوارزميات الجينية لإيجاد القيم الحقيقية لمعاملات (coefficient) النماذج المستخدمة. بينت النتائج المستحصلة قدرة الطريقة الجديدة لاختيار النماذج وتمثيل النظم الحقيقية من ناحية إيجاد الصيغة الرياضية التي تمثل تصرف هذه النظم وكذلك إيجاد المعاملات الخطية المكونة لهذه الصيغ.

1. Introduction

System identification is the process of finding a model that best produces the data obtained by a system to known inputs, i.e. construct mathematical models from measured input and output data [1]. System identification differs from modeling in that, with modeling the observed system is broken down into subsystems, which can be described mathematically. Identification builds a picture of the system using just the observed data, that is, the input-output signals.

Practical systems have inherently nonlinear characteristics such as saturation and dead zone which grown the importance of nonlinear modeling in control engineering. So the development of accurate non linear system identification is important aspect for analysis and prediction of the system characteristics or control design problem.

Block oriented models are one of the major classes of nonlinear systems, which consist of linear models followed by, or preceded by a static nonlinear model. This configuration is known as Hammerstein and Wiener models depends on the blocks order, this configuration provides a simple system architecture. More details can be found in [2-6].

The Hammerstein and Wiener models are used to model several classes of nonlinear systems, their flexibility lies in having the nonlinearity part entirely separated from the linear part [7]. Figure 1 represent the block oriented structure:-

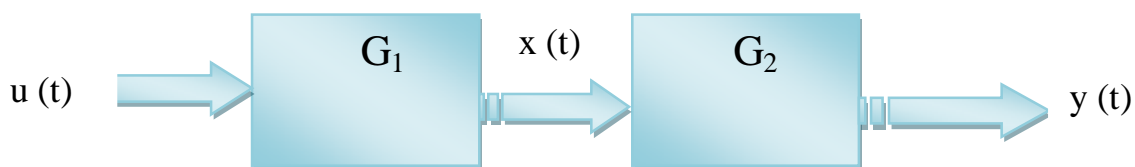


Figure (1). Structure of block oriented representation.

Where:-

$u(t)$ is the input signal to the system, $y(t)$ is the output signal from the system and $x(t)$ is the intermediate non measurable quantity.

G_1 is the static nonlinearity part for Hammerstein model and is the dynamic linear part for Wiener model.

G_2 is the dynamic linear part for Hammerstein model and is the static nonlinearity part for Wiener model.

The Hammerstein model can be described by the following equations:-

$$y(t) = \sum_{i=1}^m (a_i y(t-i)) + \sum_{j=0}^n (b_j x(t-j)) \quad (1)$$

$$x(t) = f(u(t)) \quad (2)$$

While Wiener model can be described by the following equations:-

$$y(t) = f(x(t)) \quad (3)$$

$$x(t) = \sum_{i=1}^m (a_i x(t-i)) + \sum_{j=0}^n (b_j u(t-j)) \quad (4)$$

Where:-

m is the number of poles of the linear part.

n is the number of zeros of the linear part.

a_i and b_i are linear part coefficients.

Genetic algorithms (GAs) are a family of general stochastic search methods, which can be viewed as computational models of Darwinian evolution theory. They use the analogs of evolutionary operators on a population of states in a search space to find those states that optimize a fitness function. The search space consists of character strings of fixed or variable length (chromosomes or genotypes) composed of the elements of a given alphabet (alleles).

The genotype space is mapped onto another (phenotype) search space. The fitness function is defined as a function of a state in the phenotype space [8-10].

An implementation of a genetic algorithm begins with a population of typically random chromosomes. One then evaluates these structures and allocates reproductive opportunities in such a way that those chromosomes which represent a better solution to the target problem are given more chances to reproduce than those chromosomes which are poorer solutions. The solution goodness is typically defined with respect to the current population [11]

There are many identification techniques that have been used to identify the linear and nonlinear parts of the Hammerstein model and the Wiener model each alone, examples of these identification techniques are Marconato et al [4] who used the Support Vector Machines to identify the nonlinear systems of Wiener-Hammerstein systems based on input/output measurements. Sou et al [12] used a convex semi definite programming SDP relaxation to obtain a sub-optimal solution to non-convex quadratic programming that formulates the Wiener-Hammerstein models. Marconato et al [13] presented an identification of the Wiener-Hammerstein models by a learning from example approach SVM for regression on the basis of real life benchmark data and they used the genetic algorithm to select the best model that describe the input/output relationship. Wang et al [14] presented an identification of Wiener model by a recurrent neural network with the observer Kalman filter (OKID) algorithm for unknown dynamic nonlinear system identification. There are many papers used genetic algorithm principles for identification purposes, for example Lu et al [15] used a GP models to obtain the nonlinear function and parameters of Wiener model. Dotoli et al [16] described the GA to identify nonlinear SISO Hammerstein model, they assume a bounded but not a prior known order for linear part and the nonlinear part belongs to a known structure. Xie et al [17] used an iterative method of GA to get the Wiener model but they assume a priori knowledge of the objective system and the expected object linear system.

Other uses of GA for identification of Hammerstein model can be found at Al-Duawaish et al [18], Hachino et al [19,20], while using of GA for identification of Wiener model can be found in Vázquez et al [21].

In this work, we used iterative search method combined with directed search by GA to find the Wiener and Hammerstein models parameters and identify number of delays for both

input and output also determine a good approximation for the nonlinearity type of the system under consideration.

2. Proposed algorithm

As shown in Figure (1) the Wiener-Hammerstein model consists of two sub-models. It's well known that the genetic algorithm can easily identify the sub-models, especially if the inputs and outputs of the corresponding sub-systems are known, many papers assumed that the middle stages are known [17] so the identification demands are simplified. In this work, the problem assumes there is no prior knowledge of the middle stages, only the input signals and output responses of the systems under consideration are known.

The basic idea is to walk through a lockup Table that contains a list of number of delays for the input and output. It also contains some nonlinear relationships. After selection of a specific relationship, we run the GA to search for the parameters values.

The complete algorithm can be described in the following steps which consists of two parts that work simultaneously, walking through the lockup Table and running the genetic algorithm:-

a- Part 1 : The lockup Table

The lockup Table contains two columns, the first one describes the behavior of the output by number of delays and type of nonlinearity, the second one describes the behavior of the input by number of delays and type of nonlinearity. Its well know that the linear subsystem is included within the output delays while the nonlinear subsystem is included within the input signal [2,22]. In our experiments we included some nonlinearity within the output signal which describes the unexpected behaviors of the original system like the disturbance and noise.

The selection process is as follows:-

Step 1:- Assume we select the first row; then the plant equation will be:-

$$y(t) = ay(t - 1) + bu(t - 1) \quad (5)$$

Step 2:- Start the Genetic Algorithm Phase (part two) to find the coefficients values of Step 1, if the result is not satisfied then go to Step 3

Step 3:- Select the second row for the 1st column and 1st row for the 2nd column, then the plant equation will be:-

$$y(t) = a_1y(t - 1) + a_2y(t - 2) + bu(t - 1) \quad (6)$$

Step 4:- Start the Genetic Algorithm Phase (part two) to find the coefficients values of Step 1, if the result is not satisfied then go to Step 5

Step 5:- Select the second row for the 1st column and 2nd row for the 2nd column, the plant equation will be:-

$$y(t) = a_1y(t - 1) + a_2y(t - 2) + b_1u(t - 1) + b_2u(t - 2) \quad (7)$$

Step 6:- Repeat these selection criteria until the plant estimation will satisfy the requirements or the lockup Table is finished.

Where

a_i and b_i are linear part coefficients

Figure (2) describes the complete programming flowchart of the proposed work.

Table (1). Lock up Table . Number of delays and nonlinear order and type of both inputs and outputs.

Item No.	Number of delays of o/ps	Number of delays of i/ps or nonlinearity type
1	y^{-1}	u^{-1}
2	y^{-2}	u^{-2}
3	y^{-3}	u^{-3}
4	y^{-4}	u^{-4}
5	$\frac{1}{1 + \exp(y)}$	u^2
6	$\frac{1}{1 + y}$	u^3
7		$\frac{1}{1 + \exp(u)}$
8		$\frac{\exp(-u) - \exp(-u)}{\exp(-u) + \exp(-u)}$
9		Sin(u)
10		Cos(u)

b- Part 2 : Genetic algorithm

- a- Set the population size = Number of coefficients of the selected equation that describe the system.
- b- Create the initial population values, which are the coefficients of initial subsystem. Here we used the matlab function ' crtrp' that creates real-valued initial population.
- c- Evaluate the population fitness by using the mean square error between the actual output $y(n)$ and the estimated output $\hat{y}(n)$

$$E = \frac{1}{N} \sum_{n=1}^N (y(n) - \hat{y}(n))^2 \quad (8)$$

Where: - N number of samples

- d- If E is equal or less than the satisfying error then the description of the system is fixed.

else

- i. Do the selection process to select the winning parents, which is the roulette wheel.
- ii. Do the crossover process to get the offspring , we used the single point crossover method.
- iii. Go to step c until number of iterations is reached or step d is true.

For an efficient programming, we use the matlab functions that deals with genetic algorithms which are :-

- 1- 'gaoptimset' that creates genetic algorithm options structure, and
- 2- 'ga' that implements the genetic algorithm at the command line to minimize an objective function.

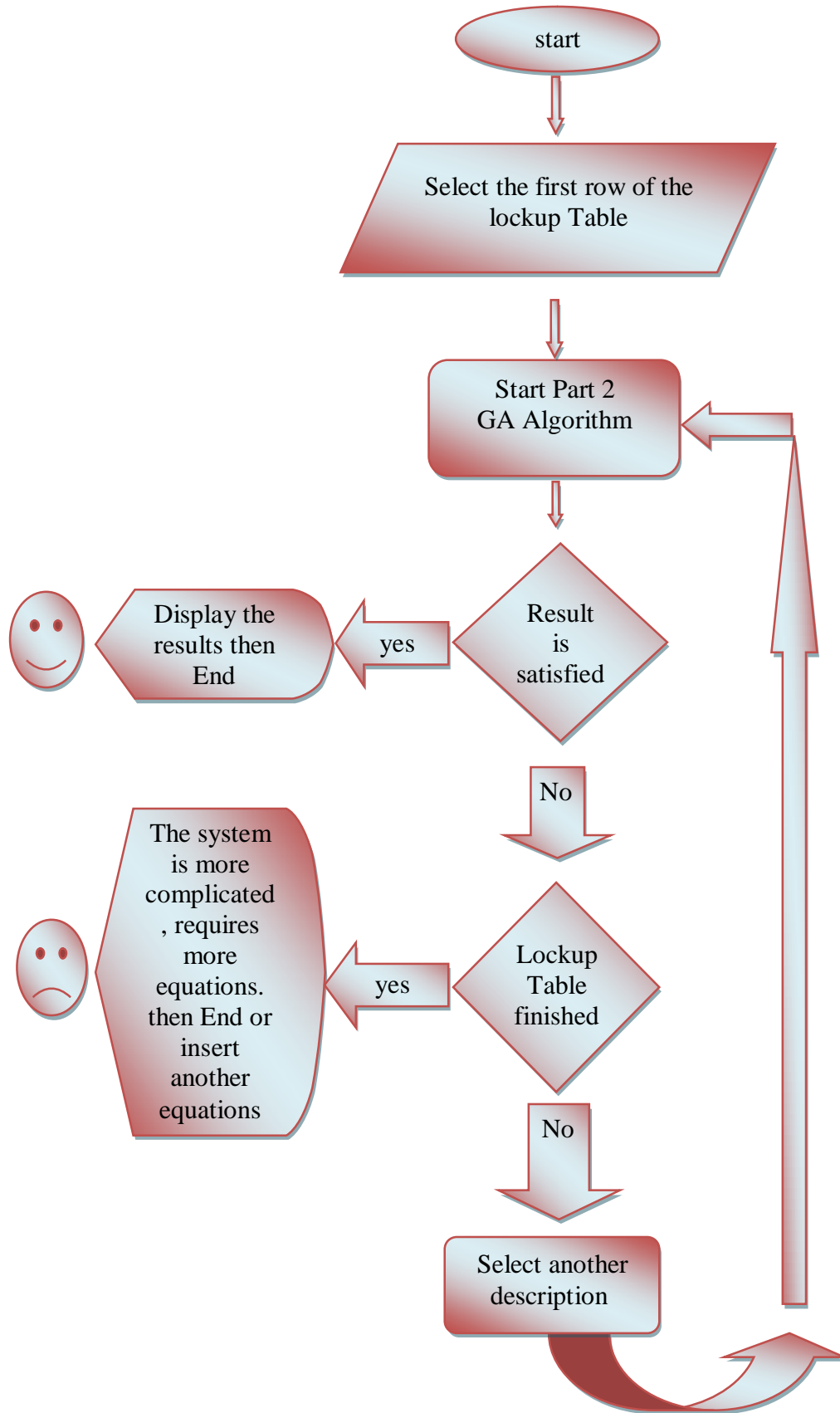


Figure (2). Proposed work flowchart.

3. Experimental results

To show the validity of the proposed algorithm many system are considered with different types:-

1- Plant 1:- Simple linear system

$$y(t) = a_1y(t - 1) + a_2y(t - 2) + bu(t - 1) \quad (9)$$

The actual values of coefficients are:-

$$a_1=0.4;$$

$$a_2=0.35; \text{ and}$$

$$b=0.8$$

After running the proposed algorithm, we used 1500 generations with 100 population sizes for the GA we found the following system:-

$$y(t) = a_1y(t - 1) + a_2y(t - 2) + bu(t - 1) \quad (10)$$

$$a_1= 0.4050;$$

$$a_2=0.3485; \text{ and}$$

$$b_1=0.801$$

Which gave a good description for both the two sub systems as shown in Figures (3), (4), and (5).

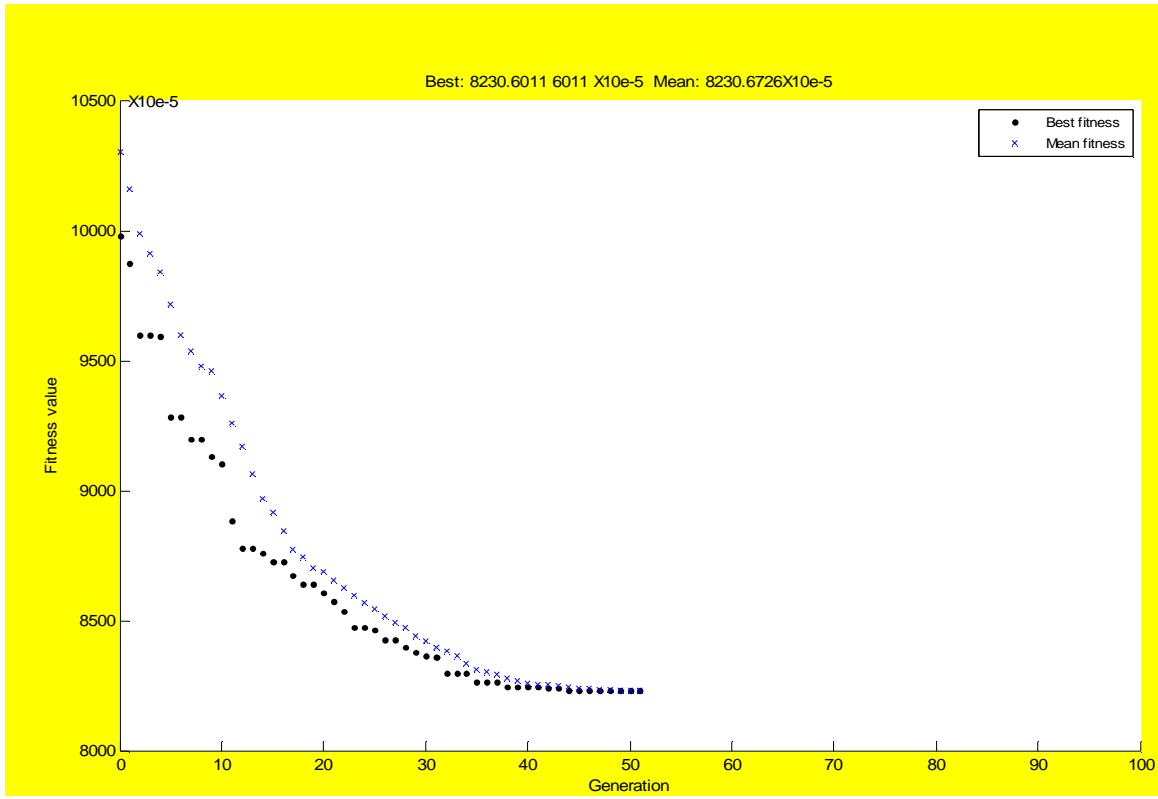


Figure (3). Genetic algorithm fitness values.

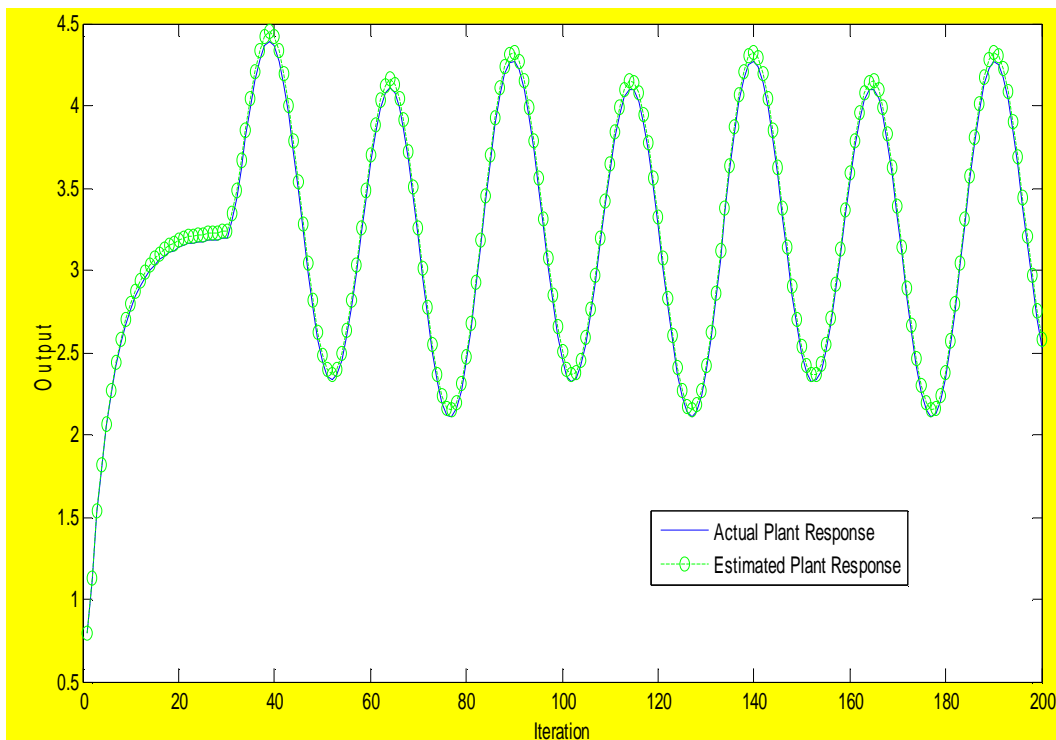


Figure (4). Actual and estimated plant output response.

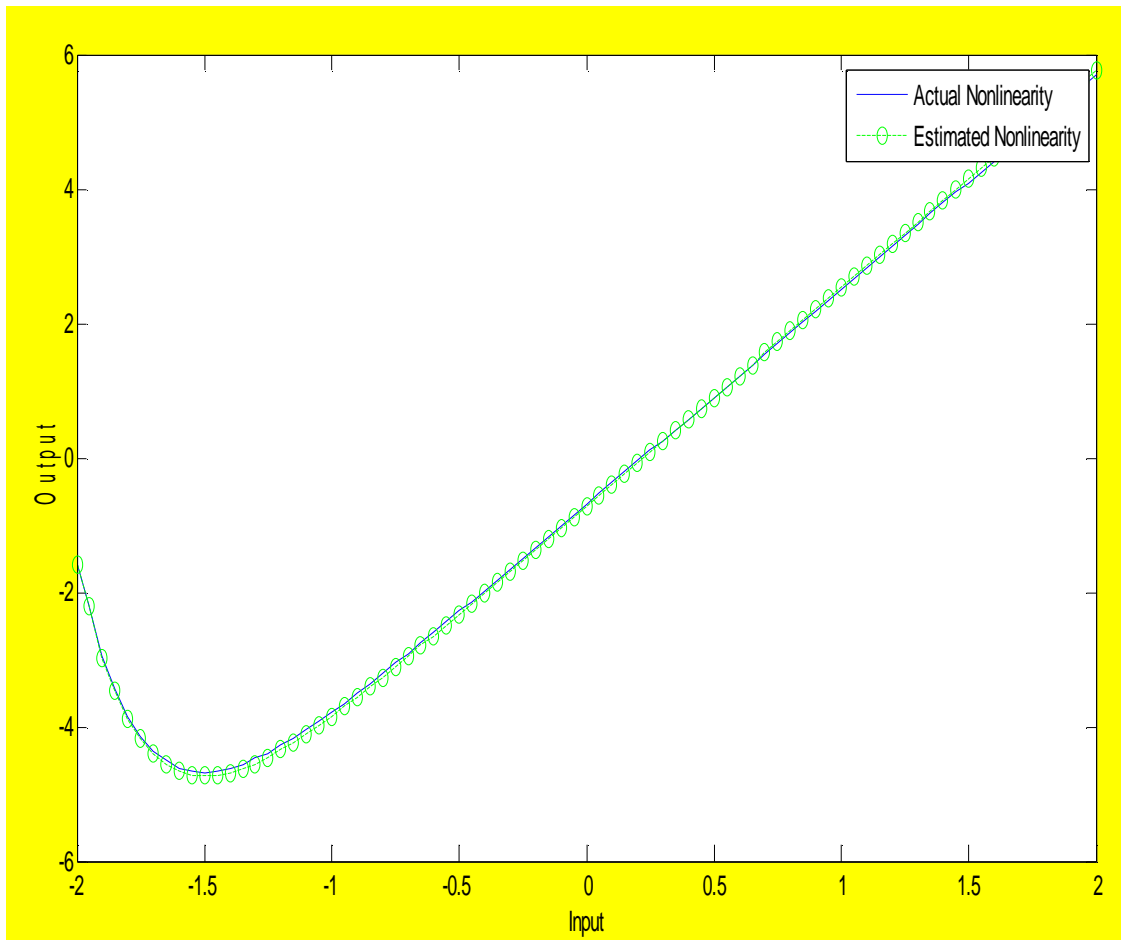


Figure (5). Nonlinearity behavior of the estimated system.

2- Plant 2:- which is 3rd order nonlinear system :-

$$y(t) = \frac{a y(t-1)}{1+y(t-1) * y(t-1)} + b x(t) \quad (11)$$

and $x(t) = (u(t-1))^3 \quad (12)$

The real values are $a=1$ and $b=1$;

After running the proposed algorithm we found the following plant equation:-

$$y(t) = \frac{a y(t-1)}{1+y(t-1) * y(t-1)} + b x(t) \quad (13)$$

$$\text{and } x(t) = (u(t-1))^3 \quad (14)$$

And estimated values are:-

$$a=1.0026$$

$$b=0.9954$$

Figures (6), (7), and (8) show the estimated and real plant equations behavior:-

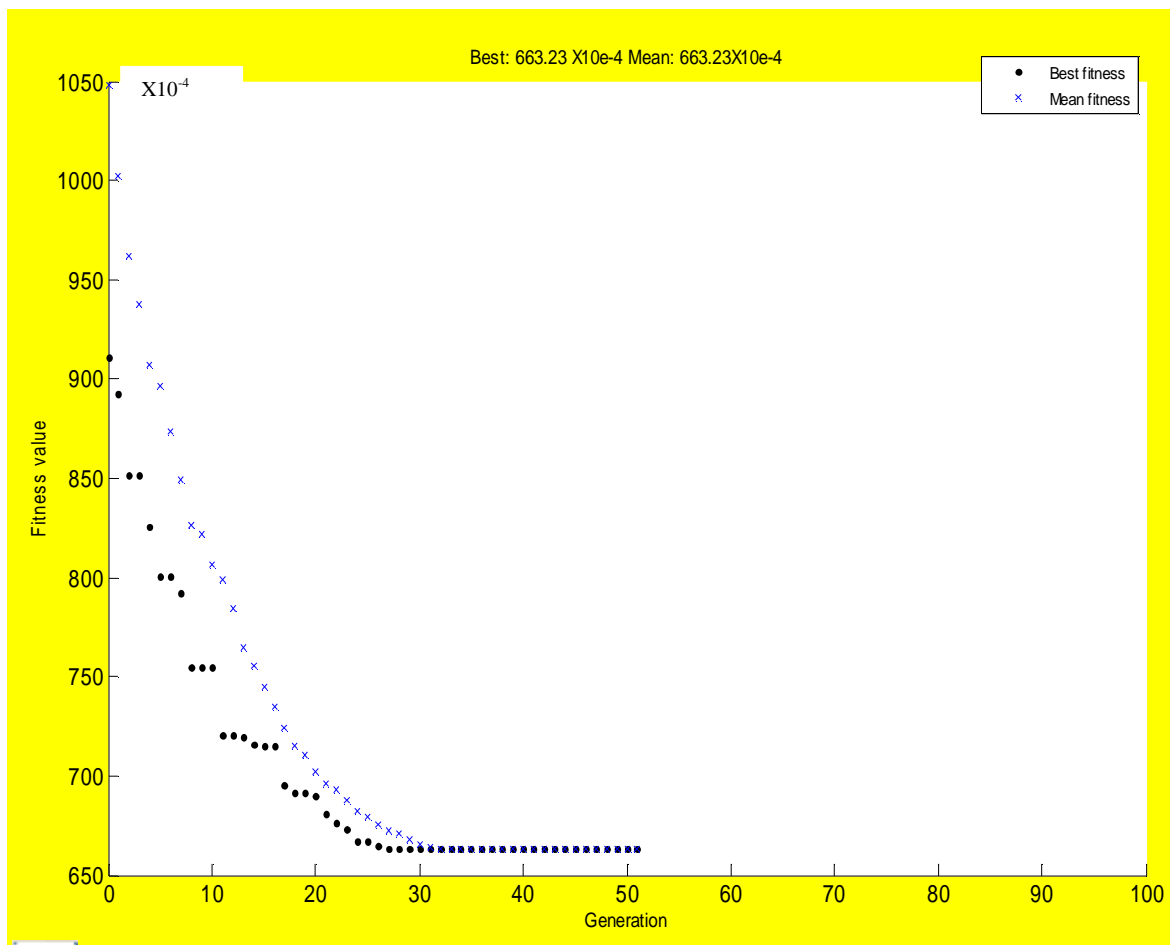


Figure (6). Genetic algorithm fitness values.

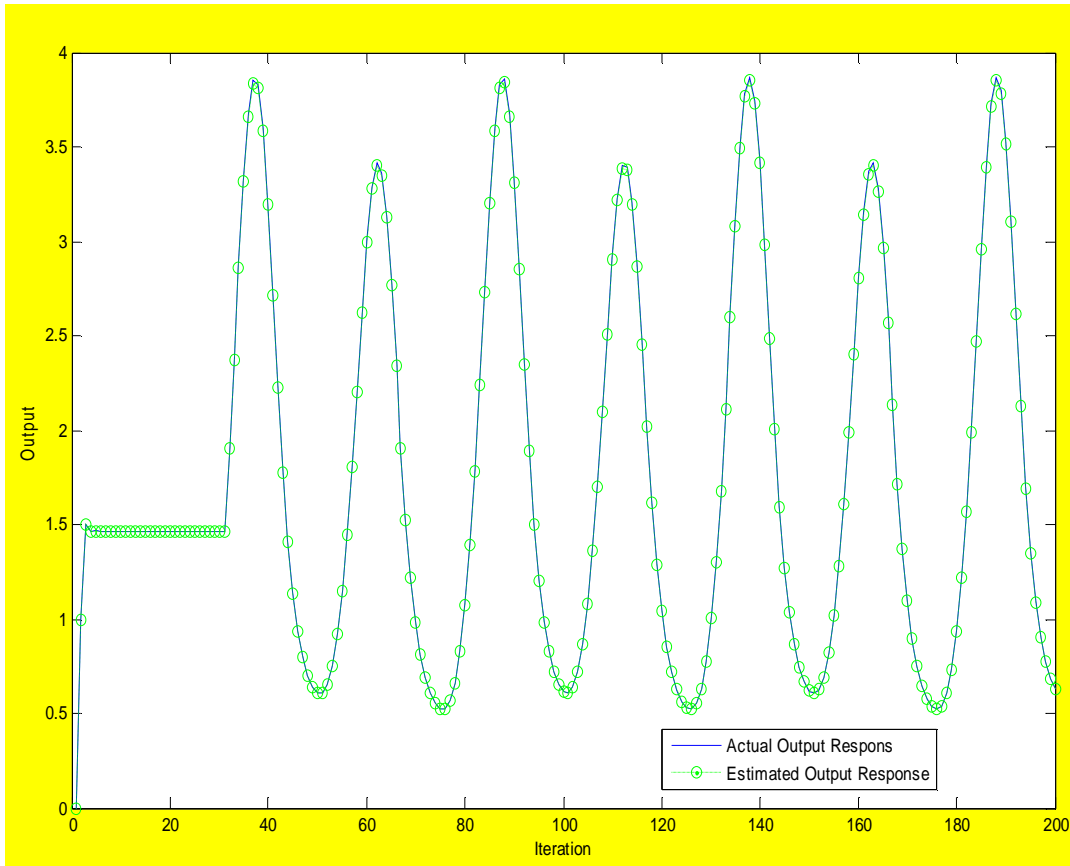


Figure (7). Actual and estimated plant output response.

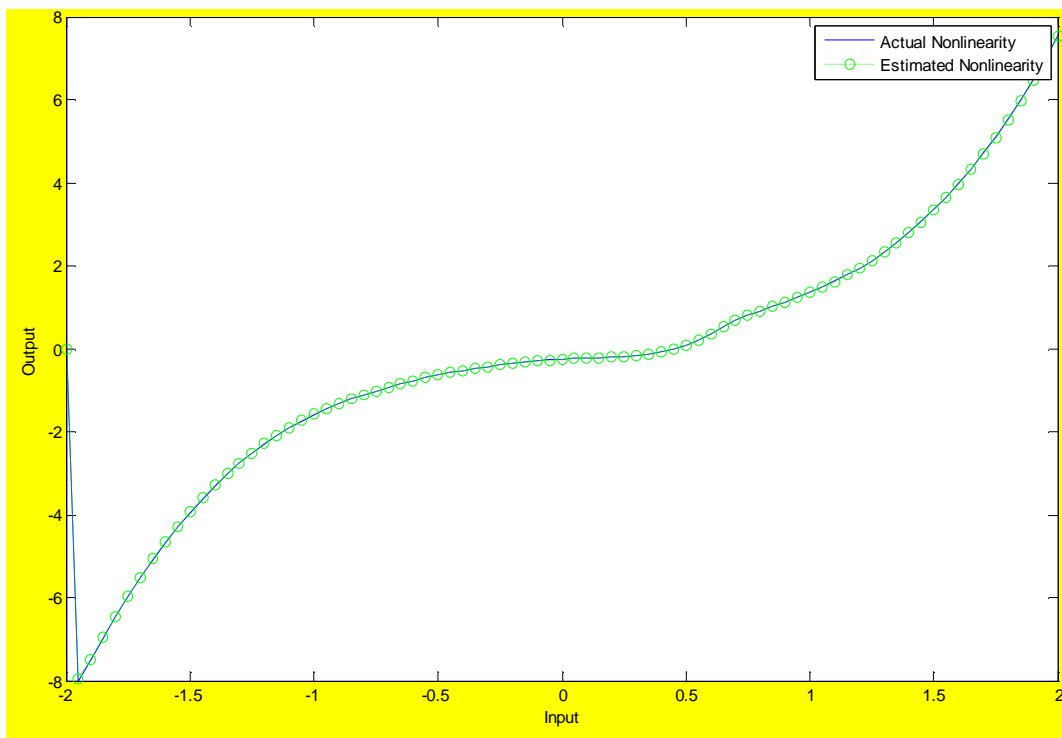


Figure (8). Nonlinearity behavior of the estimated system.

3- Plant 3:-

$$y(t) = a_1y(t-1) + a_2y(t-2) + a_3y(t-3) + b_1x(t-1) + b_2x(t-2) \quad (15)$$

$$x(t) = \cos(3*u(t)) + \exp(-u(t)) \quad (16)$$

The real values are

$$a_1=0.4;$$

$$a_2=0.35;$$

$$a_3=0.1;$$

$$b_1=0.8;$$

$$b_2=-0.2.$$

The estimated ones are:-

$$y(t) = a_1y(t-1) + a_2y(t-2) + a_3y(t-3) + b_1x(t-1) + b_2x(t-2) \quad (17)$$

$$x(t) = \cos(2.8*u(t)) + \exp(-u(t)) \quad (18)$$

$$a_1=0.387$$

$$a_2=0.361$$

$$a_3=0.08976$$

$$b_1=0.07856$$

$$b_2=-0.17878.$$

The validity is shown in figures (9), (10), and (11).

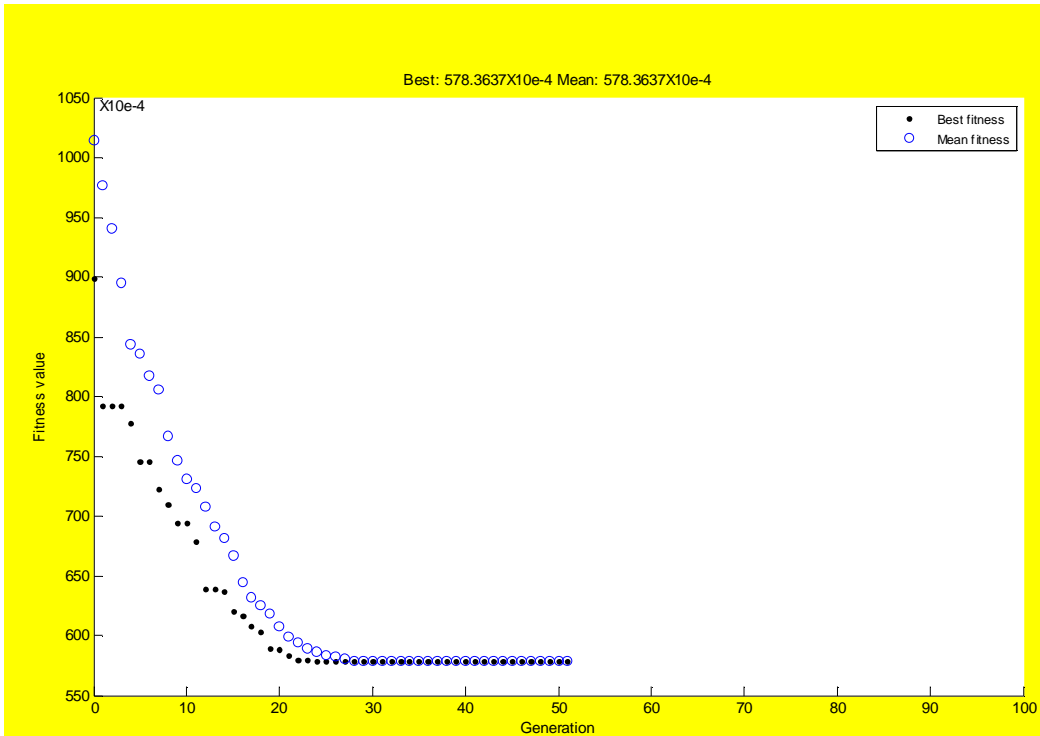


Figure (9). Genetic algorithm fitness values.

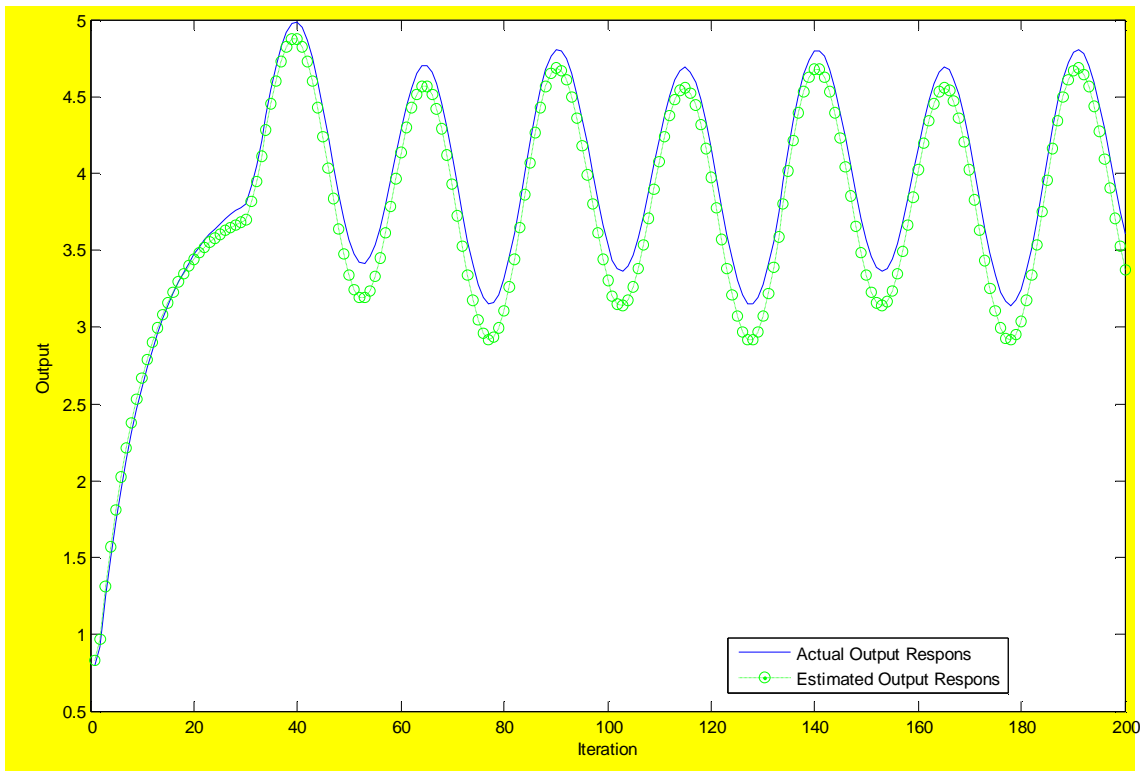


Figure (10). Actual and estimated plant output response.

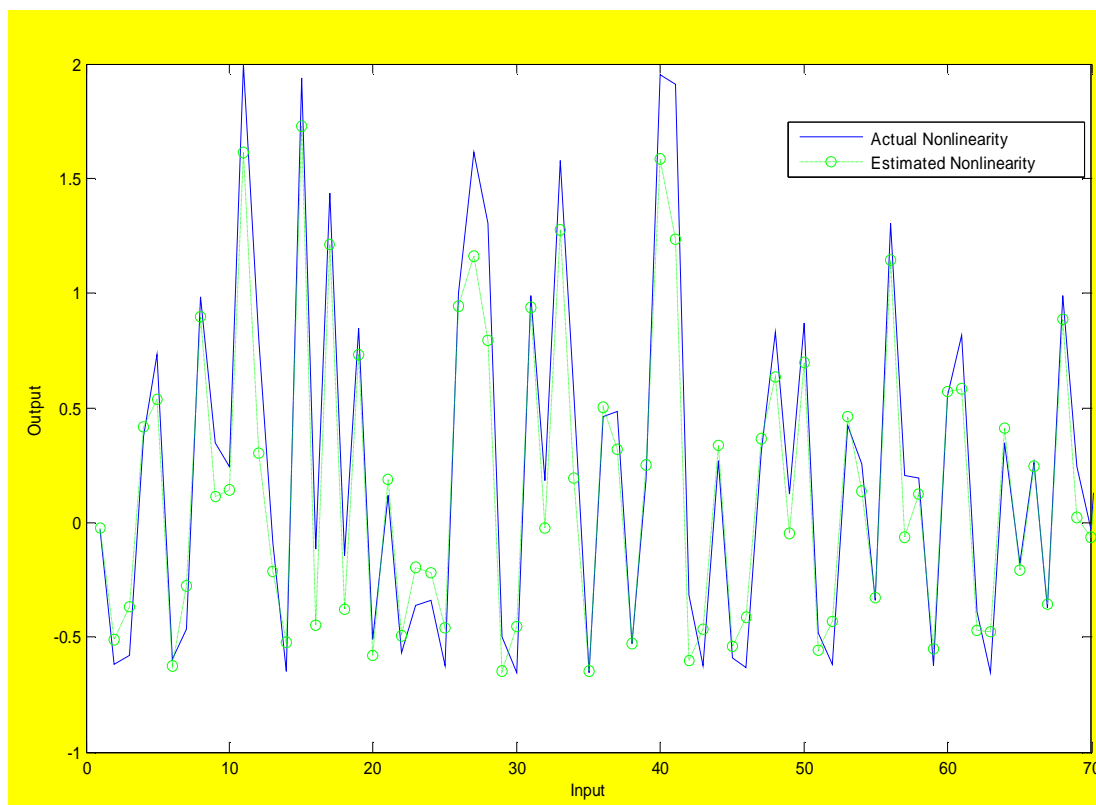


Figure (11). Nonlinearity behavior of the estimated system.

4- Conclusions

This work proposed a new method for identification of both Hammerstein and Wiener models. From the given examples of different nonlinearities systems which describe the real life plants, the results show that the proposed work is immune to noise and can be applied to real-time plants identification.

One can increase the reliability of this work by introducing more functions that describe real systems as well as using state space approach.

5- References

- [1] Ioan D. Landau and Gianluca Zito, " Digital Control Systems Design, Identification and Implementation", Springer-Verlag London Limited 2006.
- [2] Sayed Saad Azhar Ali, "Identification of Hammerstein and Wiener Models using radial basis functions neural networks", Msc Thesis in Electrical Eng. , King Fahad University of Petroleum and Minerals, Dhahran, Saudi Arabia, Dec. 2001.
- [3] Al-Duwaish H. and Naeem, Wasif, "Nonlinear Model Predictive Control of Hammerstein and Wiener Models Using Genetic Algorithms", I E E E, Proceedings of the 2001 IEEE

- International Conference on Control Applications,
http://isi.kfupm.edu.sa/proceedings/pdf/N/nonlinear_model_predictive_control_of_ha_al_duwaish_isip_000173316300082.pdf
- [4] Anna Marconato¹, Andrea Boni¹, Dario Petri¹, and Johan Schoukens²; "Nonlinear system identification by means of SVMs: choice of excitation signals", www.imeko.org/publications/tc4-2008/IMEKO-TC4-2008-152.pdf.
- [5] Anna Marconato, Johan Schoukens, "Identification of Wiener-Hammerstein Benchmark Data by Means of Support Vector Machines", 15th IFAC Symposium on System Identification Saint-Malo, France, July 6-8, 2009, <http://www.tw.vub.ac.be/elec/Papers%20on%20web/Papers/AnnaMarconato/IFAC-SYSID09-Marconato-Identification.pdf>.
- [6] R. Bhushan Gopaluni, Thomas B. Schön, and Adrian G. Wills, "Particle Filter Approach to Nonlinear System Identification under Missing Observations with a Real Application", www.control.isy.liu.se/~schon/Publications/GopaluniSW2009.pdf
- [7] Kin Cheong Sou, Alexandre Megretski and Luca Daniel, "Convex Relaxation Approach to the Identification of the Wiener-Hammerstein Model", institutes.lanl.gov/isti/isami/papers/cdc2008_WH.pdf
- [8] John R. Koza, "Genetic Programming On the Programming of Computers by Means of Natural Selection", A Bradford Book The MIT Press London, England, Sixth printing, 1998.
- [9] John S. Gero and Vladimir Kazakov, "A Genetic Engineering Volume 9, Number 1, 2001 by the Massachusetts Institute of Technology.
- [10] Randy L. Haupt and Sue Ellen Haupt, "PRACTICAL GENETIC ALGORITHMS", A JOHN WILEY & SONS, INC., PUBLICATION, second edition 2004.
- [11] Darrell Whitley, "A Genetic Algorithm Tutorial", www.stat.ucla.edu/~yuille/courses/Stat202C/ga_tutorial.pdf
- [12] Kin Cheong Sou, Alexandre Megretski and Luca Daniel, "Convex Relaxation Approach to Identification of the Wiener-Hammerstien Model", In proceedings of the 47th IEEE Conference on Decision and Control, December 2008, ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4739417
- [13] Anna Marconato and John Schoukens, "Identification of Wiener-Hammerstein Benchmark Data by Means of Support Vector Machines", 15th IFAC symposium on system identification Saint-Malo, France, July 6-8, 2009.

- [14]Jeen-Shing Wang and Yu-Liang Hsu, "Dynamic nonlinear system identification using a Wiener-type recurrent neural network with OKID algorithm", Journal of Information Science and Engineering, 24, 891-905, 2008.
- [15]Yueh-Chun Lu, Ming-Hung Chang and Te-Jen Su," Wiener Model Identification using Genetic Programming", Proceeding of the International MultiConference of Engineers and Computer Scientists 2008 Vol II, IMECS 2008, 19-21 March, 2008, Hong Kong.
- [16]Mariagrazia Dotoli, Guido Maione, Davide Naso and Biagio Turchiano, " Genetic Identification of Dynamical System with Static Nonlinearities",
- [17]Changsong Xie and Xuhui Li, "An algorithm using genetic programming for the compensation of linear distortion based on Wiener system model", Facta Universitatis (NIŠ), Elect. Eng. Vol 17, Aug 2004, 219-229.
- [18]Tomohiro Hachino, Kenji Shimoda and Hitoshi Takata, "Hybrid algorithm for Hammerstein system identification using genetic algorithm and particle swarm optimization", World academy of Science, Engineering and Technology 53, 2009.
- [19]Al-Duwaish H and Naeem Wasif, "Nonlinear model predictive control of Hammerstein and Weiner models using genetic algorithms", Proceedings of the 2001 IEEE International Conference on Control Applications, <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.2.3177&rep=rep1&type=pdf>.
- [20]Tomohiro Hachino, Katsuhisa Deguchi and Hitoshi Takata, " Identification of Hammerstein model using Radial basis function networks and genetic algorithm", Proceedings of the 5th Asian Control Conference 2004, asc2004.ee.mu.oz.au/proceedings/papers/P18.pdf
- [21]K. Rodriguez- Vázquez and P.J. Fleming, "Multi-objective genetic programming for nonlinear system identification", Electronics Letters, 30th April, 1998, Vol 34, No. 9.
- [22]Sayed Zeeshan Rizvi, " A new method for identification of MIMO Hammerstein model", Msc Thesis in Electrical Eng., King Fahad University of Petroleum and Minerals, Dhahran, Saudi Arabia, June 2008.