IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 8, 2020, accepted January 17, 2020, date of publication January 20, 2020, date of current version January 31, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2968106

Parameter Estimation of Lorenz Chaotic System Based on a Hybrid Jaya-Powell Algorithm

LI ZHUANG[1,2], LONGPENG CAO[3], YONG WU[4], YONGHONG ZHONG[4], LILI ZHANGZHONG[1,5], WENGANG ZHENG[1,5], AND LONG WANG[3], (Member, IEEE)

[1]National Research Center of Intelligent Equipment for Agriculture, Beijing 100097, China
[2]School of Data Science, City University of Hong Kong, Hong Kong
[3]School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China
[4]National Agricultural Technology Extension and Service Center, Beijing 100125, China
[5]Key Laboratory for Quality Testing of Hardware and Software Products on Agricultural Information, Ministry of Agriculture, Beijing 100097, China

Corresponding author: Wengang Zheng (zhengwg59@163.com)

This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFD0300201-3 and Grant 2017YFD0301004-4, in part by the Innovation Ability Construction Project of Beijing Academy of Agriculture and Forestry Sciences under Grant KJCX20180706, in part by the University of Science and Technology Beijing - the National Taipei University of Technology Joint Research Program under Grant TW2018008, in part by the Fundamental Research Funds for the Central Universities under Grant 06500078 and Grant 06500103, in part by the National Nature Science and Foundation of China under Grant 71801031, in part by the Beijing Natural Science Foundation under Grant 9204028, and in part by the National Key Research and Development Program of China under Grant 2018YFC0810600.

ABSTRACT Parameter estimation of Lorenz chaotic system using a novel hybrid Jaya-Powell algorithm is proposed in this paper. Since the nonlinear dynamic system is complex with multi-dimension parameters, estimating parameters of the system can be considered as a muti-objective optimization task. The proposed Jaya-Powell algorithm combines the Jaya and Powell algorithm to search for the relatively global optimum and local optimum respectively, which offers a more accurate and effective estimation. The searching strategy of the proposed algorithm facilitates the balance of the exploration and exploitation in the optimization procedure. Due to no algorithm-specific parameters are required in the Jaya and Powell algorithm, the proposed Jaya-Powell can avoid deliberate fine-tuning of corresponding parameters. To validate the accuracy and robustness of the proposed algorithm in parameter estimation, the simulation of Lorenz chaotic system and comparative experiments are conducted. Seven algorithms, including Jaya algorithm, Powell algorithm, Teaching-learning-based optimization (TLBO) algorithm, particle swarm optimization (PSO), genetic algorithm (GA), covariance matrix adaptation evolution strategy (CMA-ES), and cluster-chaotic-optimization algorithm (CCO), are considered as benchmarking algorithms in the comparison. The proposed hybrid Jaya-Powell algorithm outperforms seven benchmarking algorithms with the more accurate estimation and the relatively faster convergence. Based on the embedded system Raspberry pi 3, the proposed algorithm achieves the similar performance by comparing with the experiments conducted on the computer. The successful implementation via Raspberry pi 3 facilitates the application of the proposed algorithm in edge computing.

INDEX TERMS Chaos theory, Lorenz system, Parameter estimation, Jaya algorithm, Edge computing.

I. INTRODUCTION

As a prominent complex behavior in nonlinear dynamical systems, chaos has attracted much attention from researchers and been widely studied over the past three decades. The phenomenon of chaos has been applied in various fields, such as image encryption [1], secure communication [2], financial market [3], and etc. Lorenz chaotic system [4], which was firstly proposed by the meteorologist Edward Lorenz, is considered as one of the representative chaotic systems. The major characteristics of Lorenz chaotic system is of two-fold, the extreme sensitivity to initial conditions and unstable periodic orbits. To control and synchronize Lorenz chaotic system, it is critical to capture the information of parameters

The associate editor coordinating the review of this manuscript and approving it for publication was Mauro Gaggero.

due to the complexity of such system. However, determining parameters of Lorenz chaotic system is a challenging task in real applications. Thus, parameter estimation is a meaningful task in the analysis of Lorenz chaotic system.

A few pioneer studies related to parameter estimation of chaotic systems via the synchronization based methods have been reported in the literature. Parlitz *et al*. [5] developed a computer model to minimize the average synchronization error for parameter estimation of an unknown chaotic system. Maybhate and Amritkar [6] combined the synchronization and an adaptive control method to estimate parameters under a noisy environment. Hyun *et al*. [7] proposed an adaptive fuzzy observer for the synchronization of chaotic systems. Wang and Ge [8] discussed an adaptive backstepping method to synchronize uncertain chaotic systems, which was also reported by Yu and Zhang [9]. Yu *et al.* [10] introduced a backstepping synchronization method based on the equivalent transfer function for chaotic systems. Although the synchronization based methods are capable to estimate parameters of chaotic systems, their real implementations might be lack of feasibility and bounded to specific models.

To achieve the feasibility in parameter estimation of chaotic systems, evolutionary computation based methods have been applied in this field and address parameter estimation as an optimization problem. Numerous studies based on classical evolutionary computation algorithms have been developed to estimate the parameter of uncertain chaotic systems. Tao *et al*. [11] applied genetic algorithm (GA) into parameter estimation of chaotic time series. He *et al*. [12] developed particle swarm optimization (PSO) to estimate parameters of chaos systems and achieved more accurate solutions by comparing with GA. Tang and Guan [13] introduced PSO to address the issue of parameter estimation in a time-delay chaotic system. Sun *et al*. [14] proposed a drift particle swarm optimization (DPSO), which is an improved algorithm based PSO, to estimate parameters of chaotic systems. Peng *et al*. [15] considered parameter estimation of Lorenz chaotic system via the differential evolution algorithm (DE) and discussed the influence of the population size on the optimization performance. Xiang-Tao and Ming-Hao [16] proposed a Cuckoo search algorithm with the orthogonal learning to estimate parameters. Zhang *et al*. [17] identified the unknown parameters of the chaotic system based on teaching-learning-based optimization (TLBO). Reported evolutionary computation algorithms [11]–[17] proved the feasibility of parameter estimation for chaotic systems. However, the aforementioned evolutionary computation algorithms are susceptible to being trapped into local optima and suffering from the premature convergence.

Based on classical evolutionary computation algorithms, hybrid methods which combine two different evolutionary algorithms have been developed to obtain better estimation results. Wang and Xu [18] proposed a hybrid biogeography-based optimization(BBO) algorithm to recognize unknown parameters. Gu *et al*. [19] introduced a hybrid algorithm named as HABCDE to estimate model parameters, which combined the differential evolution algorithm (DE) and the artificial bee colony algorithm (ABC). Lazzús *et al*. [20] developed a hybrid PSO-ACO algorithm for parameter estimation of chaotic systems. Gálvez *et al.* [21] combined the characteristics of clustering and the randomness of chaotic sequence, named as cluster-chaotic-optimization, to solve the optimization problem in chaotic systems. The developed hybrid algorithms [18]–[20] have the ability to achieve the satisfactory estimation performance, which relies on fine-tuned algorithm-specific parameters.

This research proposes a hybrid algorithm named as Jaya-Powell algorithm, which combines Jaya algorithm [22] and Powell algorithm [23] to balance the exploration and exploitation in parameter estimation of Lorenz chaotic system. Firstly, Jaya algorithm, a population-based method proposed by Rao [22] free of algorithm-specific parameters, is applied to optimize parameters of Lorenz chaotic system. The main idea of Jaya algorithm is offering the probability for candidate solutions to move close to the best solution and away from the worst solution. Based on the update strategy, Jaya algorithm is capable to explore more unknown spaces to obtain useful information and avoid to fall into the local optimum. As an adaptive algorithm, Jaya only needs basic parameters including the population size and the number of maximum iterations. Next, Powell algorithm [23] is employed to exploit the local information based on the approximatively global optimum obtained by Jaya algorithm. Finally, the optimal solution of parameter estimation for Lorenz chaotic system can be attained. To validate the performance of the proposed Jaya-Powell algorithm, seven algorithms are considered as the benchmarking algorithms in the comparative experiments, including Jaya, Powell, TLBO, PSO, GA, CMA-ES, CCO. Based on simulations, the proposed algorithm outperforms seven benchmarking algorithms on the accuracy and robustness of parameter estimation for Lorenz chaotic system.

The remaining parts of this paper are organized as follows. In Section II, the problem formulation is introduced. Next, the proposed hybrid algorithm, Jaya-Powell algorithm, is described in Section III. The benchmarking algorithms employed in comparative experiments are illustrated in Section IV. Section V discussed simulation results based on different algorithms. Finally, Section VI makes the conclusion.

II. PROBLEM FORMULATION

The principle of unknown parameter estimation for chaotic systems is described in this section.

A. THE GENERIC CHAOTIC SYSTEM

A n-dimensional generic chaotic system is considered as following (1):

$$\dot{\boldsymbol{X}} = f(\boldsymbol{X}, \boldsymbol{X_0}, \boldsymbol{\varphi_0}) \tag{1}$$

where $\boldsymbol{X} = (x_1, x_2, \ldots, x_n)^T \in R^n$ represents the state vector of the original system, $\boldsymbol{X_0}$ denotes the initial state, and $\boldsymbol{\varphi_0} = (\varphi_1^0, \varphi_2^0, ..., \varphi_n^0)$ represents a set of original systematic parameters.

Assuming that the structure of the generic chaotic system (1) is known, then the corresponding estimated system can be described as (2):

$$\dot{\boldsymbol{Y}} = f(\boldsymbol{Y}, \boldsymbol{X_0}, \tilde{\boldsymbol{\varphi}}) \tag{2}$$

where $\boldsymbol{Y} = (y_1, y_2, \ldots, y_n)^T \in R^n$ denotes the state vector of the estimated system, $\tilde{\boldsymbol{\varphi}} = (\tilde{\varphi_1}, \tilde{\varphi_2}, \ldots, \tilde{\varphi_n})$ represents the set of estimated parameters.

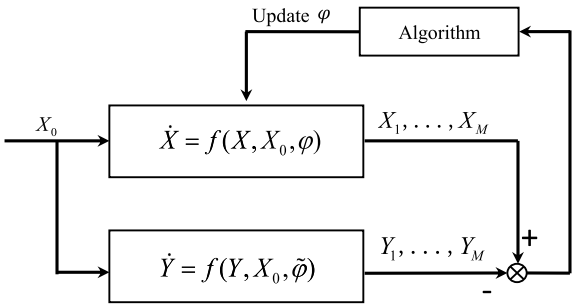

FIGURE 1. The optimization principle of parameter estimation for the generic chaotic system.

Estimating unknown parameters of the generic chaotic system can be formulated as an optimization problem, which targets on searching for a set of optimal parameters. The obtained optimal parameters can minimize the gap of behaviors between the original system and the estimated system. To attain the estimated parameters $\tilde{\varphi_1}, \ldots, \tilde{\varphi_n}$, minimizing the loss function (3) is required. The optimization principle of parameter estimation for the generic chaotic system is depicted in Fig. 1

$$min J(\tilde{\boldsymbol{\varphi}}) = \frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{X_k} - \boldsymbol{Y_k}\|^2 \tag{3}$$

where $\boldsymbol{X_k}$ $(k = 1, 2, \ldots, K)$ and $\boldsymbol{Y_k}$ $(k = 1, 2, \ldots, K)$ denote state vectors of the original system and the estimated system respectively, which are observed at time k. K is the total number of state vectors used in the estimation.

It is difficult to estimate parameters in the generic chaotic system due to its unstable dynamic orbits. Meanwhile, the behavior of the generic chaotic system is extremely sensitive to the initial state. Classical optimization algorithms are easy to fall into the local optimum, which cannot guarantees a satisfactory solution.

B. LORENZ SYSTEM

The Lorenz system [4] is a classical continuous-time system in chaos theory, which is described by a three-dimensional model for the fluid convection as illustrated in (4):

$$\begin{cases} \dot{x} = \sigma(y - x) \\ \dot{y} = x(\rho - z) - y \\ \dot{z} = xy - \beta z \end{cases} \tag{4}$$

where $\dot{x} = dx/dt, \dot{y} = dy/dt, \dot{z} = dz/dt$. $\boldsymbol{\varphi} = \{\sigma, \rho, \beta\}$ is the parameter vector and the $\boldsymbol{\varphi} = \{\sigma = 10, \rho = 28, \beta = 8/3\}$ is the original parameter vector of Lorenz system.

III. JAYA-POWELL ALGORITHM

In this section, a novel hybrid algorithm named as Jaya-Powell is proposed. The Jaya-Powell algorithm combines the adaptive Jaya algorithm and Powell algorithm to balance the exploration and exploitation, which offers the high probability to search for the global optimum. In the optimization procedure, Jaya algorithm [22] is firstly applied to find a relatively good solution and narrow down the searching space for further searching. Next, Powell algorithm [23] is employed to search for the best local solution based on the reduced searching space obtained from Jaya algorithm. The rapid convergence of Jaya algorithm and Powell algorithm ensures the searching speed during the optimization. The main optimization procedure of the proposed Jaya-Powell algorithm is described in **Algorithm 1**.

Algorithm 1 Jaya-Powell Algorithm

Input: The population size: N, Iteration: M, Tolerance: e, Model parameters of the ith population at the jth iteration: $O_{i,j}$
Output: Solution: O^*_{best}
for $i := 1$ *to* N **do**
  Initialize $O_{i,1}$;
end
Set $j = 1$;
repeat
  Update solutions $O_{i,j}$ based on Jaya algorithm;
  Get the best solution O_{best};
  Set $e = J(O_{best})$;
  Set $m = m + 1$.
until *termination criterion e or j(j <= M) satisfied*;
Update O_{best} via Powll algorithm to O^*_{best};
Return O^*_{best}.

A. JAYA ALGORITHM

Jaya algorithm is an adaptive optimization algorithm, which facilitates candidate solutions to get closer to the current optimum and move away from the worst solution over iterations. The major characteristics of Jaya algorithm is that it is free of algorithm-specific parameters, such as the mutation probability and crossover probability of GA as well as social and cognitive parameters of PSO. The algorithm-specific parameters can greatly influence the optimization performance and need to be well fine-tuned. In Jaya algorithm, only the population size and the number of iterations need to be set in advance.

The update strategy of Jaya algorithm is illustrated as (5):

$$O'_{m,p,n} = O_{m,p,n} + r_{1,p,n}(O_{m,best,n} - |O_{m,p,n}|) - r_{2,p,n}(O_{m,worst,n} - |O_{m,p,n}|) \quad (5)$$

where $O_{m,p,n}$ is the value of the mth variable for the pth candidate at nth iteration. The $O_{m,best,n}$ and $O_{m,worst,n}$ are the values of the best and worst candidates in the searching space for the mth variable at the nth iteration. $r_{1,m,n}$ and $r_{2,m,n}$ are two random numbers for the mth variable at the nth iteration in the range [0, 1]. $O'_{m,p,n}$ is the updated solution. Based on the objective function J, the updated solution will be accepted and passed to the next iteration based on the rule (6).

$$O'_{m,n} = \begin{cases} O'_{m,n} & if\ J(O'_{m,n}) \le J(O_{m,n}) \\ O_{m,n} & if\ J(O'_{m,n}) > J(O_{m,n}) \end{cases} \quad (6)$$

The optimization procedure of Jaya algorithm is summarized as follows:

Step 1: Set the population size, the range of parameters, and the maximum number of iterations.
Step 2: Initialize the population randomly with the specific range of parameters.
Step 3: Identify the best and worst solutions in the current solution space.
Step 4: Update the current solution based on the best and worst solutions according to eq. (5).
Step 5: Evaluate the value of the objective function for each updated solution. Then the updated solution will be accepted or abandoned according to eq. (6).
Step 6: If the maximum iteration is satisfied, the optimization procedure will be stopped. Otherwise, the procedure jumps to Step 3.

B. POWELL METHOD

Powell algorithm, targeting on searching for the optimum in multi-dimension solution space, is considered as a local optimization algorithm without algorithm-specific parameters. It is a searching method employing the conjugate direction to speed up the convergence rate during the optimization. Meanwhile, Powell algorithm utilizes a bi-dimensional searching strategy instead of a hopping probe step to minimize the objective funtion, which can be applied to the non-differentiable optimization problem. Let $\boldsymbol{X_0}$ be the initial state vector and the objective function is denoted by f. Initialize a set of directions u_i as the standard base vectors, $u_i = e_i$, $i = 1, 2, .., n$. The procedure of Powell method is summarized as follows:

Step 1: Set the start point $P_0 = X_i$ and tolerance error e.
Step 2: Find the value of $\gamma = \gamma_i$ to minimize $f(P_{i-1} + \gamma_i \times u_i)$ and set $P_i = P_{i-1} + \gamma_i \times u_i$ for $i = 1, 2, .., n$,.
Step 3: Set $u_{i+1} \leftarrow u_i$ for $i = 1, 2, .., n$.
Step 4: Set $u_n = P_n - P_0$.
Step 5: Set $i = i + 1$.
Step 6: Find the value of $\gamma = \gamma_{min}$ to minimize $f(P_0 + \gamma_{min} * u_n)$ and set $X_i = P_0 + \gamma_{min} \times u_n$.
Step 7: Repeat Step 1 to Step 5 until e is satisfied.

The initial vectors of Powell algorithm are replaced by a set of new conjugate vectors completely after n iterations. During the iterations, $u_n = P_n - P_0$ become linearly dependent at each stage and might contain incorrect information. The enhanced Powell method, which replaces one of the direction vectors selectively based on an obtained new direction vector, is proposed to guarantee better conjugate properties of the new group of vectors. The procedure of Powell method is illustrated in **Algorithm 2**.

Algorithm 2 Powell Algorithm

Input: Initial solution: X_0; Tolerance error: e; Objective function: f;
Output: Solution: X_{best};
repeat
  Set $P_0 = X_0$;
  for $k := 1$ *to* n **do**
    for $i := 1$ *to* n **do**
      Initialize the set of direction $u_i = e_i$;
      Find a minimizer γ_i;
      Calculate $f(P_{i-1} + \gamma_i \times u_i)$;
      Set $P_i = P_{i-1} + \gamma_i \times u_i$;
    end
    Define $\Delta f_k = f(P_k) - f(P_{k-1})$;
    Calculate $\Delta f = |\Delta f_r| = max\{|\Delta f_k|\}$;
    Record the subscript r;
    Set the u_r as the maximum decrease;
    Calculate $f_k = f(P_k), f_E = f(2P_n - P_k)$;
    if *either*
    $f_E > f_0$ *or* $2(f_0 - 2f_n - f_E) \times (f_0 - f_n - \Delta f)^2 \ge 0.5 \times \Delta f (f_0 - f_E)^2$ **then**
      Set $u = P_n - P_k$;
      Find the minimizer γ_{min};
      Calculate $f(P_0 + \gamma_{min} \times u)$;
      Update the search direction:
      $[u_1, \ldots, u_{r-1}, u_{r+1}, \ldots, u_n, u]$;
      Set $X_{k+1} = P_n + \gamma_{min} \times u$;
      Initialize $P_0 = X_{k+1}$;
      Set $i = 1, k = k + 1$.
    end
    else
      Keep the search direction: $[u_1, \ldots\ldots, u_n]$;
      Set $X_{k+1} = P_n$ or $X_{k+1} = 2P_n - P_k$;
      Initialize $P_0 = X_{k+1}$;
      Set $i = 1, k = k + 1$.
    end
  end
until *termination criterion e satisfied*;

IV. BENCHMARKING ALGORITHMS

To verify the performance of the proposed Jaya-Powell algorithm on parameter estimation for Lorenz chaotic system, seven benchmarking algorithms, including Jaya algorithm, Powell algorithm, teaching-learning-based optimization

algorithm (TLBO), particle swarm optimization (PSO), generic algorithm (GA), covariance matrix adaptation evolution strategy (CMA-ES), cluster-chaotic-optimization algorithm (CCO), are employed in comparative experiments. Brief introductions of seven benchmarking algorithms are described in this section.

A. PSO

The major characteristics of PSO is to find the optima through the collaboration and information sharing among individuals. A group of particles in PSO move around the searching space based on the historical information of their own best locations and the best location of the whole swarm. The velocity of a particle is updated after one iteration via (7):

$$v_i(t+1) = \omega \times v_i(t) + c_1 \times r_1 \times (p_{best} - x_i(t)) + c_2 \times r_2 \times (g_{best} - x_i(t)) \quad (7)$$

where $v_i(t)$ is the velocity of the ith particle at time t. c_1 and c_2 are acceleration coefficients of the personal best p_{best} and the global best position g_{best} respectively. ω denotes the scaling factor of the particle velocity. $x_i(t)$ is the current position of the ith particle at time t. The position of a particle will be updated at time $t+1$ based on (8):

$$x_i(t+1) = x_i(t) + v_i(t) \quad (8)$$

B. TLBO

TLBO algorithm consists of two components, the teacher phase and the learner phase. As a global optimization algorithm, the best solution denotes the teacher and the remaining solutions are the learners at each iteration. TBLO algorithm does not require other algorithm-specific parameters except the population size and the number of iterations.

1) TEACHER PHASE

The parameter of a student will be updated according to the difference between the average performance and the best performance via (9) and (10)

$$X'_{j,k,i} = X_{j,k,i} + r_i \times (X_{j,kbest,i} - T_F \times M_{j,i}) \quad (9)$$

$$T_F = round[1 + rand(0,1)\{2-1\}] \quad (10)$$

where $M_{j,i}$ is the mean value of jth subject after ith iterations. r_i is a random number within the range of [0,1]. $X_{j,kbest,i}$ is the best learner of jth subject. T_F is the teaching factor and the $X'_{j,k,i}$ is the updated solution.

2) LEARNER PHASE

Next, two students are selected randomly among the group of students to learn from the teacher, which offers the probability of enhancing the performance of students.

C. GA

GA is inspired by the law of evolution in nature. The optimization procedure of GA is summarized as follows:

Step 1: Initialize the number and the length of chromosomes
Step 2: Calculate the fitness value of individuals and select two individuals named as parents for next iteration.
Step 3: Recombine two parents to generate children for the next iteration (crossover).
Step 4: Mutate parents from children randomly.
Step 5: Calculate the fitness value of each individual in the population and record the best solution.
Step 6: Repeat Step 2 to Step 5 until the result is satisfied.

D. CMA-ES

CMA-ES is considered as a state-of-the-art evolutionary strategy for the derivative-free global optimization. Firstly, individuals are initialized randomly and sampled according to the multivariate normal distribution. Next, selected individuals are considered as the parents and re-ordered based on the fitness function. Then the re-ordered individuals are updated via the covariance matrix of the current distribution. The iteration will be stopped once the optimum has been found.

E. CCO

The optimization strategy in CCO combines the characteristics of the clustering and the randomness of chaotic sequences. Based on CCO, the population is firstly divided into different clusters in each generation. Next, individuals in each cluster are updated via two operators named as intra-cluster and extra-cluster. Chaotic sequence are applied in such two operators to generate random numbers for searching better solutions. The procedure will be repeated to find the best solution.

V. SIMULATION AND COMPARITIVE EXPERIMENTS

A. EXPERIMENT ENVIRONMENT

Simulation experiments are conducted on a computer having a single i5-6500 CPU and 4G memory. Meanwhile, the performance of the proposed algorithm has also been validated on Raspberry pi 3, which has the advantages of small size and portability. As the embedded device, Raspberry pi 3 is powerful for solving the ordinary problem and can be used in many conditions such as non-personal computer. It is a single-board computer developed by Raspberry Pi foundation which focuses on promoting the basic of technology on the computer. Based on Raspberry pi 3, the proposed algorithm is capable to be applied in edge computing.

B. SIMULATION

To demonstrate the superiority of the proposed algorithm, the original Lorenz system evolves freely from a random initial state firstly. A state vector $\boldsymbol{X_0} = (0., 1., 1.05)$ is set as the starting point to conduct the behavior. 300 successive state vectors, which are defined as $\boldsymbol{X_1}, \boldsymbol{X_2}, \ldots, \boldsymbol{X_{300}}$ and $\boldsymbol{Y_1}, \boldsymbol{Y_2}, \ldots, \boldsymbol{Y_{300}}$, are selected both from original and estimated systems to calculate the fitness value. The step size of chaotic behaviors t is set to 0.01. The value of the objective

TABLE 1. Statistical results of parameter estimation for Lorenz system based on different algorithms.

		Algorithm							
		Jaya-Powell	Jaya	Powell	TLBO	PSO	GA	CMA-ES	CCO
Best	σ	**10.000000**	9.801253	10.00000	9.833314	9.957756	9.626279	9.999609	9.543224
	$\frac{\lvert\sigma-10\rvert}{10}$	1.119104e-23	0.198747	2.913225e-13	0.166686	0.042244	0.373721	0.000391	0.456776
	ρ	**28.000000**	28.239758	28.000000	28.100972	28.038217	27.961971	28.000178	27.888521
	$\frac{\lvert\rho-28\rvert}{28}$	2.486899e-14	0.239758	4.583001e-13	0.100972	0.038217	0.038029	0.000178	0.111478
	β	**2.666667**	2.692626	2.666667	2.670115	2.669889	2.626392	2.666665	2.603154
	$\frac{\lvert\beta-8/3\rvert}{8/3}$	3.552713e-15	0.025959	5.728751e-14	0.003448	0.003223	0.040275	2.090676e-06	0.063512
	J	**1.291529e-25**	0.299256	6.204374e-25	0.183798	0.026173	2.524589	1.275172e-06	7.966531
Mean	σ	**10.000000**	10.250868	9.847906	9.956764	10.083554	9.842539	9.620862	9.660097
	$\frac{\lvert\sigma-10\rvert}{10}$	1.776356e-15	0.250868	0.152094	0.043236	0.083554	0.157460	0.379137	0.339903
	ρ	**28.00000**	27.473784	26.002084	27.984726	27.9451205	27.368433	24.150269	25.612109
	$\frac{\lvert\rho-28\rvert}{28}$	2.735589e-13	0.526216	1.997916	0.015274	0.054879	0.631567	3.849731	2.387890
	β	**2.666667**	2.598840	2.598070	2.662245	2.665272	2.546570	2.578566	2.532006
	$\frac{\lvert\beta-8/3\rvert}{8/3}$	4.884981e-14	0.067827	0.068596	0.004421	0.001395	0.120096	0.088101	0.134660
	J	**7.819819e-24**	3.358928	116.937687	2.416112	0.798651	44.381122	158.231152	343.656833
Worst	σ	**10.000000**	10.482337	9.458302	9.711571	10.625504	9.605912	9.000000	9.636259
	$\frac{\lvert\sigma-10\rvert}{10}$	2.842171e-12	0.482337	0.541698	0.288429	0.625504	0.394088	0.999999	0.363740
	ρ	**28.00000**	26.222595	27.048644	29.327925	27.135383	24.711447	21.868929	27.059840
	$\frac{\lvert\rho-28\rvert}{28}$	3.367973e-12	1.777405	0.951356	1.327925	0.864616	3.288553	6.131071	0.940159
	β	**2.666667**	2.410404	2.000000	2.891058	2.572937	2.058641	2.449754	2.503962
	$\frac{\lvert\beta-8/3\rvert}{8/3}$	3.503864e-13	0.256263	0.666667	0.224391	0.093730	0.608025	0.216913	0.162705
	J	**3.750901e-23**	15.501535	356.254221	10.406942	2.314511	167.399045	228.585069	453.739011

function J is calculated using the function (11):

$$J = \sum_{k=1}^{300} (\boldsymbol{X_k} - \boldsymbol{Y_k})^2 \tag{11}$$

Parameter settings of the proposed algorithm and seven benchmarking algorithms are described as follows. To ensure the same restrictive conditions for different algorithms, the maximum number of iterations, the relative error of the stopping criterion in the convergence, and the population size are set as 100, $e = 10^{-4}$, 100 respectively. The searching ranges of three parameters are set as $9.0 \leq \sigma \leq 11.0$, $20.0 \leq \rho \leq 30.0$, $2.0 \leq \beta \leq 3.0$ separately. Based on aforementioned parameters, there is no other algorithm-specific parameters to set for Jaya-Powell algorithm, Jaya algorithm, Powell algorithm, TLBO, and CCO. The algorithm-specific parameters of PSO are set as $c_1 = c_2 = 2.05$ and $\omega = 0.729$ [24]. In GA, the length of the chromosome L, the crossover rate P_r, and the mutation rate P_m are set as 3, 0.8, and 0.1 [12]. The $\sigma = 0.5$ is set in CMA-ES, which is initialized randomly. In addition, each algorithm are conducted via 20 simulation experiments under the same condition to obtain more universal results.

C. RESULTS AND DISCUSSION

To validate the optimization performance of the proposed Jaya-Powell algorithm, two types of comparative experiments are employed. In the first type of comparison, all algorithms are conducted via a single experiments to roughly illustrate the corresponding convergence. As shown in Fig. 2,

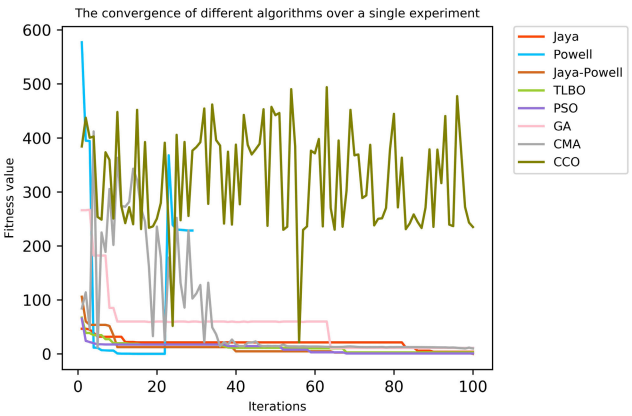

FIGURE 2. The convergence of different algorithms over a single experiment.

algorithms except Powell and CCO are capable to converge to a relatively satisfactory solution. Meanwhile, the proposed Jaya-Powell algorithm reaches the highest degree of accuracy among all algorithms, which is around e^{-20}.

In the second type of comparison, all algorithms are conducted to for parameter estimation of σ, ρ, and β over 20 independent experiments separately. The repeated experiments make it easier to spot anomalies and offer the more reliable performance of different algorithms. The optimization performance via the proposed Jaya-Powell algorithm and seven benchmarking algorithms, including the best, average, and worst performance over 20 repeated experiments, are summarized in Table 1. According to Table 1, it is clear

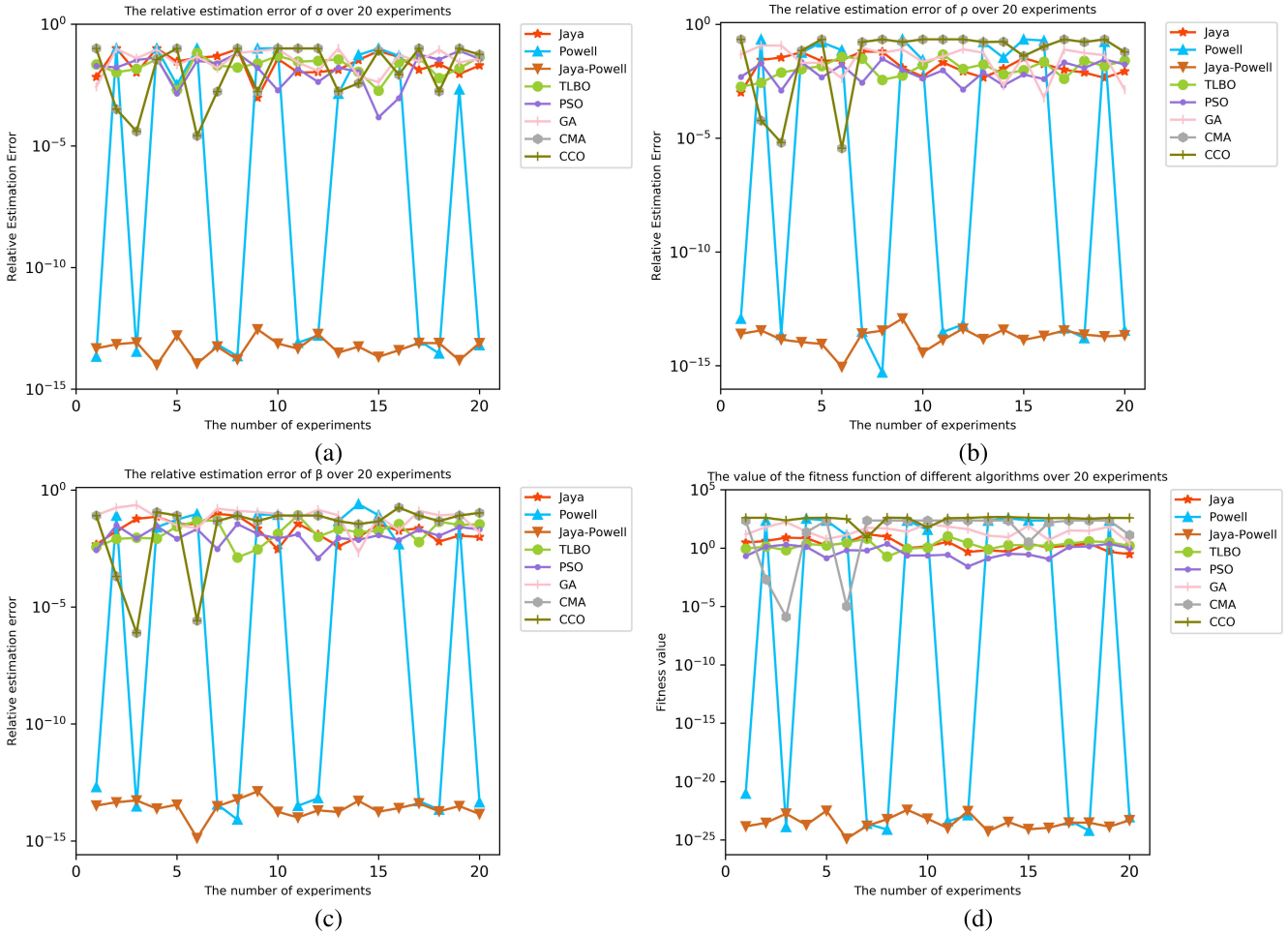

FIGURE 3. The relative estimation error of parameters and the value of the fitness function for different algorithms over 20 experiments: (a) σ; (b) ρ; (c) β; (d) the value of the fitness function.

that the proposed Jaya-Powell algorithm outperforms seven benchmarking algorithms on the best, average, and worst performance. The average performance of Jaya-Powell algorithm demonstrates its robustness of parameter estimation for Lorenz chaotic system, which also validates its independence of the random initialization. However, other benchmarking algorithms, such as Powell, GA, CMA-ES and CCO, achieves a poor level of robustness due to the overdependence on well-selected initializations. If the initialization of Lorenz chaotic system is away from the original solution or close to local optima, it is challenging for these benchmarking algorithms to search for satisfactory solutions. Fig. 3 describes the relative estimation error of parameters and the value of fitness function for different algorithms over 20 experiments. As depicted in Fig. 3, the strong randomness of initializations results in much negative effects on Powell algorithm while less impacts on Jaya algorithm. Thus, if a relatively better solution is obtained by Jaya algorithm, Powell algorithm can exploit the nearby space based on the obtained solution to achieve a more accurate solution.

TABLE 2. The overall performance of different algorithms.

Algorithm	RMSE	Time/s
Jaya-Powell	1.373699e-23	24.102933
Jaya	5.042903	20.597662
Powell	173.914969	1.358679
TLBO	3.324818	44.824379
PSO	1.084458	24.144498
GA	60.553169	12.954163
CMA-ES	187.540648	2.593789
CCO	361.918960	189.839586

To illustrate the overall performance of the proposed algorithm and seven benchmarking algorithms, the root-mean-square error (RMSE) of the fitness function and average running time are utilized for the 20 repeated experiments. Based on Table 2, the proposed algorithm attains the lowest RMSE to demonstrate its robustness on parameter estimation. There is no significant difference of the average running time among different algorithm except CCO. Thus, the proposed

Jaya-Powell algorithm achieves the best overall performance on parameter estimation of Lorenz chaotic system.

VI. CONCLUSION

In this paper, a novel hybrid algorithm combining Jaya algorithm and the Powell method named Jaya-Powell was proposed for parameter estimation of Lorenz chaotic system. The proposed hybrid algorithm focused on efficiently searching for a satisfactory solution via balancing the exploration and exploitation, which avoided to fall into the local optimum. The two types of comparative experiments demonstrated the superiority of the proposed algorithm on parameter estimation of Lorenz chaotic system by comparing with seven benchmarking algorithms. Due to the characteristics of non-specific-parameters, it took less time for Jaya-Powell algorithm to converge to the optimum based on the guarantee of the estimation accuracy.

The numerical experiments was also conducted on the embedded environment. We conducted the parameter estimation via the proposed Jaya-Powell method on the Raspberry Pi 3. It demonstrated the feasibility of parameter estimation of Lorenz chaotic system on the embedded platform, which can be applied into edge computing. Both experiments on computer and Raspberry Pi 3 have validated the outperformance of the Jaya-Powell algorithm.

The future work will extend the parameter estimation of chaotic systems through the GPU-based implementation to achieve the high-speed performance.

REFERENCES

[1] F. Özkaynak, "Brief review on application of nonlinear dynamics in image encryption," *Nonlinear Dyn.*, vol. 92, no. 2, pp. 305–313, Apr. 2018.

[2] B. Vaseghi, M. A. Pourmina, and S. Mobayen, "Secure communication in wireless sensor networks based on chaos synchronization using adaptive sliding mode control," *Nonlinear Dyn.*, vol. 89, no. 3, pp. 1689–1704, Aug. 2017.

[3] S. Lahmiri and S. Bekiros, "Chaos, randomness and multi-fractality in Bitcoin market," *Chaos, Solitons Fractals*, vol. 106, pp. 28–34, Jan. 2018.

[4] E. N. Lorenz, "Deterministic nonperiodic flow," *J. Atmos. Sci.*, vol. 20, no. 2, pp. 130–141, 1963.

[5] U. Parlitz, L. Junge, and L. Kocarev, "Synchronization-based parameter estimation from time series," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 54, no. 6, pp. 6253–6259, Jul. 2002.

[6] A. Maybhate and R. E. Amritkar, "Use of synchronization and adaptive control in parameter estimation from a time series," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 59, no. 1, pp. 284–293, Jul. 2002.

[7] C.-H. Hyun, J.-H. Kim, E. Kim, and M. Park, "Adaptive fuzzy observer based synchronization design and secure communications of chaotic systems," *Chaos, Solitons Fractals*, vol. 27, no. 4, pp. 930–940, Feb. 2006.

[8] C. Wang and S. Ge, "Adaptive synchronization of uncertain chaotic systems via backstepping design," *Chaos, Solitons Fractals*, vol. 12, no. 7, pp. 1199–1206, Jun. 2001.

[9] Y. Yu and S. Zhang, "Adaptive backstepping synchronization of uncertain chaotic system," *Chaos, Solitons Fractals*, vol. 21, no. 3, pp. 643–649, Jul. 2004.

[10] J. Yu, J. Lei, and L. Wang, "Backstepping synchronization of chaotic system based on equivalent transfer function method," *Optik*, vol. 130, pp. 900–913, Feb. 2017.

[11] C. Tao, Y. Zhang, and J. J. Jiang, "Estimating system parameters from chaotic time series with synchronization optimized by a genetic algorithm," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 76, no. 1, 2007, Art. no. 016209.

[12] Q. He, L. Wang, and B. Liu, "Parameter estimation for chaotic systems by particle swarm optimization," *Chaos, Solitons Fractals*, vol. 34, no. 2, pp. 654–661, Oct. 2007.

[13] Y. Tang and X. Guan, "Parameter estimation for time-delay chaotic system by particle swarm optimization," *Chaos, Solitons Fractals*, vol. 40, no. 3, pp. 1391–1398, May 2009.

[14] J. Sun, J. Zhao, X. Wu, W. Fang, Y. Cai, and W. Xu, "Parameter estimation for chaotic systems with a drift particle swarm optimization method," *Phys. Lett. A*, vol. 374, no. 28, pp. 2816–2822, Jun. 2010.

[15] B. Peng, B. Liu, F.-Y. Zhang, and L. Wang, "Differential evolution algorithm-based parameter estimation for chaotic systems," *Chaos, Solitons Fractals*, vol. 39, no. 5, pp. 2110–2118, Mar. 2009.

[16] X.-T. Li and M.-H. Yin, "Parameter estimation for chaotic systems using the cuckoo search algorithm with an orthogonal learning method," *Chin. Phys. B*, vol. 21, no. 5, May 2012, Art. no. 050507.

[17] H. Zhang, B. Li, J. Zhang, Y. Qin, X. Feng, and B. Liu, "Parameter estimation of nonlinear chaotic system by improved TLBO strategy," *Soft Comput.*, vol. 20, no. 12, pp. 4965–4980, Dec. 2016.

[18] L. Wang and Y. Xu, "An effective hybrid biogeography-based optimization algorithm for parameter estimation of chaotic systems," *Expert Syst. Appl.*, vol. 38, no. 12, pp. 15103–15109, Nov. 2011.

[19] W. Gu, Y. Yu, and W. Hu, "Parameter estimation of unknown fractional-order memristor-based chaotic systems by a hybrid artificial bee colony algorithm combined with differential evolution," *Nonlinear Dyn.*, vol. 84, no. 2, pp. 779–795, Apr. 2016.

[20] J. A. Lazzús,, M. Rivera, and C. H. López-Caraballo, "Parameter estimation of Lorenz chaotic system using a hybrid swarm intelligence algorithm," *Phys. Lett. A*, vol. 380, nos. 11–12, pp. 1164–1171, Mar. 2016.

[21] J. Gálvez, E. Cuevas, H. Becerra, and O. Avalos, "A hybrid optimization approach based on clustering and chaotic sequences," *Int. J. Mach. Learn. Cybern.*, vol. 11, no. 2, pp. 359–401, Feb. 2020.

[22] R. Rao, "Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems," *Int. J. Ind. Eng. Comput.*, vol. 7, no. 1, pp. 19–34, 2016.

[23] M. J. D. Powell, "An efficient method for finding the minimum of a function of several variables without calculating derivatives," *Comput. J.*, vol. 7, no. 2, pp. 155–162, Feb. 1964.

[24] M. N. Alam, B. Das, and V. Pant, "A comparative study of metaheuristic optimization approaches for directional overcurrent relays coordination," *Electr. Power Syst. Res.*, vol. 128, pp. 39–52, Nov. 2015.

LI ZHUANG received the B.Eng. degree in measurement and control technology and instrument from the Huazhong University of Science and Technology, Wuhan, China, in 2016. She is currently pursuing the Ph.D. degree with the School of Data Science, City University of Hong Kong. Her research interests focus on computer vision, machine learning, computational intelligence, and related applications in intelligent transportation, renewable energy, and the industry.

LONGPENG CAO received the B.S. degree in computer science and technology from the University of Science and Technology Beijing, in 2017, where he is currently pursuing the M.Phil. degree in computer science and technology.

YONG WU received the master's degree from Southwest University, China, in 2003. He is currently a Senior Agronomist with the National Agricultural Technology Extension and Service Center, Beijing, China. His research interests focus on the soil water engineering domain.

YONGHONG ZHONG received the master's degree from Northwest Agricultural and Forestry University, China. She is currently a Senior Agronomist with the National Agricultural Technology Extension and Service Center, Beijing, China. Her research interest focuses on the soil water engineering domain.

LILI ZHANGZHONG received the Ph.D. degree in agricultural water-soil engineering from China Agricultural University, China, in 2016. She is currently a Senior Engineer with the National Research Center of Intelligent Equipment for Agriculture, Beijing, China.

WENGANG ZHENG received the Ph.D. degree in computational mechanics from the Dalian University of Technology, China, in 2002. He is currently a Professor with the National Research Center of Intelligent Equipment for Agriculture, Beijing, China. His research interest focuses on the agricultural engineering domain.

LONG WANG (Member, IEEE) received the M.Sc. degree (Hons.) in computer science from University College London, London, U.K. in 2014, and the Ph.D. degree in systems engineering and engineering management from the City University of Hong Kong, Hong Kong, in 2017.

He is currently an Associate Professor with the Department of Computer Science and Technology, University of Science and Technology Beijing, Beijing, China. His research interests include machine learning, computational intelligence, computer vision, and their industrial applications. He was a recipient of the Hong Kong Ph.D. Fellowship, in 2014. He serves as an Associate Editor of IEEE Access and *Canadian Journal of Electrical and Computer Engineering*, as well as an Academic Editor of *PLOS ONE*. He is also a Lead Guest Editor of data science-related special issues on *Intelligent Automation & Soft Computing* and *Water*.