Parameter identification of nonlinear systems using a particle swarm optimization approach

Wei-Der Chang* Jun-Ping Cheng Ming-Chieh Hsu Liang-Chan Tsai
Department of Computer and Communication
Shu-Te University
Kaohsiung 824, Taiwan
e-mail: wdchang@stu.edu.tw

Abstract—**This paper applies a particle swarm optimization (PSO) approach to the parameter identification for a class of nonlinear systems. In the PSO optimization process, the unknown system parameters are arranged in the form of a parameter vector (i.e. a particle), and the PSO algorithm employs the velocity updating and position updating formulas to an initial population, which is constituted by a great number of particles, such that the excellent particle is generated. The proposed algorithm manipulates the parameter vectors directly as real numbers rather than binary strings. Therefore, to implement the PSO algorithm in computer codes becomes fairly straightforward. In this study, the PSO algorithm is applied to estimate the parameters of the Genesio-Tesi nonlinear chaotic systems. The estimation performance of the PSO algorithm is verified by examining different sets of random initial populations under the presence of measurement noises. The simulation results reveal that the PSO algorithm provides a simple and effective means of solving parameter estimation problem of nonlinear systems.**

Keywords-particle swarm optimization (PSO); nonlinear systems; parameter identification; Genesio-Tesi Chaos

I. INTRODUCTION

The work of system identification is significantly important and essential in many engineering applications, especially for control system engineering. If the mathematical model of a physical system can be completely obtained, then several control design techniques according to the mathematical model are presented to meet some control specifications. To solve the problem of system identification, a great number of new methods are presented such as neural network [1] and fuzzy systems [2] in last few years. The overall input-output dynamics of an unknown physical system can be estimated or forecasted by means of updating the adjustable parameters within the neural networks and fuzzy systems. The obtained neural and fuzzy models for such the system are then utilized in various control applications. It does not need to consider the actual system structures. On the other hand, when the system structures are assumed to be known, the system identification problem is changed to how to solve the correct system parameters. For solving the problem of parameter estimation, the least-squares (LS) method is usually utilized. This method is quite simple and can be analytically calculated when the structure of the estimated system has the property of being linear in the parameters [3].

Particle swarm optimization (PSO) approach was proposed by Kennedy and Eberhart in 1995 [4]. This algorithm is one of new and popular evolutionary computations and has been applied in various engineering fields in recent years [5]-[11]. It is inspired by the behavior of organisms, such as fish schooling and bird flocking. Unlike other heuristic techniques, PSO has a flexible and well-balanced mechanism to enhance the global and local exploration abilities [5]. The algorithm is initialized with a population of random solutions. Each individual, also called particle, is assigned with a randomized velocity according to its own and its companions' flying experiences, and the individuals are then flown through hyperspace. Due to the good features of PSO algorithm, nowadays it has been emerged as a new and attractive optimization tool and successfully applied in variety of different fields. For example, the authors proposed a new design method for digital differentiator by the PSO algorithm [7]. The finite impulse response (FIR) filter with linear phase structure is designed to match the prescribed differentiation response. In [9], a design method for determining the optimal fuzzy PID-controller was developed based on using PSO algorithm. The controlled plant is the active automobile suspension system and simulation results show the proposed method can obtain a better performance. To solve simultaneous localization and mapping (SLAM) problems for mobile robots, the authors presented a modified PSO algorithm called Geese PSO algorithm to tune a fuzzy supervisor for an adaptive extended Kalman filter [11].

On the other hand, for solving parameter estimation problem of nonlinear chaotic systems, several different kinds of methods have been proposed, as shown in [12]-[15], and the principal technique frequently used is the synchronization. The basic principle of adaptive synchronization is that all of system states of two same or different chaotic systems are asymptotically regulated to the same by means of control strategies. The control strategy maybe includes the adaptive control, active control, and feedback control. To guarantee the synchronization stability of the closed-loop chaotic systems, the Lyapunov theorem is usually utilized. During synchronization, an adequate adaptation mechanism is proposed to update the unknown parameters of the slave chaotic system simultaneously. These unknown parameters finally approach to the actual values, i.e., the parameters of the master chaotic system.

In this paper, a simple but effective PSO algorithm is applied to estimate the system parameters for a class of nonlinear systems. The estimation results obtained from various cases, including different initial population tests and the presence of different measurement noises, are listed and compared. The rest of this paper is organized in

978-0-7695-4893-7/12 $26.00 © 2012 IEEE
DOI 10.1109/ICNC.2012.24

the following. The problem formulation for a class of nonlinear systems is briefly addressed in Section 2. In Section 3, an overview of PSO algorithm is introduced and its mathematical iterative formula is also described in detailed. In Section 4, an illustrative example of the Genesio-Tesi nonlinear chaotic system is given to show the feasibility of the proposed PSO algorithm. Finally, a brief conclusion is stated in Section 5.

II. PROBLEM FORMULATION

In this study, a class of *n*th-order nonlinear systems, with the input $u \in \Re$ and the output $y \in \Re$, is considered as follows

$$\dot{x}_1(t) = x_2(t),\ \dot{x}_2(t) = x_3(t),\ \ldots,\ \dot{x}_{n-1}(t) = x_n(t),$$
$$\dot{x}_n(t) = f(X, \Theta) + u(t), \tag{1}$$
$$y(t) = x_1(t),$$

where $X = [x_1, x_2, \cdots, x_n]^T \in R^n$ is the state vector of nonlinear system, $\Theta = [\theta_1, \theta_2, \cdots, \theta_m] \in R^m$ represents the actual parameter vector of nonlinear system and will be estimated, $f(\cdot)$ is the nonlinear function and its structure is assumed to be known here, *u* is the system input, and *y* is the system output available for measurement. A nonlinear estimation model is constructed to expectantly match the actual nonlinear system of (1) and is given by

$$\dot{\tilde{x}}(t)_1 = \tilde{x}_2(t),\ \dot{\tilde{x}}_2(t) = \tilde{x}_3(t),\ \ldots,\ \dot{\tilde{x}}_{n-1}(t) = \tilde{x}_n(t),$$
$$\dot{\tilde{x}}_n(t) = f(\tilde{X}, \tilde{\Theta}) + u(t), \tag{2}$$
$$\tilde{y}(t) = \tilde{x}_1(t),$$

where $\tilde{X} = [\tilde{x}_1, \tilde{x}_2, \cdots, \tilde{x}_n]^T \in \Re^n$ is the state vector of the estimation model, $\tilde{\Theta} = [\tilde{\theta}_1, \tilde{\theta}_2, \cdots, \tilde{\theta}_m] \in R^m$ is the estimated parameter vector that expectantly approximates the actual parameter vector $\Theta = [\theta_1, \theta_2, \cdots, \theta_m]$, the input *u* is the same as that of (1), and $\tilde{y}$ is the output of the estimation model. In the view of PSO algorithm, the parameter vector $\tilde{\Theta}$ is referred to as a particle which is a candidate solution of the parameter estimation problem, and many particles further constitute a population.

III. PSO ALGORITH-BASED NONLINEAR PARAMETER ESTIMATION

Kennedy and Eberhart initially proposed the PSO algorithm in 1995 [4]. It is one of the optimization techniques and a kind of evolutionary computation technique. The method has been proven to be efficient in solving optimization problem featuring nonlinearity and non-differentiability, multiple optima, and high dimensionality through adaptation. The motivation of PSO algorithm is derived from the social-psychological theory. It conducts searches using a population of particles, corresponding to individuals. Each particle represents a candidate solution to the problem. In a PSO algorithm, particles change their positions by flying around the multidimensional search space until a relatively unchanged position has been encountered, or until computational iterations are achieved. In the social science context, the PSO system combines a social-only model and a cognition-only model. The social-only component suggests that individuals ignore their own experience and adjust behavior according to the successful beliefs of individuals in the neighborhood. On the other hand, the cognition-only component treats individuals as isolated beings. A particle changes its position using these models [5][6].

Since the PSO algorithm depends only on the objective function to guide the search, it must be defined properly before the PSO algorithm can be executed. In the continuous-time domain, there are three objective functions to be often used: the integrated absolute error (*IAE*), integral of squared error (*ISE*), or integral of time weighted squared error (*ITSE*). The three integral objective functions have their own advantages and disadvantages. The feature of using *IAE* or *ISE* criteria is that its minimization can result in a response with relatively small overshoot but a long setting time because they weights all errors equally independent of time. Although the *ITSE* criterion can overcome the disadvantage of the *IAE* and *ISE* criteria, the derivation processes of the analytical formula are complex and time-consuming [6]. Fortunately, in the present work we only focus on the topic of nonlinear parameter estimation rather than specification requests of the overshoot or setting time. Therefore, the simple *ISE* criterion is adopted in the PSO algorithm and defined by

$$ISE = \int_0^{T_i} [y(t) - \tilde{y}(t)]^2 dt = \int_0^{T_i} e^2(t) dt, \tag{3}$$

where T_i is the time of integration. Additionally, other basic elements of PSO algorithm are briefly stated and defined in the following [5].

Particle $\tilde{\Theta}(k)$: It is a candidate solution of evaluating nonlinear system parameters, which is represented by an *m*-dimensional real-valued vector, where *m* is the number of estimated parameters. At *k*th iteration, the *i*th particle $\tilde{\Theta}_i(k)$ can be described as $\tilde{\Theta}_i(k) = [\tilde{\theta}_{i,1}(k), \tilde{\theta}_{i,2}(k), \cdots, \tilde{\theta}_{i,m}(k)]$, where $\tilde{\theta}_{i,j}(k)$ represents the position of the *i*th particle with respect to the *j*th dimension. The search upper and lower bounds for parameter $\tilde{\theta}$ is constrained by the interval $[\tilde{\theta}_{\min}, \tilde{\theta}_{\max}]$. If any resulting parameter violates the interval during the search, set it the corresponding bound, i.e.

$$\tilde{\theta} = \begin{cases} \tilde{\theta}_{\min} & if\ \tilde{\theta} < \tilde{\theta}_{\min} \\ \tilde{\theta} & if\ \tilde{\theta}_{mjn} \le \tilde{\theta} \le \tilde{\theta}_{\max} \\ \tilde{\theta}_{\max} & if\ \tilde{\theta} > \tilde{\theta}_{\max}. \end{cases} \tag{4}$$

Population: Population consists of many particles. In this study, we assume that a set of *H* particles builds up a population as $[\tilde{\Theta}_1, \tilde{\Theta}_2, \cdots, \tilde{\Theta}_H]^T$.

Particle velocity $V(k)$: It is the velocity of moving particles also represented by an *m*-dimensional real-valued vector. At *k*th iteration, the *i*th particle velocity $V_i(k)$ can be described as $V_i(k) = [v_{i,1}(k), v_{i,2}(k), \cdots, v_{i,m}(k)]$, where $v_{i,j}(k)$ is the velocity component of the *i*th particle with respect to the *j*th dimension.

Individual best $P(k)$: As a particle moves through the search space, it compares its objective function at the current position to the best objective function it has ever attained so far. The best position associated with the best objective function is called the individual best $P(k)$. For each particle in the population, $P(k)$ can be determined and updated during the search. To solve a minimization problem with objective function *ISE*, the individual best of the *i*th particle $P_i(k)$ is determined so that $ISE(P_i(k)) \le ISE(\widetilde{\Theta}_i(\tau))$, for $\tau \le k$. For *i*th particle, the individual best $P_i(k)$ can be expressed by $P_i(k) = [p_{i,1}(k), p_{i,2}(k), \cdots, p_{i,m}(k)]$, where $p_{i,j}(k)$ is the position of the individual best of the *i*th particle with respect to the *j*th dimension.

Global best $G(k)$: It is the best position among all of the individual best positions achieved so far. The global best $G(k) = [g_1(k), g_2(k), \cdots, g_m(k)]$ at *k*th iteration is determined such that $ISE(G(k)) \le ISE(P_i(k))$, for $i = 1, 2, \cdots, H$.

Velocity updating $V(k+1)$: Based on the individual and global best positions, the *i*th particle velocity with respect to the *j*th dimension is updated using the following equation:

$$v_{i,j}(k+1) = wv_{i,j}(k) + c_1 r_r \left(p_{i,j}(k) - \widetilde{\theta}_{i,j}(k)\right) + c_2 r_2 \left(g_j(k) - \widetilde{\theta}_{i,j}(k)\right), \quad (5)$$

where w is called the inertia weight that controls the impact of the current velocity on the next velocity, and here it is given by a constant for simplification, acceleration coefficients c_1 and c_2 are positive constants that pull each particle toward the individual best and global best positions, respectively, and r_1 and r_2 are uniformly random variables with range $[0, 1]$.

Position updating $\widetilde{\Theta}(k+1)$: Using the above velocity updating formula, each particle moves its position according to the following equation:

$$\widetilde{\theta}_{i,j}(k+1) = \widetilde{\theta}_{i,j}(k) + v_{i,j}(k+1). \quad (6)$$

Termination condition: There are two conditions under which the PSO algorithm will terminate: (a) the objective function is less than certain pre-specified value or (b) the number of iterations achieves the maximum allowable number *N*. In this study, the second criterion is adopted to terminate the search process.

The basic steps of the PSO-based optimization process for the nonlinear system parameter estimation can be summarized as follows:

Data: An unknown nonlinear systems described by (1), time of integration T_i for the objective function in (3), lower and upper bounds of the search interval $[\widetilde{\theta}_{\min}, \widetilde{\theta}_{\max}]$ in (4), population size *H*, inertia weight *w* and acceleration coefficients c_1 and c_2 in (5), and number of iterations *N*.

Goal: Identify the optimal set of nonlinear system parameters in (2) by minimizing the objective function *ISE* in (3).

Generate an initial population containing H particles drawn at random uniformly from the interval $[\widetilde{\theta}_{\min}, \widetilde{\theta}_{\max}]$.

$k = 1$;

While $(k \le N)$

Calculate the objective function $ISE(\widetilde{\Theta}_i)$ for each $\widetilde{\Theta}_i$ using (3).

For each particle, find the individual best $P_i(k)$ such that $ISE(P_i(k)) \le ISE(\widetilde{\Theta}_i(\tau))$, for $\tau \le k$.

Find the global best $G(k)$ such that $ISE(G(k)) \le ISE(P_i(k))$, for $i = 1, 2, \cdots, H$.

Perform the velocity updating using (5) for each position of a particle.

Perform the position updating using (6) for each position of a particle.

Check each position and apply the upper and lower bounds in (4) if the position falls outside of the search interval.

End

Fig. 1 shows a simple block diagram of the PSO-based nonlinear system parameter estimation process with measurement noise *w* taken into consideration.

IV. ILLUSTRATIVE EXAMPLE

This section demonstrates the effectiveness of the proposed PSO-based optimization process in estimating the parameters of nonlinear systems. The Genesio-Tesi nonlinear chaotic system is illustrated. Its third-order differential dynamic equation, with initial values $x_1(0) = 1$, $x_2(0) = 1$, and $x_3(0) = 1$, is described by [16]

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= x_3, \\ \dot{x}_3 &= -\theta_1 x_1 - \theta_2 x_2 - \theta_3 x_3 + x_1^2 + u, \\ y &= x_1, \end{aligned} \quad (7)$$

where $\theta_1 = 6$, $\theta_2 = 2.92$, $\theta_3 = 1.2$. Fig. 2 shows three-dimensional phase portrait of states x_1, x_2, and x_3 to reveal the chaotic behavior of Genesio-Tesi system when $u = 0$.

In the present simulations, the packet software of Borland C++ is programmed to implement the above PSO algorithm. We assume that the sampling time is set to 0.01 for simulating the differential chaotic dynamic equation. The exciting input u is generated randomly within the interval $[-1, 1]$ and the values assigned to the variables of the PSO algorithm in the current search for the optimal nonlinear system parameters listed in Tab. 1. To examine the feasibility and effectiveness of the PSO search algorithm to identify global optimal solutions, several cases are provided. Two different kinds of measurement noises with uniform distributions are added to the output of the unknown nonlinear system, i.e., $w = 0$ and $w \in [-0.01, 0.01]$, respectively. Furthermore, for each measurement noise four optimization runs with different random sets of initial populations are also considered. Tabs. 2 and 3 compare the actual parameter values of the Genesio-Tesi nonlinear chaotic system with the optimal solutions obtained from the proposed PSO-based method (Runs 1-4). It is clear that the simulation results obtained by the PSO-based optimization method closely approximate to the actual parameter values even though the measurement noise occurs. Figs 3 and 4 then display the corresponding convergence curves of the objective functions with respect to the number of updating global best. These figures obviously show that all of curves quickly converge to zero. This means that the proposed method has the excellent ability for solving the parameter estimation problem of the Genesio-Tesi nonlinear system.

V. CONCLUSION

This paper has successfully proposed the PSO algorithm to optimally solve the parameter estimation problem for a class of nonlinear systems. In the PSO algorithm, the unknown system parameters are arranged and referred to as a particle, and many particles further build up a population. The velocity and position updating formulas are performed for each particle such that the objective function of integral of squared error is minimized. To demonstrate the estimation performance, some examining conditions are considered including different sizes of noises and different random sets of initial populations. From several simulation results, it is rather obvious that the unknown system parameters can be accurately solved by the proposed PSO algorithm.

ACKNOWLEDGEMENT

This work was supported by the National Science Council of Taiwan under the grant NSC 100-2221-E-366-002.

REFERENCES

[1] Omatu S, Khalid M, Yusof R, Neuro-control and its applications. London: Springer; 1995.

[2] Wang LX, Course in fuzzy systems and control. NJ: Prentice-Hall; 1997.

[3] Astrom KJ, Wittenmark B, Adaptive control. Massachusetts: Addison-Wesley; 1995.

[4] Kennedy J, Eberhart R, Particle swarm optimization, in Proc. IEEE Int. Conf. Neural Networks, vol. IV, Perth, Australia, 1995, pp.1942-1948.

[5] Abido MA, Optimal design of power-system stabilizers using particle swarm optimization. IEEE Transactions on Energy Conversion 2002; 17:406-413.

[6] Gaing ZL, A particle swarm optimization approach for optimum design of PID controller in AVR system. IEEE Transactions on Energy Conversion 2004; 19:384-391.

[7] Chang WD, Chang DM, Design of a hinger-order digital differentiator using a particle swarm optimization approach. Mechanical Systems and Signal Processing 2008; 22:233-247.

[8] Acharjee P, Mallick S, Thakur SS, Ghoshal SP, Detection of maximum loadability limits and weak buses using chaotic PSO considering security constraints. Chaos, Soliton & Fractals 2011; 44:600-612.

[9] Chiou JS, Tsai SH, Liu MT, A PSO-based adaptive fuzzy PID-controllers. Simulation Modeling Practice and Theory 2012; 26:49-59.

[10] Shayeghi H, Jalili A, Shayanfar HA, Multi-stage fuzzy load frequency control using PSO. Energy Conversion and Management 2008; 49:2570-2580.

[11] Chatterjee A, Matsuno F, A Geese PSO tuned fuzzy supervisor for EKF based solutions of simultaneous localization and mapping (SLAM) problems in mobile robots. Expert Systems with Applications 2010; 37:5542-5548.

[12] Elabbasy EM, Agiza HN, El-Dessoky MM, Adaptive synchronization of Lü system with uncertain parameters. Chaos, Solitons & Fractals 2004; 21:657-667.

[13] Chen S, Lü J, Parameters identification and synchronization of chaotic systems based upon adaptive control. Phys Lett A 2002; 299:353-358.

[14] Kim JH, Park CW, Kim E, Park M, Adaptive synchronization of T-S fuzzy chaotic systems with unknown parameters. Chaos, Solitons & Fractals 2005; 24:1353-1361.

[15] Yassen MT, Adaptive synchronization of Rossler and Lü systems with fully uncertain parameters. Chaos, Solitons & Fractals 2005; 23:1527-1536.

[16] Chen MY, Zhou DH, Shang Y, A sliding mode observer based secure communication shceme. Chaos, Solitons & Fractals 2005; 25:573-578.

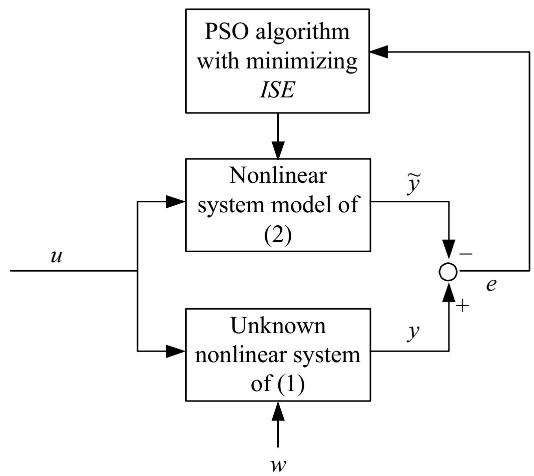

Figure 1. PSO algorithm estimation of nonlinear systems with measurement noise.

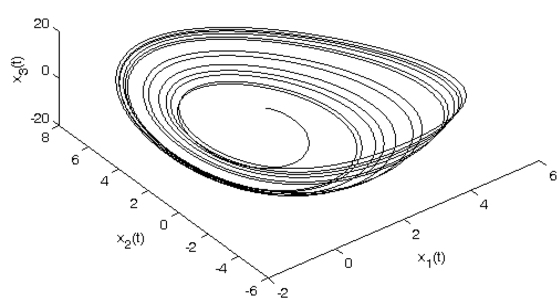

Figure 2. Three-dimensional phase portrait of states x_1, x_2, and x_3 for the unforced Genesio-Tesi chaotic system.

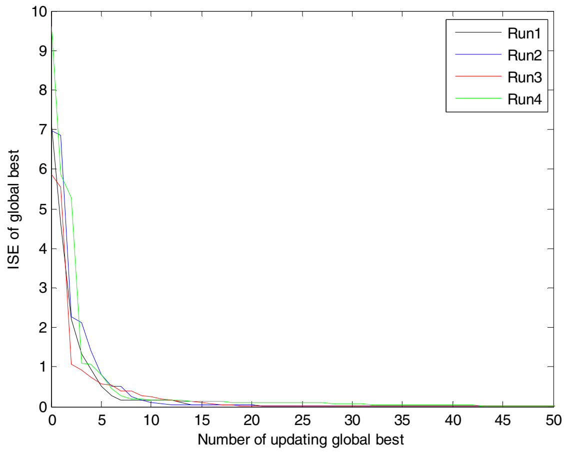

Figure 3. Convergence curves of objective function *ISE* under $w=0$.

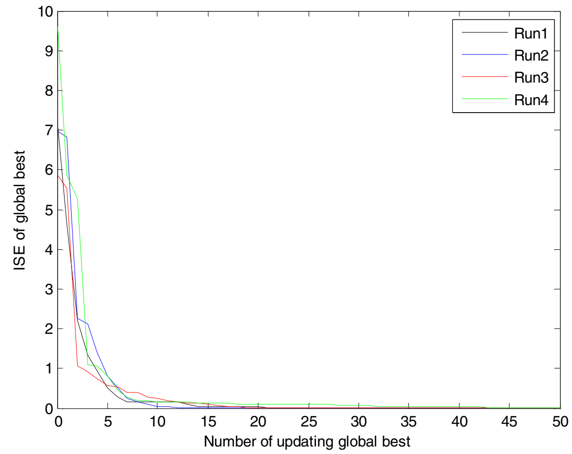

Figure 4. Convergence curves of objective function *ISE* under $w\in[-0.01, 0.01]$.

TABLE 1. Variables of PSO algorithm used in nonlinear Genesio-Tesi chaotic system parameter estimation.

Variables	Values
Integration time T_i	4
Lower and upper bounds $[\tilde{\theta}_{\min}, \tilde{\theta}_{\max}]$	$[0, 10]$
Population size H	20
Inertia weight w	0.5
Coefficient c_1	2
Coefficient c_2	2
Number of iterations N	500

TABLE 2. Comparison of actual system parameters with those estimated by PSO-based method under $w=0$.

	Actual parameters	Run 1	Run 2	Run 3	Run 4	Mean
θ_1	6	6	6	6	6	6
θ_2	2.92	2.92	2.92	2.92	2.92	2.92
θ_3	1.2	1.2	1.2	1.2	1.2	1.2

TABLE 3. Comparison of actual system parameters with those estimated by PSO-based method under $w\in[-0.01, 0.01]$.

	Actual parameters	Run 1	Run 2	Run 3	Run 4	Mean
θ_1	6	6.0011	5.9960	6.0014	5.9962	5.9987
θ_2	2.92	2.9196	2.9216	2.9196	2.9216	2.9206
θ_3	1.2	1.2004	1.1983	1.2005	1.1984	1.1994