Physics Letters A ••• (••••) •••–•••

Contents lists available at ScienceDirect

Physics Letters A

www.elsevier.com/locate/pla

Parameter estimation of Lorenz chaotic system using a hybrid swarm intelligence algorithm

Juan A. Lazzús *, Marco Rivera, Carlos H. López-Caraballo

Departamento de Física, Universidad de La Serena, Casilla 554, La Serena, Chile

ARTICLE INFO

Article history:
Received 23 June 2015
Received in revised form 19 January 2016
Accepted 26 January 2016
Available online xxxx
Communicated by F. Porcelli

Keywords:
Chaotic systems
Parameter estimation
Particle swarm optimization
Ant colony optimization

ABSTRACT

A novel hybrid swarm intelligence algorithm for chaotic system parameter estimation is present. For this purpose, the parameters estimation on Lorenz systems is formulated as a multidimensional problem, and a hybrid approach based on particle swarm optimization with ant colony optimization (PSO–ACO) is implemented to solve this problem. Firstly, the performance of the proposed PSO–ACO algorithm is tested on a set of three representative benchmark functions, and the impact of the parameter settings on PSO–ACO efficiency is studied. Secondly, the parameter estimation is converted into an optimization problem on a three-dimensional Lorenz system. Numerical simulations on Lorenz model and comparisons with results obtained by other algorithms showed that PSO–ACO is a very powerful tool for parameter estimation with high accuracy and low deviations.

1. Introduction

Chaos theory is one of the most important achievements in the nonlinear system research field [1]. In recent years, nonlinear systems have drawn considerable attention to describe several phenomena related to both complex and dynamical systems [2]. In this context, partial differential equations have played an important role in the characterization of these phenomena, allowing to compare experiment with theory. Although information about the physical properties for many of these systems is available, not all dynamical parameters are usually known, and therefore they need to be estimated [3].

Parameter estimation for chaotic systems is an important issue in non-linear science (such as signal processing and control theory), which has attracted increasing interest in various research fields and which could be essentially formulated as a multidimensional optimization problem [4]. So far, different kinds of classical techniques have been developed to handle these problems [3]. Among them, the meta-heuristic based methods (such as the genetic algorithm, the particle swarm optimization algorithm, and the differential evolution algorithm) are some of the most popular methods used to formulate the parameter estimation problem as a multidimensional optimization problem [5].

The study of the parameter estimation problem has a long history, and it has been carried out with emphasis on the Lorenz chaotic system. This system is the first chaotic attractor in a three-dimensional autonomous system, and it was proposed by Lorenz in 1963 when he was studying atmospheric convection [6]. In the last years, different meta-heuristic algorithms have been proposed for parameter estimation on this system. In this way, Dai et al. [7] used a genetic algorithm (GA) to estimate parameters of Lorenz system (but one-dimensional parameter estimation was only taken into consideration). He et al. [4] proposed a particle swarm optimization (PSO) for estimating parameters of this system. Later, Gao et al. [8] used a similar PSO approach. In other approaches on Lorenz system, Gao et al. [9] used a novel quantum-behaved particle swarm optimization (NQPSO). Yang et al. [10] applied a quantum-behaved particle swarm optimization (QPSO). Sun et al. [11] proposed a variant of PSO called drift particle swarm optimization (DPSO). Modares et al. [12] applied an improved particle swarm optimization (IPSO). Alfi [3] introduced a novel adaptive particle swarm optimization (APSO) combining an adaptive mutation mechanism and a dynamic inertia weight into the conventional PSO algorithm. Alfi [13] proposed a novel particle swarm optimization namely dynamic inertia weight PSO (DIW-PSO). Using the most recently introduced swarm-based algorithms, Li et al. [14] and Peng et al. [15] used a newly biologically inspired search algorithm called chaotic ant swarm (CAS). Gao et al. [16] proposed a novel artificial bee colony algorithm (ABC), with an optimization technique based on the foraging behavior of honeybees. Li and Yin [5] combined the stochastic exploration of the cuckoo search and the exploitation capability of the orthogonal learning strategy (OLCS). Using evolutionary algorithm, Chan et al.

* Corresponding author.
E-mail address: jlazzus@dfuls.cl (J.A. Lazzús).

http://dx.doi.org/10.1016/j.physleta.2016.01.040
0375-9601/© 2016 Published by Elsevier B.V.

[17] applied an evolutionary programming (EP) approach. Peng et al. [18], and Banerjee and Abu-Mahfouz [19] used a differential evolution algorithm (DE). Wang and Li [20] used an effective hybrid quantum-inspired evolutionary algorithm with differential evolution (HQEDE). Li et al. [21] introduced a chaotic gravitational search algorithm (CGSA). Additionally, Palaniyandi and Lakshmanan [22] introduced a simple method to estimate the system parameters in continuous dynamical systems from their time series in another approaches. Huang et al. [23] proposed an adaptive controller with parameters identification based on Lyapunov stabilization theory. Zhao et al. [24] used an adaptive feed-back controlling method to identify uncertain parameters for this chaotic system.

In this work, parameters estimation for chaotic systems is formulated as a multidimensional optimization problem, and a hybrid swarm algorithm based on particle swarm–ant colony optimization (PSO–ACO) is implemented to solve the problem. To the best of the authors' knowledge, this is the first research on a PSO–ACO algorithm to estimate parameters of Lorenz chaotic system. Numerical simulations based on this system and comparisons with results obtained by other methods demonstrate the effectiveness, efficiency and robustness of this hybrid swarm algorithm.

2. Hybrid swarm algorithm

The proposed hybrid technique is developed combining particle swarm optimization (PSO) and ant colony optimization (ACO). Proposed by Kennedy et al. [25], PSO is one of the recent metaheuristic algorithms based on the behavior of a flock of birds or the sociological behavior of a group of people. Furthermore, ACO is an algorithm based on the foraging behavior of ants, and it was first introduced by Dorigo and Gambardella [26]. Here, hybrid PSO–ACO is based on the common characteristics of particle swarm optimization and ant colony optimization, like, survival as a swarm (colony) by coexistence and cooperation, individual contribution to food searching by a particle (ant) by sharing information locally and globally in the swarm (colony) between particles (ants) [27].

2.1. Description of PSO–ACO

The implementation of PSO–ACO algorithm consists of two stages. At the first stage, particle swarm optimization is applied, while ant colony optimization is applied at the second stage. Ant colony works as a local search, wherein, ants apply pheromone-guided mechanism to refine the positions found by particles at the particle swarm stage [28].

PSO–ACO is initialized by a population of random particles and the algorithm searches for optima by updating generations. In this system, each particle is moved through the multi-dimensional search space by adjusting its position in search space according to its own experience and the experience of neighboring particles. Therefore, the particle makes use of the best position encountered by itself and by that of its neighbors' to position itself towards an optimal solution [27]. The performance of each particle is evaluated using a pre-defined fitness function, which encapsulates the characteristics of the optimization problem [29].

The position S^i and the velocity V^i of the i-th particle in the n-dimensional search space can be represented as $S^i = (s_1^i, s_2^i, \cdots, s_n^i)^T$ and $V^i = (v_1^i, v_2^i, \cdots, v_n^i)^T$, respectively. Each particle has its own best position $P^i = (p_1^i, p_2^i, \cdots, p_n^i)^T$ corresponding to the personal best objective value obtained so far at generation t. And the global best particle is denoted by $P^g = (p_1^g, p_2^g, \cdots, p_n^g)^T$, which represents the best particle found so far at generation t in the entire swarm [25].

Then, the velocity of each particle is given by:

$$v_{t+1}^i = \omega_t v_t^i + c_1 r_1 (p_t^i - s_t^i) + c_2 r_2 (p_t^g - s_t^i) + c_3 r_3 (R_t^i - s_t^i) \quad (1)$$

where ω is the inertia weight, c_1 and c_2 are the acceleration constants, c_3 is the passive congregation coefficient, R_t^i is a particle selected randomly from the swarm, and r_1, r_2, and r_3 are the elements from three random sequences in the range $(0, 1)$. The current position of the particle is determined by s_t^i, and v_{t+1}^i is the new velocity at time $t+1$; p_t^i is the best one of the solutions that this particle has reached and p_t^g is the best one of the all solutions that the particles have reached [27].

The new particle position is computed by adding the velocity vector to the current position. Thus, the position of each particle at each generation is updated according to the following equation:

$$s_{t+1}^i = s_t^i + v_{t+1}^i \quad (2)$$

where s_{t+1}^i is the particle position at time instant t [25].

If $f(s^i)$ is the objective function (or fitness function) which will be minimized, then the best position p^i can be determined by:

$$p_{t+1}^i = \begin{cases} p_t^i & \text{if } f(s_{t+1}^i) \geq f(s_t^i) \\ s_{t+1}^i & \text{if } f(s_{t+1}^i) < f(p_t^i) \end{cases} \quad (3)$$

and P^g can be found by $P^g = p_{t+1}^g$ [29].

Variable ω in Eq. (1) is responsible for dynamically adjusting the velocity of the particles, which is responsible for balancing between local and global search. Hence this approach requires few iterations for reaching the convergence [30]. Note that, a low inertia weight value implies a local search while a high value leads to a global search [1]. Applying a large inertia weight at the start of the algorithm and making it decay to a small value through the particle swarm optimization execution makes the algorithm globally search at the beginning of the search, and locally search at the end of the execution [27,29]. The following weighting function is used for Eq. (1):

$$\omega = \omega_{max} - \frac{\omega_{max} - \omega_{min}}{t_{max}} t \quad (4)$$

where the subscripts *min* and *max* are the minimum and maximum values selected for such parameters. Generally, the value of each component in V^i can be clamped to the range $[-v_{max}^i, +v_{max}^i]$ to control the excessive roaming of particles outside the search space [30].

In PSO–ACO, a simple pheromone-guided mechanism of ant colony optimization is proposed to be applied as a local search [31]. The ant colony algorithm handles a number of ants equal to the number of particles in the swarm. Each ant i generates a solution ξ_t^i around p_t^g the global best-found position among all particles in the swarm up to iteration count t as:

$$\xi_t^i = \mathcal{N}(p_t^g, \eta) \quad (5)$$

where $\mathcal{N}(p_t^g, \eta)$ denotes a random number obtained by Gaussian function with mean value p_t^g and standard deviation η, where initially at $t = 1$ value of η and is updated at the end of each iteration as:

$$\eta = \eta \times d \quad (6)$$

where d is a parameter in range $0.25 \leq d < 1$. If $\eta < \eta_{min}$ then $\eta = \eta_{min}$, where η_{min} is a parameter in $(10^{-2}, 10^{-4})$. In the standard ant colony algorithms, the probability of selecting a path with more pheromones is greater than those of other paths. Similarly, the probability of selecting a solution in the neighborhood of p_t^g is greater than the others, in the Gaussian functions [32].

In this hybrid algorithm, the objective function value $f(\xi_t^i)$ is computed and the current position of ant ξ_t^i is replaced with the position s_t^i of the current position of the particle i in the swarm, only if $f(s_t^i) \geq f(\xi_t^i)$ and the current ant is in the feasible space