ISSN 1392 - 1207. MECHANIKA. 2010. Nr.6(86)

Real-time parameter identification for highly coupled nonlinear systems using adaptive particle swarm optimization

B. Ranjbar Sahraei*, A. Nemati, A. A. Safavi*****

**School of Electrical and Computer Engineering, Shiraz University, Shiraz, Iran, E-mail: b.ranjbar@ieee.org*

***School of Mechanical Engineering, Sharif University of Technology, Tehran, Iran, E-mail: arnemati@gmail.com*

****School of Electrical and Computer Engineering, Shiraz University, Shiraz, Iran, E-mail: safavi@shirazu.ac.ir*

1. Introduction

Model identification is one of the most important requirements for advanced control systems. Fast response and quick disturbance recovery of an advanced controller [1, 2] cannot be achieved without an accurate model of the plant. Although after years of development linear methods for model identification [3 - 5] have become mature, nonlinear identification methods are still needed in high performance applications to maintain sufficient identification accuracy over the entire operation range.

When analytical nonlinear methods, e.g. least-squared algorithms based on quadratic error functions [6], are not able to find a global solution for nonlinear systems, a promising alternative to these traditional methods is the swarm intelligence based optimization algorithms. In references [7-11], Genetic Algorithm (GA) is applied to some optimization problems and good results are achieved, however, the number of manipulations and required memory increases with and increase in the dimension of searching space. Therefore it is difficult for the GA algorithms to maintain searching velocity and convergence for the real-time identification applications. Referring to [12-15] Particle Swarm Optimization (PSO) algorithm acts well when dealing with multimodal and multidimensional problems. PSO algorithm is theoretically simple, and computationally efficient. It provides many advantages for complex engineering problems.

In this paper, a parameter identification approach using Adaptive Particle Swarm Optimization (APSO) is utilized, which is a development of PSO based identification approach [13]. Furthermore, the proposed method is applied to a real ball on plate setup, which is a typical example of highly coupled, nonlinear electromechanical system.

The ball on plate system is an extension of the traditional ball and beam [16] that moves a ball on a rigid or flexible surface. The mechatronic design of a ball on plate, which uses inexpensive materials with a simple setup, can be valuable. In addition, control of such a 2D coupled and nonlinear dynamical model is a very interesting and challenging example for studying and testing various control methods.

In principle, rolling resistance of a ball on plate system is required for modeling the system, but in practice, the varying nature of such a resistance with the ball's material, mass and dimensions make it difficult to be measured. To the best of our knowledge, such a real-time identification solution has not been reported before. It is the first report of identifying this critical value for modeling a ball on plate system.

The rest of this paper is organized as follows: Section 2 provides a brief introduction on ball on plate mechatronic design. System modeling and problem formulation will be discussed in Section 3. APSO will be briefly introduced in Section 4. Section 5 shows an APSO approach for ball on plate parameter identification. Simulation results and experimental verification are presented in Section 6 and 7 respectively and concluding remarks are given in Section 8.

2. Ball on plate mechatronic design

This section describes the design and development of a ball on plate system. This electromechanical balancing system, due to its inherent complexity, presents a challenging design problem. In the context of such an unconventional problem, the relevance of mechatronic design methodology becomes apparent.

One of the main differences between a ball on plate setup and a traditional ball and beam system is the ball position sensing dimensionality. Various 2D position sensing methods can be considered like touch sensors, image-processing with overhead camera, resistive grid, infrared sensors, etc. [17, 18].

Fig. 1 Designed ball on plate test-bed

In the proposed mechatronic design, an overhead digital camera is used, which is economically efficient and makes the mechanical setup of the system optimized in terms of time and cost.

The next problem is choosing the proper actuation system to change the plate orientations in the space with 2-degrees of freedom. Some proposed methods are "using of a pair of linear actuators in two corners of plate", "using of cable and pulley to turn the plate with two motors", "using of four-beam linkage with servomotors", etc. The proposed mechatronic design uses cables and pulleys in order to rotate the plate fast and accurate.

Generally, the designed ball on plate balancing system (Fig. 1) compared to the other existing ones [18, 19] is cheaper, quicker and more reliable to build and more compact.

3. Ball on plate modelling

In this section, the differential equations of the ball on plate system are driven, while the following assumptions hold.

Assumption1. Plate angles of rotation are small.

Assumption2. The Ball remains in contact with the plate and rolling occurs without slipping at any time.

Assumption3. The Ball does not rotate about its vertical axis.

Assumption4. The plate is always considered as a complete square in the sight of digital camera.

Assumption5. Distance between the camera and the ball remains constant.

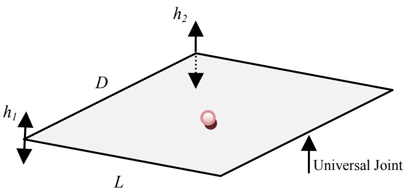
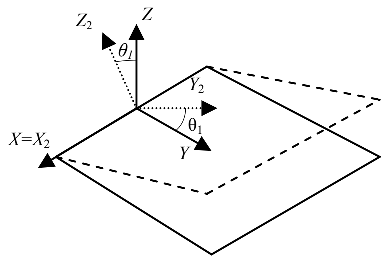
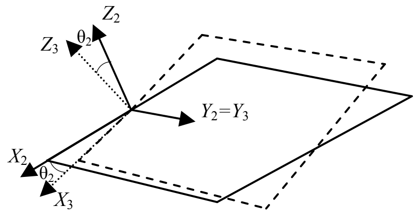


Fig. 2 Plate rotating schematic

The slope of the plate can be adjusted by setting the height of two adjacent corners (h_1 and h_2) as shown in Fig. 2.



a



b

Fig. 3 Physical model (a) when plate rotates about $X = X_2$ axis, (b) when plate rotates about new axis ($Y_2 = Y_3$)

Physical model of the ball on plate setup is shown in Fig. 3 and differential equations of the ball on plate system are derived in Eq. (1)

$$
\begin{cases}
\dot{x}_1 = x_3 \\
\dot{x}_2 = x_4 \\
\dot{x}_3 = -\dfrac{1}{\left(mr^2 + I_{ball}\right)}\left(\dfrac{mr^2 z\left(\dfrac{\ddot{h}_1}{2} - \dfrac{\ddot{h}_2}{2}\right)}{D} - \dfrac{gmr^2\left(\dfrac{h_1}{2} - \dfrac{h_2}{2}\right)}{D} + \dfrac{gmr^2 \mu x_3}{S} - \dfrac{mr^2 \mu x_1 x_3\left(\dfrac{\ddot{h}_1}{2} - \dfrac{\ddot{h}_2}{2}\right)}{DS} + \dfrac{mr^2 \mu x_2 x_3\left(\dfrac{\ddot{h}_1}{2} + \dfrac{\ddot{h}_2}{2}\right)}{LS}\right) \\
\dot{x}_4 = -\dfrac{1}{\left(mr^2 + I_{ball}\right)}\left(-\dfrac{mr^2 z\left(\dfrac{\ddot{h}_1}{2} + \dfrac{\ddot{h}_2}{2}\right)}{D} + \dfrac{gmr^2\left(\dfrac{h_1}{2} + \dfrac{h_2}{2}\right)}{D} + \dfrac{gmr^2 \mu x_3}{S} - \dfrac{mr^2 \mu x_1 x_3\left(\dfrac{\ddot{h}_1}{2} - \dfrac{\ddot{h}_2}{2}\right)}{DS} + \dfrac{mr^2 \mu x_2 x_3\left(\dfrac{\ddot{h}_1}{2} + \dfrac{\ddot{h}_2}{2}\right)}{LS}\right) \\
y_1 = x_1 \\
y_2 = x_2
\end{cases}
\tag{1}
$$

where

$$S = \sqrt{x_3{}^2 + x_4{}^2} \tag{2}$$

State variables x_1, x_2, x_3 and x_4 represent the ball position and velocity in x and y directions respectively. Other model parameters are introduced in Table 1.

Table 1

Model parameter nomenclature

m, kg	mass	g, m/s^2	gravity
r, m	ball radius	D, m	plate width
I_{ball}, kgm^2	ball mass moment of inertia	L, m	plate length
μ	rolling resistance coefficient	z, m	distance between universal joint and plate

4. Particle swarm optimization (PSO)

4.1. Simple PSO

Inspired by the social behaviour of natural organisms like fish schooling and bird flocking, Kennedy and Eberhart introduced PSO in 1995 [13] for the first time. In PSO, the population (swarm) of individuals (particles) in the search space is randomly initialized. Each individual in PSO moves through the searching space and remembers the best position it has ever seen. Members of a swarm communicate good positions to each other and dynamically update their positions based on these good positions.

At each iteration, each particle can adjust its velocity vector, based on its momentum and the influence of its best position (p_{id}) as well as the best position of its neighbours (p_{gd}), and then concluded to a new position that the particle is to fly to. Supposing the dimension of searching space is D, the total number of particles is n, the position of the ith particle can be expressed as vector $X_i = (x_{i1}, x_{i2}, \ldots, x_{iD})$; the best position of the ith particle searched until now is denoted as $P_{id} = (p_{i1}, p_{i2}, \ldots, p_{iD})$, and the best position of the whole swarm searched until now is denoted as vector $P_{gd} = (p_{g1}, p_{g2}, \ldots, p_{gD})$; the velocity of the ith particle is represented by vector $V_{id} = (v_{i1}, v_{i2}, \ldots, v_{iD})$. Then the traditional PSO is described as

$$V_{id}(t+1) = wV_{id}(t) + c_1 r_1 (P_{id}(t) - X_i(t)) + c_2 r_2 (P_{gd}(t) - X_i(t)) \tag{3}$$

$$X_i(t+1) = X_i(t) + V_{id}(t) \tag{4}$$

where c_1, c_2 are acceleration constants with positive values; r_1 , r_2 are random numbers between 0 and 1; w is the inertia weight. In addition to the w, c_1 and c_2 parameters, adjusting the parameters w, c_1 and c_2, is needed to achieve its best searching ability.

4.2. Adaptive PSO (APSO)

Good performance of traditional PSO is highly depended on its parameter settings [13, 15]. Therefore, in many applications the performance of general PSO is not satisfactory in the whole experiment. The particle velocity is very important factor because wrong direction and improper magnitude can lead the system divergent and it is essential to set the parameters based on characteristics of the problem. Inertia weight controls the influence of the previous velocity on the current one. Therefore, it is important to make balance between exploration and exploitation by the proper adaptive value of the inertia weight. Too much stress on exploration results in a pure random search whereas too much exploitation results in a pure local search. Using [15] w is made dependable with objective function of the locally best and globally best solutions and adaptive inertia weight is chosen.

Consider the cost function $C(.)$ belongs to $R^D \to R$. The update law for inertia weight of the ith particle (X_i) is defined as follows

$$w_i = w_{min} + \frac{C(P_{gd})\left(C(X_i) - C(P_{gd})\right)}{C(X_i)\left(C(X_i) - C(P_{id})\right)} \tag{5}$$

where w_{min} is a lower bound for the inertia weight, and some w_{max} is defined by designer.

For the problems involved with higher cost, higher learning factors are required to provide higher step size for maintaining the sufficient global exploration. Whereas, fine-tuning of the concerned solution needs lower learning factors when objective function of that solution is nearer to the best solution. Considering this phenomenon learning factors update law is proposed as following

$$c_{1i} = \sqrt{C(X_i)/C(P_{id})} \tag{6}$$

$$c_{2i} = \sqrt{C(X_i)/C(P_{gd})} \tag{7}$$

From the above equations, it can be concluded that learning factors are always greater than unity.

Another improvement in proposed APSO scheme is a modification on random coefficients r_1 and r_2 in Eq. (1).

Since we generate two random parameters r_1 and r_2 independently, there are cases in which they both are too small. In this case, both the exploration and exploitation behaviour are neglected, and in some cases when both of the coefficients are too large the exploration and exploitation behaviour are overused. Therefore, the convergence performance of the algorithm is undermined in both cases. In other words, the two random weighting parameters cannot be completely independent. By assuming the sum of two inter-related weighting parameters as 1, we can generate one single random number r_1 and calculate r_2 as

$$r_2 = 1 - r_1 \tag{8}$$

Using modifications on inertia weights from Eq. (5), learning factors from Eqs. (6) and (7), and random coefficients generation Eq. (8), the adaptive form of Eq. (1) is considered as following

$$V_{id}(t+1) = w_i V_{id}(t) + c_{1i} r_1 (P_{id}(t) - X_i(t)) + c_{2i}(1 - r_1)(p_{gd}(t) - X_i(t)) \tag{9}$$

APSO algorithm can be divided into six steps and represented as following

[step1] initialize the population set
[step2] determine cost function for each particle
[step3] select pbest solution
[step4] update the whole population
[step5] select gbest solution
[step6] check convergence, if the terminal condition has not been satisfied go back to step2.

From the aforementioned algorithm of APSO and the updating rules, it can be seen that APSO is very simple in concept and easy in realization. In the next section, the APSO algorithm will be used to solve parameter identification problem.

5. Real-time identification design

Generally, identification methods are classified into structured and unstructured methods. The APSO identification approach proposed in this study is an effective structured identification method. Therefore, it is assumed that the structure of the plant is known and unknown parameters can be estimated using proposed method. The basic idea is to achieve a cost function, which represents how well the model output tracks the system output. This cost function can be defined as the difference between system and model outputs.

Consider a general state space form of system dynamic as following

$$\begin{cases} \dot{X} = F(X, P_k, P_u, U) \\ Y = G(X, P_k, P_u) \end{cases} \tag{10}$$

where X is the system state vector, Y represents the output vector, functions $F(.)$ and $G(.)$ can be either linear or nonlinear and the vectors P_k and P_u are the known and unknown parameters respectively.

To estimate the plant output Eq. (10) following model dynamic is introduced

$$\begin{cases} \dot{\hat{X}} = F(\hat{X}, P_k, \hat{P}_u, U) \\ \dot{\hat{Y}} = G(\hat{X}, P_k, \hat{P}_u) \end{cases} \tag{11}$$

where $\hat{X}$ is the estimated states vector, $\hat{Y}$ represents the estimated output and U is the same control signal used in Eq. (10).

In real-time applications after the N th sampling, a cost function is evaluated using the following weighted quadratic function

$$C(\hat{P}_u) = \sum_{i=1}^{N} (Y_i - \hat{Y}_i)^T W (Y_i - \hat{Y}_i) \tag{12}$$

where Y_i and $\hat{Y}_i$ are system output and model output in the i^{th} sampling and W is usually chosen as a diagonal positive definite matrix. Obviously, the resultant cost is a function of the estimated parameter vector $\hat{P}_u$.

Now, the identification problem is replaced with a multidimensional optimization problem. Although many identification methods can be used to solve such an optimization problem, a highly coupled nonlinear multidimensional optimization cannot be solved with any traditional methods. Therefore, in this research an optimization algorithm based on swarm intelligence methods is proposed to find the most optimized solutions in real-time applications.

In real-time identification applications, fast identification algorithms have a very important effect on system performance. Therefore, in the proposed method; in each iteration, half of the initial population of the APSO algorithm is chosen randomly and the rest is chosen from the best individuals in previous iterations.

In the next two sections, a ball on plate system is chosen as an experimental setup. The rolling resistance μ of the electromechanical system is assumed unknown in Eq. (1).

6. Simulation results

In this section, a ball on plate setup is simulated in MATLAB® to illustrate the performance of the proposed APSO identification approach applied to electromechanical system parameter identification. Although Parameter identification can be used to design some robust and high performance controllers, this is not the main goal of this paper and in the next two sections, just a simple proportional-integral-derivative (PID) controller, which is widely used in industrial control applications, is applied to the plant. Nominal parameters of the simulated ball on plate dynamical model Eq. (1) are listed in Table 2.

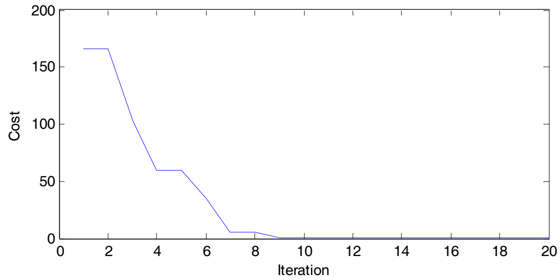
To show the effectiveness of the identification method, the rolling resistance coefficient used in Eq. (1) is assumed unknown. However, more parameters (e.g. m, r or I) can easily assumed unknown and good simulation results will be achieved.

During simulations, each swarm generation contains 10 particles. Coefficient w_{min} is set to 0.8 in Eq. (5) and L_1, L_2 and w are determined adaptively using Eqs. (5), (6) and (7). Fig. 4 shows the cost function value and unknown identified value for within 20 iterations.

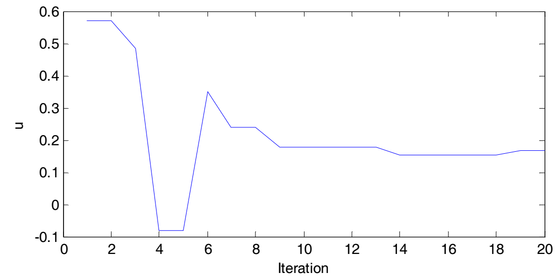
Table 2

Nominal parameters used for modelling

	Value		Value
m	245 gr	z	2 cm
r	13.5 mm	$x_1(0)$	-30
μ	0.15	$x_2(0)$	30
D	60 cm	$x_3(0)$	0
L	60 cm	$x_4(0)$	0
I_{ball}	17.8 kg.cm^2	g	9.806 m/s^2



a



b

Fig. 4 Offline APSO identification (a) Cost function (b) estimated rolling resistance

Fast convergence and identification accuracy of the approach make it suitable for applications that require real-time solutions.

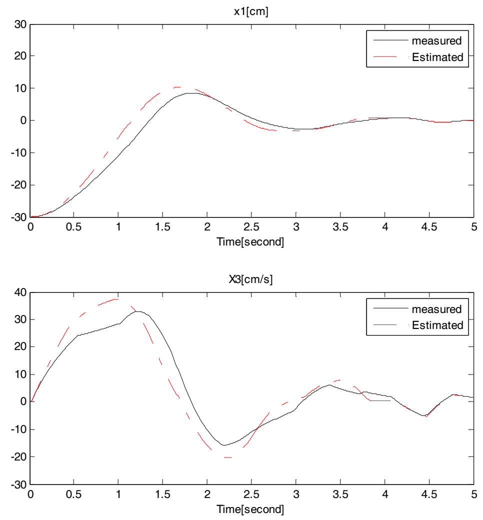


Fig. 5 Model ouputs tracking the simulated outputs

To run the real-time simulation, the identification algorithm is repeated during the simulation. The maximum number of allowed iterations after each data sampling is set to 20. In addition, half of the initial particles used after each new sampling consists of the best-estimated parameters used in previous iterations.

Fig. 5 shows the tracking performance of the model output and simulated system.

Obviously, after less than 4.5 seconds model output tracks simulated plant output perfectly.

7. Experimental verification

In order to verify the effectiveness of the proposed identification method, experiments are implemented on a real ball on plate setup (Fig. 1). This section presents the experimental results.

As mentioned in Section 2, all the components used for mechatronic setup (Fig. 1) are inexpensive and commercially available. A pair of 300 rpm DC motors derived by a microcontroller-based driver is used for changing the altitude of two adjacent corners of the plate. Typical encoders measure the altitude and send data to the PC by a DAQ card. Any typical PC can be used to run image-processing software. Main controller laws and identification algorithms.

The procedure of this experimental verification is similar to the simulation described previously in section 6, except that the real data are utilized for identification procedure. The experimental-identification results are presented for five different balls each one with a specific type of material. These different balls used in the experiment and the rolling resistance coefficients identified with proposed APSO algorithm, are listed in Table 3.

Table 3

Experimentally identified rolling resistance coefficient (RRC)

No.	Ball	Mass, gr	Dia, mm	RRC
1	Steel (solid)	245.0	27.0	0.1635
2	Steel (shell)	75.0	27.0	0.0815
3	Plastic (solid)	11.0	30.0	0.0325
5	Sponge (solid)	3.0	35.0	0.0215

To evaluate the performance of identification method the initial ball position is $[x, y, \dot{x}, \dot{y}] = [-30, 30, 0, 0]$. The plate dimensions are $60\,\text{cm} \times 60\,\text{cm}$ and the steel solid ball is used in this experiment. As shown in Fig. 6 estimated model outputs are compared with the measured data. It is shown that using some arbitrary rolling resistance coefficient in the model dynamics, the model outputs (E_2) do not track the plant outputs; however, with using the APSO identifier to identify the unknown parameter; the effective tracking of (E_1) is proved.

Another advantage of the proposed method is the inherent noise filtering feature of the method. In many real-time electromechanical experiments, the measured data consists of noises, which are filtered digitally in many previous published works. However, when using APSO, there is no need to an additional filtering algorithm and this leads to speed up of the real-time experiments by omitting the filter block during system identification and control.

In order to have a quantitative comparison between differenet identification methods based on swarm intelligence, $e = \| y - \hat{y} \|$ is defined as the identification error and Integral Absolute Error (IAE) is selected as the criterion. Table 4 gives the IAE values for three Swarm Inteligence (SI) based identification methods. Genetic algorithm (GA) identification results are presented as a traditional SI. PSO and APSO are the next two methods which are compared.

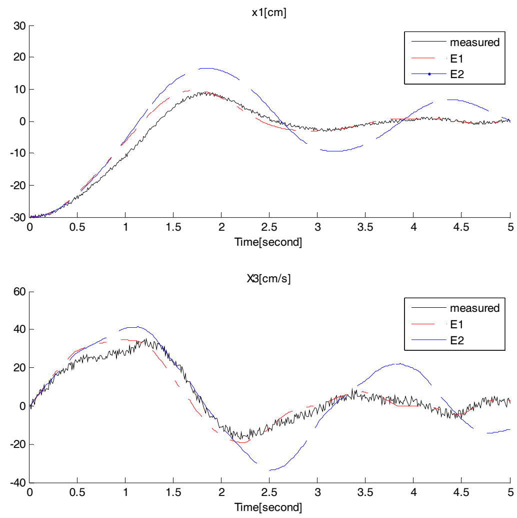


Fig. 6 Estimated model outputs tracking the real data. "*measured*" is the raw plant output, "*E*1" is the model estimated output using APSO and "*E*2" shows the model estimated output with unidentified parameter

Table 4

IAE comparison between GA, PSO and APSO

	$\int_0^{2.5} \lvert e(t) \rvert dt$	$\int_0^{5} \lvert e(t) \rvert dt$
GA	21.1243	35.3120
PSO	15.5874	25.1007
APSO	15.0351	17.1159

Using Table 4 we can see that Adaptive PSO compared to original PSO has more convergence accuracy as exploitation plays a more important role than exploration, while the particles converge to the final answer. Also comparing GA results with other two methods shows the GA performance criteria is unsatisfactory at all. The main reason for that, is its time consuming manipulations like selection, crossover and mutation [10] which makes the algorithm not suitable for real-time parameter identifications.

8. Conclusion

An identification scheme has been proposed in order to overcome the limitations in the suitability and effectiveness of existing methods for identification of multimodal and multidimensional electromechanical systems. These limitations relate both to the computational effectiveness in real-time identification, and also to the global optimization of nonlinear, highly constrained optimization problems. Simulation results have been used to confirm the efficiency of the proposed approach. In addition, this approach is applied to a ball on plate test-bed, in order to directly verify its capability. Rolling resistance of the ball is identified using the proposed method and quantitative comparisons have been presented to prove its superiority, compared with some existing real-time identification methods. Potentially, the proposed identification method can be used to design more robust controllers for nonlinear constrained, multiple-input multiple-output (MIMO) systems with parameter uncertainties.

References

1. **Tatjewski, P.** Advanced control and on-line process optimization in multilayer structures. -Annual Reviews in Control, April 2008, 32(1), p.71-85.
2. **Stan, S.-D., Bălan, R., Mătieş, V., Rad., C.** Kinematics and fuzzy control of ISOGLIDE3 medical parallel robot. -Mechanika. -Kaunas: Technolgija, 2009. Nr.1(75), p.62-66.
3. **Liliana, R. Castro, Osvaldo, E. Agamennoni, Marcela, P. Álvarez**, From linear to nonlinear identification: One step at a time. -Mathematical and Computer Modelling, 2010, 51(5-6), p.473-486.
4. **Chao-Ming Ying, Babu Joseph.** Identification of linear systems using polynomial kernels in the frequency domain. -Journal of Process Control, 2002, 12(5), p.645-657.
5. **Caigny, J.D., Camino, J.F., Swevers**, **J.** Interpolating model identification for SISO linear parameter-varying systems. -Mechanical Systems and Signal Processing, November 2009, 23(8), p.2395-2417.
6. **Jones, J.C.P., Muske**, **K.R.** Identification and adaptation of linear look-up table parameters using an efficient recursive least-squares technique. -ISA Transactions, October 2009 , 48(4), p.476-483.
7. **Banzhaf, W., Nordin, P., Keller, R.E., Francone**, **F.D.** Genetic Programming: An Introduction: On the Automatic Evolution of Computer Programs and Its Applications. -Morgan Kaufmann, 1998.-451p.
8. **Oysu, C., Bingul**, **Z.** Application of heuristic and hybrid-GASA algorithms to tool-path optimization problem for minimizing airtime during machining. -Engineering Applications of Artificial Intelligence, April 2009, 22(3), p.389-396.
9. **Belevičius, R., Šešok**, **D.** Global optimization of grillagees using genetic algorithms, -Mechanika. -Kaunas: Technologija, 2008, Nr.6(74), p.38-44.
10. **Lute, V., Upadhyay, A., Singh**, **K.K.** Computationally efficient analysis of cable-stayed bridge for GA-based optimization. -Engineering Applications of Artificial Intelligence, June 2009, 22(4-5), p.750-758.
11. **Antal, T.A.** A new algorithm for helical gear design with addendum modification. -Mechanika. -Kaunas: Technologija, 2009, Nr.3(77), p.53-57.
12. **Kennedy, J., Eberhart**, **R.** Particle swarm optimization. -In IEEE Int. Conf. on Neural Networks. 1995.
13. **Liu, L., Liu, W., Cartes**, **D.A.** Particle swarm optimization-based parameter identification applied to permanent magnet synchronous motors. -Engineering Applications of Artificial Intelligence, October 2008 , 21(7), p.1092-1100.
14. **Shi, Y., Eberhart, R.C.** Fuzzy adaptive particle swarm optimization. -In IEEE Int. Congress on Evolutionary Computation. May 2001.
15. **Acharjee, P., Goswami**, **S.K.** Expert algorithm based on adaptive particle swarm optimization for power flow analysis. -Systems with Applications, April 2009, Part 1. 36(3), p.5151-5156.
16. **Oh, S.K., Jang, H.J., Pedrycz**, **W.** The design of a fuzzy cascade controller for ball and beam system: A study in optimization with the use of parallel genetic algorithms. -Engineering Applications of Artificial Intelligence, March 2009, 22(2), p.261-271.
17. **Awtar, S., Bernard, C., Boklund**, **N., Master, A., Ueda, D., Craig, K.** Mechatronic design of a ball-on-plate balancing system. -Mechatronics, March 2002, 12(2), p.217-228.
18. **Fan, X., Zhang, N., Teng**, **S.** Trajectory planning and tracking of ball and plate system using hierarchical fuzzy control scheme. -Fuzzy Sets and Systems, June 2004, 144(2), p.297-312.

B. Ranjbar Sahrei, A. Nemati, A.A. Safavi

REALAUS LAIKO NUSTATYMAS OPTIMIZUOJANT ADAPTYVIŲ STIPRIAI SUJUNGTŲ DALELIŲ SPIEČIAUS NETIESINES SISTEMAS

R e z i u m ė

Šiame straipsnyje pasiūlytas adaptyvus detalių spiečiaus optimizavimo metodas stipriai sujungtoms elektromechaninėms sistemoms identifikuoti. Naudojant kai kurių stipriai sujungtų dalelių spiečiaus modifikacijas skaičiavimo efektyvumas padidėjo. Šiuo būdu yra pagerinta realaus laiko identifikavimo procedūra. Pasiūlyto identifikavimo metodo efektyvumui padidinti papildomai yra panaudotas realus maketas su ant plokštumos sumontuotu rutuliu ir sukurtas šio įtaiso dinaminis modelis. Tiek imitavimo, tiek eksperimento rezultatai rodo, kad parametrų identifikavimo, naudojant pasiūlytąjį algoritmą, rezultatai yra kur kas geresni, nei panaudojant kitus identifikavimo metodus, pagrįstus tradicinį dalelių spiečiaus optimizavimu ir genetiniu algoritmu.

B. Ranjbar Sahrei, A. Nemati, A.A. Safavi

REAL-TIME PARAMETER IDENTIFICATION FOR HIGHLY COUPLED NONLINEAR SYSTEMS USING ADAPTIVE PARTICLE SWARM OPTIMIZATION

S u m m a r y

The In this paper, an Adaptive Particle Swarm Optimization (APSO) method is proposed for parameter identification of highly coupled electromechanical systems. Using some modifications on the APSO, better computational efficiency is achieved. In this way, the speed of real-time identification procedure is improved. In addition, to show the effectiveness of the proposed method, it is implemented on a real ball on plate setup and its dynamic model is achieved. Both the simulation and the experimental results show that parameter identification of the proposed algorithm is significantly improved when compared with other existing identification methods based on the traditional PSO and Genetic Algorithm (GA).

Б. Ранибар Сахрей, А. Немати, А. А. Сафави

ОПРЕДЕЛЕНИЕ ПАРАМЕТРОВ РЕАЛЬНОГО ВРЕМЕНИ ДЛЯ СИЛЬНО СВЯЗАННЫХ НЕЛИНЕЙНЫХ СИСТЕМ ИСПОЛЬЗУЯ АДАПТИВНУЮ ОПТИМИЗАЦИЮ РОЯ ЧАСТИЦ

Р е з ю м е

В данной статье метод адаптивной оптимизации роя частиц предложен для идентификации сильно связанных электромеханических систем. При использовании некоторых сильно связанных модификациях роя частиц получен более эффективный расчет. Таким образом улучшена процедура идентификации реального времени. Дополнительно для оценки эффективности предложенного метода идентификации создан реальный макет со смонтированным шаром на плоскости и создана его динамическая модель. Результаты имитации и эксперимента показывают, что имитация параметров при использовании предложенного алгоритма является существенно предпочтительнее по сравнению с другими методами, использующими традиционную оптимизацию роя частиц и генетический алгоритм.

Received July 12, 2010
Accepted December 07, 2010