Research Article

Multiobjective Lightning Flash Algorithm Design and Its Convergence Analysis via Martingale Theory

Jiandong Duan, 1 Jing Wang, 1 Xinghua Liu, 1 and Gaoxi Xiao 2

1 School of Electrical Engineering, Xi’an University of Technology, Xi’an 710048, China
2 School of Electrical and Electronic Engineering, Nanyang Technological University, Nanyang, Singapore

Correspondence should be addressed to Xinghua Liu; liuxh@xaut.edu.cn

Received 10 May 2020; Revised 9 August 2020; Accepted 21 September 2020; Published 9 October 2020

Academic Editor: Danilo Comminiello

Copyright © 2020 Jiandong Duan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, a novel multiobjective lightning flash algorithm (MOLFA) is proposed to solve the multiobjective optimization problem. The charge population state of the lightning flash algorithm is defined, and we prove that the charge population state sequence is a Markov chain. Since the convergence analysis of MOLFA is to investigate whether a Pareto optimal solution can be reached when the optimal charge population state is obtained, the development of a charge population state is analyzed to achieve the goal of this paper. Based on the martingale theory, the MOLFA convergence analysis is carried out in terms of the supermartingale convergence theorem, which shows that the MOLFA can reach the global optimum with probability one. Finally, the effectiveness of the proposed MOLFA is verified by a numerical simulation example.

1. Introduction

Lightning flash algorithm (LFA), a new heuristic algorithm proposed by Kheshti [1], has been successfully applied to large-scale nonconvex economic dispatch [2], nonconvex combined emission economic dispatch [3], protective relay coordination [4], and intelligent inertial control of wind turbines for grid frequency control [5, 6]. As demonstrated in [3], LFA not only searches the global optimal solution more quickly but also has better reliability and effectiveness, compared with other intelligent algorithms such as particle swarm optimization (PSO) [7], lightning search algorithm (LSA) [8], firefly algorithm (FA) [9], and harmony search algorithm (HSA) [10]. In a word, the lightning flash algorithm can synthesize the outstanding computational characteristics of both swarm intelligence algorithm and evolutionary algorithm. Since most of problems need to optimize two or more objectives simultaneously, the multiobjective optimization algorithm is increasingly prevalent, such as multiple objective particle swarm optimization (MOPSO) [11] and nondominated sorting genetic algorithm (NSGA-II) [12]. However, a comprehensive theoretical study on LFA or multiple objective LFA is largely still in vain. The missing theoretical results include a global convergence analysis which is of high importance for further studies and improvements.

The convergence analysis of an intelligent algorithm is typically to investigate whether the algorithm can converge to a global optimal value. For example, the artificial immune algorithm is proposed based on the principle of the artificial immune system, however with a relatively low computational efficiency and slow convergence [13]. To overcome such drawbacks, a novel hybrid immune algorithm is proposed in [14], with its convergence under weaker conditions being proven using a quasidescent method. Jin et al. [15] analyze and deduce the stability of mean-square convergence of the PSO algorithm based on the theory of stochastic processes and give a sufficient condition for the system to be mean-square stable. For multiobjective optimization methods, a simple theoretical proof of the convergence of the general Pareto-based bacteria colony chemotaxis algorithm is presented in [16].

Markov chain is a class of important stochastic processes with nonaftereffect property [17–20]. As it is proved that the
optimization processes of some classical intelligent algorithms also have this property [21], Markov chain has been widely applied in convergence analysis of many intelligent algorithms [22–28], e.g., chicken swarm optimization, ant colony algorithm, and genetic algorithm (GA). In [22], the finite homogeneous Markov chain model of chicken swarm optimization is established with some of its properties being analyzed. It is proved that the algorithm satisfies two convergence criteria and guarantees the global convergence. In [23], Huang et al. analyze the convergence time of the ant colony algorithm based on the absorbing Markov chain model. Markov models of genetic algorithm and artificial bee colony algorithm are established, respectively, in [24, 25]. The Markov chain model and a dynamic matrix model of the bat algorithm are proposed in [26]. It is shown that the transition matrix of the wolf pack algorithm can converge to a limit value with probability one, which demonstrates its global convergence property [27]. The state transition probability of particles is discussed in [28]. Martingale theory is the basis of modern probability theory and stochastic process and also has important applications in financial pricing decision-making and control model [29]. To avoid complex calculations for solving eigenvalue of the transfer matrix, the state sequence of particles with the best fitness value is transformed to a supermartingale, and the convergence of standard PSO and improved algorithm QPSO is analyzed in [30]. The analysis for the convergence properties of the ant colony algorithm is conducted by using Markov chains and martingale theory. The pheromone trail vector almost always converges to the optimal solution when the iteration time is infinite [31]. Despite of all these efforts, however, the convergence analysis of LFA and MOLFA has not been fully studied.

To shorten such a gap, this study proposes a multiobjective lightning flash algorithm and analyzes the global convergence of LFA. Major contributions are summarized as follows:

1. Based on the repository mechanism, a novel multiobjective lightning flash algorithm (MOLFA) is proposed to solve multiobjective optimization problems. MOLFA has high computational efficiency and can obtain better Pareto optimal front.

2. The Markov chain of LFA is modelled, and martingale theory is applied to the optimal fitness value of population state discussing the convergence of the algorithm.

3. Numerical are conducted by employing the latest test function of CEC2017, which commendably validate the effectiveness of the proposed MOLFA in this paper.

The remainder of this paper is organized as follows: Section 2 introduces the mechanism of the lightning flash algorithm; Section 3 presents the multiobjective lightning flash algorithm and establishes the convergence of the multiobjective lightning flash algorithm via martingale theory; Section 4 provides a numerical example to verify the effectiveness of the proposed MOLFA; and finally, concluding remarks are given in Section 5.

2. Basic Principle of Lightning Flash Algorithm

Lightning is a natural phenomenon, generated by storm clouds (cumulonimbus clouds). The storm clouds generate negative electricity at the bottom and positive electricity at the top, as shown in Figure 1, which can be seen in [1]. As the electric field between the bottom of cloud and the ground becomes stronger, the charges ionize the around atmosphere, eventually breakdown air and open a way to the ground. In other words, when the electric field becomes stronger, an air column ionized appears at the bottom of cloud, also called stepped leader. After the formation of stepped leader, the charge branches connected to the tip of stepped leader develop rapidly and jump and struggle to find a path to the Earth. However, during the process of jumping, only one branch reaches the ground successfully. These charges at the tip of the winner branch also mean the better solution of the optimization problem in the algorithm.

As mentioned above, the lightning flash algorithm learns from the lightning phenomenon between clouds and ground. In this algorithm, each individual (called charge) represents a possible solution and has two characteristics: position and velocity. The initial populations of LFA are some different charges, randomly generated in a bounded interval, as shown in the following equation:

$$X = X_{\text{min}} + r (X_{\text{max}} - X_{\text{min}}),$$

(1)

where $X_{\text{max}}$ and $X_{\text{min}}$ are the maximum and minimum values of charge individual, respectively. $r \in [0, 1]$ is a random value.

The initial populations find the optimal solution by iterating and updating. First, all charge particles move to the ground along its channel; namely, each charge updates value according to the stepped leader of the channel. The moving distance of charge $i$ at channel $j$ is expressed as equation (2), and the position of charge is updated as equation (3). If any updated charge is a better fitness value than the stepped leader, the stepped leader will be replaced by the charge:

$$f_{ij}^t = w f_{ij}^{t-1} + c_1 r_1 (X_{ij}^{t-1} - X_{ij}^{t-1}),$$

(2)

$$X_{ij}^{t} = X_{ij}^{t-1} + f_{ij}^t,$$

(3)

where $t$ is the iteration number, $w$ is the inertia weight, $r_1 \in [0, 1]$ is a random value, $f_{ij}^t$ denotes the moving distance of charge $i$ in channel $j$ at the $t$-th iteration, $X_{ij}^{t-1}$ denotes the stepped leader of channel $j$ at the $(t-1)$-th iteration, $X_{ij}^{t}$ denotes the position of charge $i$ in lightning channel $j$ at the $t$-th iteration, and $c_1$ is a constant to ensure that all charges move downward in line with the stepped leader’s position.

Before the jump process takes place, the lightning channels have a pause time to rest. During the short rest time, the distance between the best and the worst charge in each channel is measured, which is a decision parameter in the jumping process later. As described in the following equation, $d_{ij}$ is the measured distance and $X_{ij}^{t}$ is the worst charge with farthest distance to the ground in channel $j$:
3. Multiobjective Lightning Flash Algorithm and Its Convergence

In this section, as an extension of LFA, we propose a multiobjective lightning flash algorithm to provide a reference method for the multiobjective optimization problem. After that, the convergence of MOLFA is analyzed on the basis of martingale theory.

3.1. Multiobjective Lightning Flash Algorithm. As described in [32], given a function \( f: \Omega \subset \mathbb{R}^n \rightarrow \mathbb{R}^n \), for \( X \in \Omega \), the value \( f(X^*) = -\infty \) is called a global minimum if and only if \( \forall X \in \Omega: f(X^*) \leq f(X) \), where \( X^* \) is the global minimum solution, \( f \) is the objective function, and the set \( \Omega \) is the feasible region of \( X \). The goal of determining the global minimum solution is called the global optimization problem for a single-objective problem.

Multiobjective problems are those problems where the goal is to optimize simultaneously \( D \) objective functions \( f_1(X), f_2(X), \ldots, f_D(X) \) and forming a vector function \( F(X) = [f_1(X), f_2(X), \ldots, f_D(X)]^\top \). A multiobjective minimum optimization problem is defined as

\[
\min F(X) = \min [f_1(X), f_2(X), \ldots, f_D(X)]^\top,
\]

where \( f_i(X) \) is the \( i \)-th objective function and \( X \) represents decision variable. If \( f_i(X^*) \leq f_i(X) \), \( i = 1, 2, \ldots, D \), and at least one \( i \) makes \( f_i(X^*) < f_i(X) \), \( i = 1, 2, \ldots, D \) true, then \( X^* \) is nondominated solutions; namely, \( X^* \) is superior to \( X \). If the inequalities are strictly hold, \( X^* \) is called optimal solution.

Based on the multiobjective particle swarm optimization (MOPSO) proposed by Coello et al. [33], the multiobjective lightning flash algorithm (MOLFA) is proposed in this paper. The principle of MOLFA is to use the adaptive grid mechanism to save external population. When the number of individuals in the external population exceeds a predetermined value, the space of their objective function is divided into hypercubes with equal spacing, and the number of individuals \( g_i \) in each grid is calculated (where \( i \) represents the grid number). The probability of particle to be selected in the \( i \)-th grid is \( p_i = 1/g_i \); that is, the more crowded the lower selection probability. The multiobjective lightning flash algorithm is described as Algorithm 1.

\[
\begin{align*}
    f_{ij}^t &= w f_{ij}^{t-1} + c_i r_i(X_{ij}^{t-1} - X_{ij}^{t-1}), \\
    X_{ij}^t &= X_{ij}^{t-1} + f_{ij}^t,
\end{align*}
\]

where \( X_{ij}^{t-1} \) is a solution variable taken from the repository and represents the stepped leader in this equation.

(2) Compute and evaluate the fitness values of new Pop, and then update the Rep together with the geographical representation of the particles within the hypercubes. If the number of charge particles in Rep is full, the adaptive grid criterion can be used to retain the nondominated solutions.
(3) Measure the critical distance between the best charge and the worst, and conduct the accumulation of charges. Update the charge by jump process.

(4) Go to 2.

(5) Increase the loop counter.

Step 5. End While.

3.2. Markov Chain of MOLFA. In MOLFA, charge particles in each channel have similar optimization development. In other words, one charge only relates to the condition of itself channel and independent of other lightning channels. Therefore, the following convergence analysis is given on a certain lightning channel for convenience.
According to equations (2)–(6), the state of charge particle \( i \) at the \( t \)-th iteration consists of \( X_{i}^{t-1} \) and \( X_{i}^{t} \), the stepped leader (the best charge) \( X_{L}^{t} \), and the worst charge \( X_{W}^{t} \); i.e., it is defined as \( c_{i}^{t} = (X_{i}^{t-1}, X_{i}^{t}, X_{L}^{t}, X_{W}^{t}) \). The set of all possible charge states is called charge state space defined as \( S \). For all charge particles, the charge population state at the \( t \)-th iteration is denoted as \( s_{t} = \{c_{1}^{t}, c_{2}^{t}, \ldots, c_{n}^{t}\} \), where \( n \) is the population size. The set of all possible charge population states is called charge population state space \( S \).

Proposition 1. If acceleration factors \( c_{1}^{r}, c_{2}^{r} \) and \( a_{3} \) of LFA satisfy uniform distribution, i.e., \( c_{1}^{r}, c_{2}^{r} \sim U(0, a) \) and \( a_{3} \sim U(0, a) \), then the probability of charge \( i \) transferring from \( X_{i}^{t} \) to the spherical region centered at \( X_{i}^{t+1} \) with radius \( \varepsilon \) can be computed as follows:

\[
P(c_{i}^{t+1} \mid c_{i}^{t}) = \begin{cases} \frac{\varepsilon^{2}}{a_{w} \|X_{i}^{t} - X_{i}^{t-1}\| \cdot \|X_{L}^{t} - X_{W}^{t}\|} & \text{case 1,} \\ \frac{\varepsilon^{3}}{a_{w} \|X_{i}^{t} - X_{i}^{t-1}\| \cdot \|X_{L}^{t} - X_{W}^{t}\| \cdot \|X_{W}^{t} - X_{L}^{t}\|} & \text{case 2,} \end{cases}
\]

where \( a_{w} \) is a constant, as described in equation (2). Case 1 is \( \|X_{L}^{t} - X_{W}^{t}\| \leq \delta \), and case 2 is \( \|X_{L}^{t} - X_{W}^{t}\| > \delta \).

Proof. In view of the basic principle of the lightning flash algorithm, we introduce the following three steps (downward movement, absorption, and jumping) to develop the transition probability of charge \( i \). We use \( P_{\text{down}}, P_{\text{abs}}, \) and \( P_{\text{jump}} \) to represent the probabilities of the downward movement step, absorption step, and jumping step, respectively.

Part (a): downward movement. When the population completes initialization, all charge particles start moving down. From equations (2) and (3), the downward movement formula can be described as follows:

\[
X_{i}^{t+1} = X_{i}^{t} + w(X_{i}^{t} - X_{i}^{t-1}) + c_{1} r (X_{L}^{t} - X_{i}^{t}).
\]

For case 1, since \( \delta \) is close to zero, we obtain \( X_{i}^{t+1} = X_{i}^{t} + w(X_{i}^{t} - X_{i}^{t-1}) \). Thus, the state transition probability of charge \( i \) from \( X_{i}^{t} \) to the spherical region centered at \( X_{i}^{t+1} \) with radius \( \varepsilon \) is calculated as

\[
P_{\text{down}}(X_{i}^{t+1} \mid c_{i}^{t}) = \frac{\int_{X_{i}^{t} - \varepsilon}^{X_{i}^{t} + \varepsilon} dy}{\int_{X_{i}^{t} - \varepsilon}^{X_{i}^{t} + \varepsilon} d y} = \frac{\varepsilon}{w \|X_{L}^{t} - X_{W}^{t}\|}.
\]

For case 2, we obtain \( X_{i}^{t+1} = X_{i}^{t} + w(X_{i}^{t} - X_{i}^{t-1}) + c_{1} r (X_{L}^{t} - X_{i}^{t}) \). Thus, the state transition probability of charge \( i \) is defined as

\[
P_{\text{down}}(X_{i}^{t+1} \mid c_{i}^{t}) = \left( \int_{X_{i}^{t} - \varepsilon}^{X_{i}^{t} + \varepsilon} dy \right) \left( \int_{X_{i}^{t} - \varepsilon}^{X_{i}^{t} + \varepsilon} dy \right) = \frac{\varepsilon}{a \|X_{L}^{t} - X_{W}^{t}\|}.
\]

Part (b): absorption: the absorption process means that the stepped leader will unify all charges in the channel; namely, the state of charge \( i \) must transfer from \( X_{i}^{t} \) to \( X_{L}^{t} \), and the state transition probability is

\[
P_{\text{abs}}(X_{L}^{t} \mid c_{i}^{t}) = 1
\]

Part (c): jumping: according to equations (5) and (6), the formula of the jumping process can be described as

\[
X_{i}^{t+1} = X_{i}^{t} + a_{3} (X_{L}^{t} - X_{W}^{t}).
\]

The transition probability is then computed as follows:
Due to the fact that \( X_t^l \) and \( X_t^r \) are both determined by \( X_t^{i+1} \), the transition probability of charge \( i \) from \( c_t^l \) to \( c_t^{2l+1} \) is obtained as follows:

\[
P(c_t^{i+1} | c_t^l) = \frac{\varepsilon}{\|X_t^l - X_t^r\|} \cdot \frac{\varepsilon}{\|X_t^L - X_t^r\|} \cdot \frac{\varepsilon}{\|X_t^r - X_t^W\|}.
\]  

(15)

Moreover, we assume that

\[
\frac{\varepsilon}{\|X_t^L - X_t^r\|} = 1, \quad \text{when} \|X_t^L - X_t^r\| \leq \delta.
\]  

(17)

On the basis of equation (16), we can easily obtain equation (9). This completes the proof.

Next, we will proceed to develop the transition probability of charge population state. The transition of charge population state means that, from \( s' \) to \( s'^{i+1} \), namely, all charge particles’ states of \( s' \) transferred to \( s'^{i+1} \) at the same time. So, the transition probability from \( s' \) to \( s'^{i+1} \) can be followed that

\[
P(s'^{i+1} | s') = \prod_{m=1}^{n} P(c_t^{i+1} | c_m^l).
\]  

(18)

**Proposition 2.** The charge population state sequence \( \{s', t \geq 1\} \) is a Markov chain.

**Proof.** According to equations (2)–(6), the charge population state \( s'^{i+1} = \{c_t^{i+1}, c_t^{2i+1}, \ldots, c_n^{i+1}\} \) only depends on the state of last iteration. Thus, \( \{s', t \geq 1\} \) is a Markov chain. The proof is completed.

In this paper, \( X_{Rep} \) is employed to denote a certain solution of the Pareto optimal front (POF). If the stepped leader \( X_t^l \) of charge population equals \( X_{Rep} \), it indicates that a certain optimum solution of POF has been found by population. The optimal charge population state is defined as \( s_t = (c_t^1, c_t^2, \ldots, c_t^n) \), where \( c_t^i = (X_t^{i-1}, X_t^r, X_{Rep}, X_t^W) \). Besides, the optimal charge population state set is defined as \( \Gamma = \{s_t = (c_t^1, c_t^2, \ldots, c_t^n), t \in N^+\} \).

### 3.3. Convergence Analysis of MOLFA

The martingale is a stochastic process defined by conditional mathematical expectation, which is a basic tool of probability analysis. In the following, we present two definitions and one lemma to derive our main results.

**Definition 1.** A stochastic process \( \{Y_t, \ t = 0, 1, 2, \ldots\} \) is a martingale if for all \( t \) the following conditions are satisfied:

1. \( \mathbb{E}[|Y_t|] < \infty \).
2. \( \mathbb{E}[Y_{t+1} | Y_0, Y_1, \ldots, Y_t] = Y_t. \)

**Definition 2.** For given two stochastic processes \( \{Y_t, t = 0, 1, 2, \ldots\} \) and \( \{Z_t, t = 0, 1, 2, \ldots\} \), \( \{Y_t, t = 0, 1, 2, \ldots\} \) is a supermartingale (or submartingale) with respect to \( Z_t \) if \( Y_t \) meets the following conditions:

1. \( Y_t \) is a function of \( Z_0, Z_1, \ldots, Z_t \).
2. \( \mathbb{E}[|Y_t|] < \infty \).
3. \( \mathbb{E}[Y_{t+1} | Z_0, Z_1, \ldots, Z_t] \leq (\geq) Y_t. \)

**Lemma 1** (see [30]). If stochastic process \( \{Y_t, t = 0, 1, 2, \ldots\} \) is a supermartingale with respect to \( Z_t \) and \( \sup_{t \geq 0} \mathbb{E}[|Y_t|] < \infty \), then the state sequence \( \{Y_t, t = 0, 1, 2, \ldots\} \) converges to a state \( Y^\infty \) with probability one, namely, \( \mathbb{P} \lim_{t \rightarrow \infty} Y_t = Y^\infty = 1 \).

The purpose of MOLFA is to search for Pareto optimal front, a best repository, i.e., the optimal charge population state. The POF of charge population state is defined as

\[
F(s') = [f_1(X_t^l), f_2(X_t^l), \ldots, f_D(X_t^l)],
\]  

(19)

where \( X_t^l \) is a vector of nondominated solutions of \( \text{Rep} \). For solving multiobjective optimization problem, POF can be obtained only when the optimal charge population state is searched. Therefore, the convergence analysis of MOLFA can be transformed into the convergence study of \( F(s') \). When the conditional expectation of \( F(s') \) with new charge population state is less than or equal to the current population state, \( F(s') \) can be transformed into a supermartingale. Since \( F(s') \) is a monotone nonincremental process, the convergence of the supermartingale can be investigated.

**Theorem 1.** The stochastic process of population state sequence \( \{F(s'), t = 1, 2, \ldots\} \) is a nonnegative and bounded supermartingale with respect to population state sequence \( \{s', t = 1, 2, \ldots\} \), i.e., \( \mathbb{E}[F(s^{t+1}) | s^1, s^2, \ldots, s^t] \leq F(s') \). Moreover, as \( t \rightarrow \infty \), \( s' \) will converge to \( s^\infty \) with probability 1, i.e., \( \mathbb{P} \lim_{t \rightarrow \infty} s' = s^\infty = 1 \).

**Proof.** Stochastic process \( \{F(s'), t = 1, 2, \ldots\} \) is obviously nonnegative and bounded. According to Proposition 2, \( \{s', t = 1, 2, \ldots\} \) is a Markov chain. It then follows that

\[
\mathbb{E}[F(s^{t+1}) | s^1, s^2, \ldots, s^t] = \mathbb{E}[F(s^{t+1})] = F(s') \leq F(s').
\]  

(20)

It can be seen that we only need to prove that

\[
\forall y \in S, \mathbb{E}[F(s^{t+1}) | s' = y] \leq F(y).
\]  

(21)

Since the left side of (21) follows that
the proof can be transformed to show
\[ \sum_{z \in S} F(z) P(s^{t+1} = z \mid s^t = y) \leq \sum_{z \in S} F(y) P(s^{t+1} = z \mid s^t = y) = F(y), \]

which indicates that \( \{F(s^t), t = 1, 2, \ldots\} \) is a nonnegative and bounded supermartingale with respect to \( \{s^t, t = 1, 2, \ldots\} \). This completes the proof. \( \square \)

**Theorem 2.** The stochastic process \( \{F(s^t), t = 1, 2, \ldots\} \) will converge to the global optimal solution \( \Gamma \).

**Proof.** According to Theorem 1, we know that \( \{F(s^t), t = 1, 2, \ldots\} \) is a bounded supermartingale. Then, by Lemma 1, it can be deduced that it can converge to the global optimal solution \( \Gamma \) with probability one. This completes the proof. \( \square \)

**Remark 2.** In Theorem 2, the convergence of MOLFA is proposed by using the martingale sequence convergence theorem. In terms of the principle of the algorithm, the jumping process guarantees the diversity of the population and prevents the population falling into local optimum. Therefore, the MOLFA can converge to the Pareto optimal front with probability one.

### 4. Experiments on the Performance of MOLFA

As further discussed in [1], it indicates that the LFA method has better search ability and convergence property than many other heuristic algorithms. In this paper, the latest test function of CEC2017 is utilized to validate performance and effectiveness of the proposed MOLFA, where similar multiobjective optimization test problems are considered in [34, 35] and the latest test function of CEC2017 can be found in [36]. It should be pointed out that the test functions include ZDT1, ZDT2, ZDT3, ZDT4, and ZDT6 problems, where the Pareto front of ZDT3 is noncontinuous; ZDT2 and ZDT6 are convex; and ZDT1 and ZDT4 are nonconvex. All problems have two objective functions described in Table 1. The table also shows the number of variables and their bounds for each problem. These typical test functions could cover the basic types of multiobjective optimization problems. In the following experiments, a charge population size of 100 and a repository size of 250 are utilized to test these multiobjective problems.

For multiobjective optimization problems, we need to consider that whether the final optimal solutions are close to the true Pareto front, and the distribution of the solution is uniform and smooth. We use three indices, e.g., generational distance, spacing, and error ratio, to conduct a quantitative assessment of the performance of MOLFA.

Generational distance (GD) [37] is to estimate the gap between the nondominated vectors produced by algorithm and true Pareto optimal front, which can be defined as
\[ GD = \frac{\sum_{i=1}^{n} d_i^2}{n}, \]

where \( n \) is the number of nondominated solutions obtained by the algorithm and \( d_i \) is the Euclidean distance between the \( i \)-th solution and the nearest member of the Pareto optimal front. It indicates that all the nondominated vectors can be regarded as the true Pareto optimal solution set when GD = 0.

Spacing (SP) [38] is used to evaluate the diversity of the obtained Pareto optimal solutions by comparing the uniform distribution and the deviation of solutions, which can be defined as
\[ SP = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (\overline{d} - d_i)^2}, \]

where \( \overline{d} \) is the average value of \( d_i \) and \( d_i \) also denotes the Euclidean distance between the \( i \)-th solution and the nearest member of the Pareto optimal front. It can be seen that all nondominated solutions are evenly spread if SP = 0.

Error ratio (ER) [39] is employed to report the number of nondominated vectors in Pareto front that are not members of true Pareto front, which can be defined as
\[ ER = \frac{\sum_{i=1}^{n} e_i}{n}, \]

where \( e_i = 0 \) or 1. For the case \( e_i = 0 \), it indicates that the \( i \)-th solution can be a member of the true Pareto optimal front. Otherwise, it means that the \( i \)-th solution is not a member of the true Pareto optimal front.

The simulated results for these test problems are shown in Figures 3–7. The continuous red line is the true Pareto front. To conveniently show the distribution of the nondominated solution set produced by MOLFA, all solutions are connected with straight lines, as shown in the blue dot line. The closer the blue dot line is to the red line, the more accurate the solution obtained by the MOLFA. Besides,
Table 1: Standard test problems.

| Problem | $n$ | Variable bounds | Objective functions |
|---------|-----|-----------------|---------------------|
| ZDT1    | 30  | $x_i \in [0, 1], 1 \leq i \leq n$ | $f_1(x) = x_1, f_2(x) = g(x)[1 - \sqrt{x_i/g(x)}], g(x) = 1 + 9 \sum_{i=2}^{n} x_i/n - 1$ |
| ZDT2    | 30  | $x_i \in [0, 1], 1 \leq i \leq n$ | $f_1(x) = x_1, f_2(x) = g(x)[1 - (x_i/g(x))^2], g(x) = 1 + 9 \sum_{i=2}^{n} x_i/n - 1$ |
| ZDT3    | 30  | $x_i \in [0, 1], 1 \leq i \leq n$ | $f_1(x) = x_1, f_2(x) = g(x)[1 - (x_i/g(x)) - (x_i/g(x))\sin(10\pi x_i)], g(x) = 1 + 9 \sum_{i=2}^{n} x_i/n - 1$ |
| ZDT4    | 10  | $x_i \in [-5, 5], 2 \leq i \leq n, x_1 \in [0, 1]$ | $f_1(x) = x_1, f_2(x) = g(x)[1 - \sqrt{x_i/g(x)}], g(x) = 1 + 10(n - 1) + \sum_{i=2}^{n} [x_i^2 - 10 \sin(4\pi x_i)]$ |
| ZDT6    | 10  | $x_i \in [0, 1], 1 \leq i \leq n$ | $f_1(x) = 1 - \exp[-4x_1] \sin^n(6\pi x_1), f_2(x) = g(x)[1 - (f_1(x)/g(x))^2], g(x) = 1 + 9(\sum_{i=2}^{n} x_i/n - 1)^{0.25}$ |

Figure 3: Pareto front produced by MOLFA for the ZDT1 test function.

Figure 4: Pareto front produced by MOLFA for the ZDT2 test function.

Figure 5: Pareto front produced by MOLFA for the ZDT3 test function.

Figure 6: Pareto front produced by MOLFA for the ZDT4 test function.
Table 2 shows the results of metric for the test problems. It can be seen that most of solutions are close to the true Pareto front and are unevenly distributed. Therefore, it is demonstrated that the proposed MOLFA method is able to solve these standard multiobjective optimization problems, which provide an effective nondominated solution set and cover the full Pareto front.

5. Conclusion

In this paper, a multiobjective optimization algorithm MOLFA is proposed. In order to discuss the convergence property of MOLFA, the Markov chain model and Martingale theory have been introduced. The stochastic process of the optimal charge population state has been transformed into a supermartingale, and the global convergence of MOLFA algorithm has been proved. At last, the convergence property has been investigated on the standard test functions. The simulation results show that the MOLFA method has a good performance on converging to the Pareto optimal front. The convergence property of the MOLFA method is firstly proved from theoretical aspect via the martingale theory, which may provide a new way for the theoretical research of multiobjective optimization problems.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported in part by the National Key R&D Program of China (Grant no. 2016YFB0900600), National Natural Science Foundation of China (Grant nos. 51877174 and 61903296), State Key Laboratory of Advanced Electromagnetic Engineering and Technology (Huazhong University of Science and Technology) (Grant no. AEET2018KF001), State Key Laboratory of Electrical Insulation and Power Equipment (Xi’an Jiaotong University) (Grant no. EIPE18201), and High Level Talents Plan of Shaanxi Province for Young Professionals, and partially supported by National Research Foundation (NRF), Singapore, under Future Resilient Systems Programme-Stage II.

References

[1] M. Kheshti, A novel optimization method-lightning flash algorithm and its applications in power system, Ph.D. Dissertation, Xi’an Jiaotong University, Xi’an, China, 2017.
[2] M. Kheshti, X. Kang, Z. Bie, Z. Jiao, and X. Wang, “An effective lightning flash algorithm solution to large scale non-convex economic dispatch with valve-point and multiple fuel options on generation units,” Energy, vol. 129, pp. I–15, 2017.
[3] M. Kheshti, X. Kang, J. Li, P. Regulski, and V. Terzija, “Lightning flash algorithm for solving non-convex combined emission economic dispatch with generator constraints,” IET Generation, Transmission & Distribution, vol. 12, no. 1, pp. 104–116, 2018.
[4] M. Kheshti and X. Kang, “Optimal overcurrent relay coordination in distribution network based on lightning flash algorithm,” Engineering Computations, vol. 35, no. 3, pp. 1140–1160, 2018.
[5] M. Kheshti, L. Ding, M. Nayeripour, X. Wang, and V. Terzija, “Active power support of wind turbines for grid frequency events using a reliable power reference scheme,” Renewable Energy, vol. 139, pp. 1241–1254, 2019.
[6] M. Kheshti, L. Ding, W. Bao, M. Yin, Q. Wu, and V. Terzija, “Toward intelligent inertial frequency participation of wind farms for the grid frequency control,” IEEE Transactions on Industrial Informatics, vol. 16, no. 11, pp. 6772–6786, 2020.
[7] B. Cao, J. Zhao, Z. Lv et al., “Distributed parallel particle swarm optimization for multi-objective and many-objective large-scale optimization,” IEEE Access, vol. 5, pp. 8214–8221, 2017.
[8] M. A. Hannan, J. A. Ali, A. Hussain et al., “A quantum lightning search algorithm-based fuzzy speed controller for induction motor drive,” IEEE Access, vol. 6, pp. 1214–1223, 2018.
[9] H.-C. Huang and S.-K. Lin, “A hybrid metaheuristic embedded system for intelligent vehicles using hypermutated firefly algorithm optimized radial basis function neural network,” IEEE Transactions on Industrial Informatics, vol. 15, no. 2, pp. 1012–1019, 2019.
[10] A. A. Al-Omoush, A. A. Alsewari, H. S. Alamri, and K. Z. Zamli, “Comprehensive review of the development of the harmony search algorithm and its applications,” IEEE Access, vol. 7, pp. 14233–14245, 2019.
[11] Q. Lin, J. Li, Z. Du, J. Chen, and Z. Ming, “A novel multi-objective particle swarm optimization with multiple search strategies,” European Journal of Operational Research, vol. 247, no. 3, pp. 732–744, 2015.

[12] R. Shigenobu, A. S. Noorzad, A. Yona, and T. Senjyu, “Multi-objective optimisation of step voltage regulator operation and optimal placement for distribution systems design using linkage combination update-non-dominated sorting genetic algorithm-II,” IET Generation, Transmission & Distribution, vol. 12, no. 1, pp. 20–30, 2018.

[13] H. Y. Tian, S. J. Li, T. Q. Wu, and M. Yao, “An extreme learning machine based on artificial immune system,” Computational Intelligence and Neuroscience, vol. 2018, Article ID 3635845, 10 pages, 2018.

[14] X. Y. Liu, A. L. Zhang, Y. L. Gao, and W. Zhao, “A novel hybrid immune algorithm and its convergence based on the steepest descent algorithm,” Applied Mathematics and Computation, vol. 218, no. 4, pp. 1291–1296, 2011.

[15] X. L. Jin, L. H. Ma, T. J. Wu, and J. X. Qian, “Convergence analysis of the particle swarm optimization based on stochastic processes,” Acta Automatica Sinica, vol. 33, no. 12, pp. 1263–1268, 2007.

[16] Z.-G. Lu, H. Zhao, H.-F. Xiao, H.-R. Wang, and H.-J. Wang, “An improved multi-objective bacteria colony chemotaxis algorithm and convergence analysis,” Applied Soft Computing, vol. 31, pp. 274–292, 2015.

[17] Y. Wei, J. Qiu, P. Shi, and L. Wu, “A piecewise-markovian lyapunov approach to reliable output feedback control for fuzzy-affine systems with time-delays and actuator faults,” IEEE Transactions on Cybernetics, vol. 48, no. 9, pp. 2723–2735, 2018.

[18] Y. Wei, J. Qiu, H. R. Karimi, and W. Ji, “A novel memory filtering design for semi-Markovian jump time-delay systems,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 48, no. 12, pp. 2229–2241, 2018.

[19] X. Liu, G. Ma, P. R. Pagilla, and S. S. Ge, “State-estimator-based asynchronous repetitive control of discrete-time Markovian switching systems,” Complexity, vol. 2020, pp. 1–13, 2020.

[20] X. Liu, G. Ma, P. R. Pagilla, and S. S. Ge, “Dynamic output feedback asynchronous control of networked Markovian jump systems,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 50, no. 7, pp. 2705–2715, 2020.

[21] O. Francois, “An evolutionary strategy for global minimization and its Markov chain analysis,” IEEE Transactions on Evolutionary Computation, vol. 2, no. 3, pp. 77–90, 1998.

[22] D. Wu, S. Xu, and F. Kong, “Convergence analysis and improvement of the chicken swarm optimization algorithm,” IEEE Access, vol. 4, pp. 9400–9412, 2016.

[23] H. Huang, C. G. Wu, and Z. F. Hao, “A pheromone-rate-based analysis on the convergence time of ACO algorithm,” IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 39, no. 4, pp. 910–923, 2009.

[24] T. Nakama, “Markov chain analysis of genetic algorithms applied to fitness functions perturbed concurrently by additive and multiplicative noise,” Computational Optimization and Applications, vol. 51, no. 2, pp. 601–622, 2012.

[25] A. Ning and X. Y. Zhang, “Convergence analysis of artificial bee colony algorithm,” Control and Decision, vol. 28, no. 10, pp. 1554–1558, 2013.

[26] S. Chen, G.-H. Peng, X.-S. He, and X.-S. Yang, “Global convergence analysis of the bat algorithm using a Markovian framework and dynamical system theory,” Expert Systems with Applications, vol. 114, pp. 173–182, 2018.

[27] J. Xue, Y. Wang, H. Li et al., “A smart wolf pack algorithm and its convergence analysis,” Control and Decision, vol. 31, no. 12, pp. 2131–2139, 2016.

[28] F. Pan, X.-T. Li, Q. Zhou, W.-X. Li, and Q. Gao, “Analysis of standard particle swarm optimization algorithm based on Markov chain,” Acta Automatica Sinica, vol. 39, no. 4, pp. 381–389, 2013.

[29] T. Yang and Y.-X. Ren, “Limit theorem for derivative martingale at criticality w.r.t branching Brownian motion,” Statistics & Probability Letters, vol. 81, no. 2, pp. 195–200, 2011.

[30] G. Xu and G. Yu, “On convergence analysis of particle swarm optimization algorithm,” Journal of Computational and Applied Mathematics, vol. 333, pp. 65–73, 2018.

[31] H. B. Duan, D. Wang, and X. F. Yu, “Markov chains and martingale theory based convergence proof of ant colony algorithm and its simulation platform,” in Proceedings of the 6th World Congress on Intelligent Control and Automation, pp. 3057–3061, Dalian, China, June 2006.

[32] G. Chiandussi, M. Codecone, S. Ferrero, and F. E. Varesio, “Comparison of multi-objective optimization methodologies for engineering applications,” Computers & Mathematics with Applications, vol. 63, no. 5, pp. 912–942, 2012.

[33] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga, “Handling multiple objectives with particle swarm optimization,” IEEE Transactions on Evolutionary Computation, vol. 8, no. 3, pp. 256–279, 2004.

[34] E. Zitzler, K. Deb, and L. Thiele, “Comparison of multi-objective evolutionary algorithms: empirical results,” Evolutionary Computation, vol. 8, no. 2, pp. 173–195, 2000.

[35] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, “Scalable multi-objective optimization test problems,” in Proceedings of the 2002 Congress on Evolutionary Computation, pp. 825–830, Honolulu, HI, USA, May 2002.

[36] N. H. Awad, M. Z. Ali, J. J. Liang, B. Y. Qu, and P. N. Suganthan, “Problem definitions and evaluation criteria for the CEC2017 special session and competition on single objective bound constrained real-parameter numerical optimization,” Technical Report, Nanyang Technological University, Singapore, Singapore, 2016.

[37] K. Deb, A. Pratap, S. Agrawal, and T. Meyarivan, “A fast elitist multiobjective genetic algorithm: NSGA-II,” IEEE Transactions on Evolutionary Computation, vol. 6, no. 2, pp. 182–197, 2002.

[38] J. R. Schott, “Fault tolerant design using single and multi-criteria genetic algorithm optimization,” M. S. thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA, USA, 1995.

[39] D. A. Van Veldhuizen, “Multiobjective evolutionary algorithms: classifications, analyzes, and new innovations,” Ph.D. Dissertation, Department of Electrical Computer Engineering, Air Force Institute of Technology, Wright-Patterson, OH, USA, 1999.