Theoretical analysis of garden balsam optimization algorithm

Xiaohui Wang and Shengpu Li

College of Information Engineering, Pingdingshan University, Pingdingshan, People’s Republic of China; College of Information Science and Technology, Donghua University, Shanghai, People’s Republic of China

ABSTRACT

Garden balsam optimization (GBO) is a new proposed evolutionary algorithm based on swarm intelligence. Convergence and time complexity analyses are very important in evolutionary computation, but the research on GBO is still blank. Same as other evolutionary algorithms, the optimization process of the GBO algorithm can be regarded as a Markov process. In this paper, a Markov stochastic model of the GBO algorithm is defined and used to prove the convergence of GBO algorithm. Finally, the approximation region of the estimated convergence time of GBO algorithm is calculated, which characterizes the evolution of the evolutionary process of the proposed algorithm.

1. Introduction

Evolution algorithm (EA) is a kind of adaptive search algorithm that is generated by a natural evolution process. With the fruitful results of evolutionary algorithms in the application of various types of optimization problems, the theoretical study of such algorithms has increasingly attracted the attention of scholars.

As a new evolution algorithm, garden balsam optimization (GBO) was proposed by Li and Sun (2020). The GBO algorithm was inspired by the unique propagation process of garden balsam seeds. The fruit of the garden balsam is called a capsule. When the capsules are ripe, they crack on their own. The mechanical force of the bursting process scatters the seeds around the parent body. The number and distribution of seeds are directly related to the environment in which the parent grows. The seed that falls to the ground has the opportunity to move under the force of nature. To simulate this process, the GBO algorithm generates seeds by a mechanical propagator operator and a second propagator operator to search for an optimal solution in the problem space.

The flow chart of GBO algorithm is shown in Figure 1. It can be seen that the algorithm first needs to initialize the population, and then, the mechanical transmission operator and the second transmission operator are executed in turn. When the seed crosses the boundary, it needs to be pulled back according to the mapping rules. If the number of seeds is too large, a selection strategy should be implemented to eliminate them. This cycle iterates until the termination condition is met, that is, the accuracy requirement of the problem is met or the maximum number of iterations is reached (Li & Sun, 2020).

Although the mechanism of GBO algorithm is simple, it has been proved that it can converge effectively on constrained optimization problems and get the optimal solution (Li & Sun, 2020). The ability of the algorithm to solve multi-dimensional functions was also proved by (Li & Sun, 2020). The GBO algorithm is also used to solve practical problems, for example, Li et al. (2019) used it to optimize the parameters of the adaptive-network-based fuzzy inference system (ANFIS).

As a new kind of population-based evolutionary algorithm, the GBO algorithm has not been analyzed for its running time, since it was proposed. The analysis of runtime is a hot topic in the theoretical study of evolutionary algorithms in recent years (Han et al., 2008; Oliveto et al., 2007). Intuitively, the goal of runtime analysis is to find at least one optimal solution or a good approximate optimal solution in operation. The runtime can be measured by the time it first arrives at a particular set of states in the dependent process (Doerr et al., 2012). Due to the random nature of evolutionary algorithms, the computational time analysis of such algorithms is not easy. The runtime is helpful to deepen the understanding of the evolutionary algorithm, evaluate the efficiency of the algorithm, and improve it.

Early studies focused on the runtime of \((1 + 1)\)EA and other simple evolutionary algorithms to solve pseudo-
Boolean functions, which usually have good structural properties (Droste et al., 2002). These studies show some useful mathematical methods and tools and lead to some theoretical results related to some examples. At present, the computational time analysis of $(1 + 1)$EA has gradually expanded from simple pseudo-Boolean functions to combinatorial optimization problems with a practical application background. Oliveto et al. analyzed the computing time of some instances of $(1 + 1)$EA solving vertex covering problems (Oliveto et al., 2009). Lehre et al. selected several examples of computing unique input-output sequences and analyzed the computing time of $(1 + 1)$EA (Lehre & Yao, 2014). Zhou et al. carried out a series of approximate performance analysis for some examples of $(1 + 1)$EA solving the following combinatorial optimization problems: minimum label spanning tree problem (Lai et al., 2014), multi-processor scheduling problem (Zhou, Lai, et al., 2015), maximum cutting problem (Zhou, Lai, et al., 2015), and maximum leaf spanning tree problem (Xia et al., 2015), and achieved fruitful theoretical results.

With the development of $(1 + 1)$ EA theory research, many mathematical methods and tools have been proposed, such as Markov chain (He & Yao, 2003), absorbing and absorbing Markov chain (Yu & Zhou, 2008), switch analysis (Yu et al., 2015), and a method based on adaptive value partitioning (Sudholt, 2013). Drift analysis (He & Yao, 2001), introduced by He et al., has proved to be a powerful technique for evolutionary algorithms to runtime analysis.

He and Yao (2002) used the Markov Modal and first hitting time theory to study the first hit probability of the $(N + N)$ evolutionary algorithm, and found that it is feasible to appropriately increase the population size. A new method for estimating the expected first hit time was proposed by Yang and Zhou (2008), which is also used to analyze the evolutionary algorithm of different configurations. Based on the absorption Markov process, the convergence of ACO algorithm was studied by (Huang et al., 2009). Chen et al. (2010) analyzed the time complexity of a simple EDA to further understand its complexity. Yi et al. (2011) drew a conclusion that QEA converges in odds under some loose assumptions. Ding and Yu (2012) introduced some time complexity analysis techniques of EAs based on limited search space. The precise analytical expression of the average first hit is obtained, that is, evolutionary algorithms reach the optimum solution.

In this paper, the Markov process and expected convergence time are used to analyze the convergence and time complexity of GBO algorithm. First, the Markov stochastic process of GBO is given and its theoretical model is established. Then, combining with the basic mechanism of GBO algorithm, its convergence will be studied. Finally, the expected convergence time of GBO will be discussed in detail.

The remainder of this paper is organized as follows. Section 2 constructs the Markov modal of the GBO algorithm. The global convergence of GBO algorithm is proved in Section 3. Section 4 analyzes the time complexity of GBO algorithm. Section 5 concludes this paper.

2. The random modal of GBO

The GBO algorithm is mainly used to solve continuous optimization problems (take the global minimization problem for example) as follows:

$$
\min f(x) \text{ s.t. } x \in S \subset \mathbb{R}^d
$$

where the objective function $f(x) \neq \text{const}$ maps from problem space $S$ to state space $R$ and $d$ is the number of dimensions. In combination with the GBO algorithm flow described in the previous section, its mathematical model is given here.

Definition 2.1: Let $\{\xi(\tau)\}_{\tau=0}^{\infty}$ be a stochastic process of the GBO algorithm, where $\xi(\tau) = (W(\tau), J(\tau))$. Then $W(\tau) = (W_1(\tau), W_2(\tau), \ldots, W_n(\tau))$ denotes the position of $N$ parents in the step $\tau$. Moreover, $J(\tau) = (L(\tau), Z(\tau))$, where

- $W_1(\tau)$ represents the first seed transmitted to the next generation population.
- $W_2(\tau)$ represents the second seed transmitted to the next generation population.
- $W_3(\tau)$ represents the cross-border seed mapping.
- $W_4(\tau)$ represents the screening of the next generation population.

Figure 1. Framework of the GBO algorithm.
where \( L(t) = (L_1(t), L_2(t), \ldots, L_n(t)) \) denotes the transmission distance of \( N \) parents and \( Z(t) = (Z_1(t), Z_2(t), \ldots, Z_n(t)) \) denotes the seed number of \( N \) parents.

**Definition 2.2:** Let \( R_e = \{ x \in S | f(x) - f(x^*) < \varepsilon, \varepsilon < 0 \} \) be the optimum region of \( f(x) \), where \( x^* \) denotes the finest solution in \( S \).

In accordance with this definition, for the objective function \( f(x) \) to have a solution, it has to satisfy \( \nu(R_e) > 0 \), where \( \nu(R_e) \) denotes the Lebesgue measure of \( R_e \).

**Definition 2.3:** The finest condition of GBO is defined as \( \xi^*(\tau) = \{ W^*(\tau), J(\tau) \} \), where \( W_i(\tau) \in R_e \) and \( W_i(\tau) \in W^*(\tau), i \in 1, 2, \ldots, n \).

In accordance with the above definition, it means that the optimum parent in the optimum condition \( \xi^*(\tau) \) of GBO is in the optimum region \( R_e \). So exist \( W_i(\tau) \in R \) and \( f(W_i(\tau)) - f(x^*) < \varepsilon, x^* \in R_e \).

**Lemma 2.1:** The random process of GBO, \( \{ \xi(\tau) \}_{\tau=0}^{+\infty} \) is Markov random process.

**Proof:** \( \{ \xi(\tau) \}_{\tau=0}^{+\infty} \) is the discrete random process, because the condition \( \xi(\tau) = \{ W(\tau), J(\tau) \} \) is decided by the \( \{ W(\tau - 1), J(\tau - 1) \} \), so the odds \( P[\xi(\tau + 1)|\xi(1), \xi(2), \ldots, \xi(\tau)] = P[\xi(\tau + 1)|\xi(\tau)] \), which means that the odds of \( (\tau + 1) \)-th condition occurring are not related to the odds of \( \tau \)-th condition occurring.

So, the \( \{ \xi(\tau) \}_{\tau=0}^{+\infty} \) is the Markov random process. The proof is completed.

**Definition 2.4:** Let \( U \) be the state space of GBO’s state \( \xi(\tau) \), \( U^* \subset U \). If for any condition \( \xi^*(\tau) = \{ W^*(\tau), \tau \} \in U \), there exists a solution \( g^* \in W^* \), then \( g^* \in R_e \). Then \( U^* \) is called optimum state space.

In accordance with Definition 2.4, when the state of the garden balsam optimization can search the optimum state space, there is a seed in the optimum region \( R_e \) and the optimum position of search space has been fined by the GBO. After that, the optimal solution must always be in the optimum region.

**Definition 2.5:** Given a Markov random process \( \{ \xi(\tau) \}_{\tau=0}^{+\infty} \) and optimum condition space \( U^* \subset U \), if \( \{ \xi(\tau) \}_{\tau=0}^{+\infty} \) s.t. \( P[\xi(\tau + 1) \notin U^*|\xi(\tau) \in U^*] = 0 \), \( \{ \xi(\tau) \}_{\tau=0}^{+\infty} \) is called an absorbing Markov course.

**Lemma 2.2:** The random process of GBO, \( \{ \xi(\tau) \}_{\tau=0}^{+\infty} \), is an absorbing Markov random process.

**Proof:** In accordance with Lemma 2.1, the random process of GBO, \( \{ \xi(\tau) \}_{\tau=0}^{+\infty} \), is a Markov random process. If \( \{ \xi(\tau) \}_{\tau=0}^{+\infty} \) stays in the optimum problem space \( R_e \), the condition \( \xi(\tau) = \{ W(\tau), J(\tau) \} \) must belong to the optimum condition space \( U^* \).

Let \( W_i(\tau) \) is the optimum site in \( N \) parents of GBO, then \( f(W_i(\tau + 1)) - f(W_i(\tau)) \leq 0 \). So, the condition \( \xi(\tau + 1) \) must belong to the optimum condition space \( U^* \).

Hence, \( P[\xi(\tau + 1) \notin U^*|\xi(\tau) \in U^*] = 0 \), the random process of GBO, \( \{ \xi(\tau) \}_{\tau=0}^{+\infty} \), is an absorbing Markov course. The proof is completed.

3. Convergence of GBO

Convergence is an important index to measure the performance of evolutionary algorithms. Because there are many kinds of optimization problems, it is impossible for any optimization algorithm to converge quickly on all optimization problems. Without the loss of generality, this paper discusses the convergence of GBO when dealing with simple continuous optimization problems. A detailed derivation is given below.

**Definition 3.1:** Suppose that \( \{ \xi(\tau) \}_{\tau=0}^{+\infty} = \{ W(\tau), J(\tau) \} \) is an absorbing Markov course, \( U^* \subset U \) is the optimum condition space, and \( \lambda(\tau) = P[\xi(\tau) \in U^*] \) denotes the probability that the stochastic state arrives at the optimal state in time \( \tau \). If \( \lim_{\tau \to +\infty} \lambda(\tau) = 1 \), then \( \{ \xi(\tau) \}_{\tau=0}^{+\infty} \) is convergent.

As can be seen from Definition 2.6, the probability of \( P[\xi(\tau) \in U^*] \) determines whether the Markov random process converges.

**Theorem 3.1:** Suppose that \( \{ \xi(\tau) \}_{\tau=0}^{+\infty} \) is an absorbing Markov course of GBO and \( U^* \subset U \) is the optimum condition space. If \( P[\xi(\tau) \in U^*|\xi(\tau - 1) \notin U^*] \geq \alpha > 0 \), \( P[\xi(\tau) \in U^*] \geq 1 \) for any \( \tau \), then \( P[\xi(\tau) \in U^*] \geq 1 - (1 - \alpha)^\tau \).

**Proof:** Let \( \tau = 1 \),

\[
P[\xi(1) \in U^*] = P[\xi(1) \in U^*|\xi(0) \in U^*] \cdot P[\xi(0) \in U^*] \\
+ P[\xi(1) \in Y^*|\xi(0) \notin U^*] \cdot P[\xi(0) \notin U^*] \\
\geq P[\xi(0) \notin U^*] + \alpha \cdot P[\xi(0) \notin U^*] \\
= P[\xi(0) \in U^*] + \alpha \cdot (1 - P[\xi(0) \in U^*]) \\
= \alpha + (1 - \alpha) \cdot P[\xi(0) \in U^*]
\]

Because \( 1 - \alpha \geq 0 \), so \( \alpha + (1 - \alpha) \cdot P[\xi(0) \in Y^*] \geq \alpha \), then \( P[\xi(1) \in Y^*] \geq \alpha = 1 - (1 - \alpha)^1 \).
Suppose that \( P(\xi(\tau) \in U^*) \geq 1 - (1 - \alpha)^\tau \) is held when \( \tau < a - 1 \), then for \( \tau = a \),

\[
P(\xi(a) \in U^*) = P(\xi(a) \in U^* | \xi(a - 1) \in U^*) \cdot P(\xi(a - 1) \in U^*) + P(\xi(a) \in U^* | \xi(a - 1) \notin U^*) \cdot P(\xi(a - 1) \notin U^*)
\]

\[
\geq P(\xi(a - 1) \in U^*) + \alpha \cdot (1 - P(\xi(a - 1) \in U^*)) = \alpha + (1 - \alpha) \cdot (1 - \alpha)^{a-1} = 1 - (1 - \alpha)^a
\]

Therefore, \( P(\xi(\tau) \in U^*) \geq 1 - (1 - \alpha)^\tau \) is held when \( \tau \geq 1 \).

The proof is completed. ■

The GBO algorithm has a second transmission operator. In the algorithm implementation process, the secondary propagation operator can use Gaussian variation and so on. Here, for simplicity, simple random variation is used.

**Theorem 3.2:** Suppose that \( \{\xi(\tau)\}^{+\infty}_{\tau=0} \) is an absorbing Markov course of GBO and \( U^* \subset U \) is the optimum condition space, then \( \lim_{\tau \to +\infty} \lambda(\tau) = 1 \), which means that \( \{\xi(\tau)\}^{+\infty}_{\tau=0} \) will converge to the optimum condition \( U^* \).

**Proof:** The GBO algorithm has a second transmission operator. After the second transmission, the probability of the seed falling into \( R_e \) can be expressed as

\[
P_{se}(\tau) = \frac{\nu(R_e) \times n_{sec}}{\nu(S)}
\]

where \( \nu(\cdot) \) denotes Lebegue measure, \( n_{sec} \) denotes the number of seeds in the second transmission.

We know that \( \nu(R_e) > 0 \), so \( P_{se}(\tau) > 0 \) for any \( \tau \geq 0 \).

In terms of the random Markov course \( \{\xi(\tau)\}^{+\infty}_{\tau=0} \) of GBO, it holds that

\[
\lambda(\tau) = P(\xi(\tau) \in U^* | \xi(\tau - 1) \notin U^*) = P_{me}(\tau) + P_{se}(\tau)
\]

where \( P_{me}(\tau) \) denotes the probability of the seed falling into \( R_e \) by mechanical transmission.

So, \( P(\xi(\tau) \in U^* | \xi(\tau - 1) \notin U^*) \geq P_{se}(\tau) > 0 \).

Hence, because the course \( \{\xi(\tau)\}^{+\infty}_{\tau=0} \) of GBO is an absorbing Markov course, then

\[
P(\xi(\tau) \in U^*) \geq 1 - (1 - P_{mu}(\tau))^\tau.
\]

So, \( \lim_{\tau \to +\infty} P(\xi(\tau) \in U^*) = 1 \).

Consequently, the Markov course of GBO will converge to the optimum condition \( U^* \).

The proof is completed. ■

It should be noted that when solving practical problems, due to the limitations of various factors, it cannot achieve full convergence.

**4. Time complexity of GBO**

Han et al. have finished research on the time complexity of evolutionary algorithm for ant colony optimization and evolutionary programming (Han et al., 2007). Liu et al. have researched on fireworks algorithm (Liu et al., 2014). Some concepts and theorems about the time complexity of GBO algorithm are defined.

**Definition 4.1:** Given the GBO algorithm, an absorbing condition Markov course \( \{\xi(\tau)\}^{+\infty}_{\tau=0} \) and optimum condition space \( U^* \subset U \) is a random value such that if \( \tau = \mu, \xi(\tau) \in U^* \); if \( 0 \leq \tau \leq \mu \), then \( \xi(\tau) \notin U^* \). \( E_\mu \) is the expected first hitting time (EFHT).

**Definition 4.2:** Given the GBO algorithm, an absorbing condition Markov course \( \{\xi(\tau)\}^{+\infty}_{\tau=0} \) and optimum condition space \( U^* \subset U \); let \( \gamma \) be a random nonnegative value, when \( \tau \geq \gamma, P(\xi(\tau + 1) \in U^*) = 1 \); and when \( 0 \leq \tau \leq \gamma, P(\xi(\tau + 1) \in U^*) < 1 \), then the \( \gamma \) is known as the convergence time of GBO. \( E_\gamma \) is the expected convergence time (ECT).

When the expected value \( E_\gamma \) is small, it means that the convergence of GBO algorithm is faster.

The calculation method of \( E_\gamma \) is as follows:

**Theorem 4.1:** Given GBO an absorbing condition, Markov course \( \{\xi(\tau)\}^{+\infty}_{\tau=0} \) and optimum condition space \( U^* \subset U \). If \( \lambda(\tau) = P(\xi(\tau) \in U^*) \) and \( \lim_{\tau \to +\infty} \lambda(\tau) = 1 \), then \( E_\gamma = \sum_{\tau=0}^{+\infty} (1 - \lambda(\tau)) \).

**Proof:**

\[
\lambda(\tau) = P(\xi(\tau) \in U^*) = P[\mu \geq \tau]
\]

\[
\Rightarrow \lambda(\tau) - \lambda(\tau - 1) = P[\mu \leq \tau] - P[\mu \leq \tau - 1]
\]

\[
\Rightarrow P[\mu = \tau] = \lambda(\tau) - \lambda(\tau - 1)
\]

then, \( \sum_{\tau=0}^{+\infty} \tau \cdot P[\mu = \tau] \),

\[
E_\mu = \sum_{\tau=0}^{+\infty} \tau \cdot (\lambda(\tau) - \lambda(\tau - 1)) = \sum_{\tau=1}^{+\infty} (\lambda(\tau) - \lambda(\tau - 1)) + \sum_{\tau=2}^{+\infty} (\lambda(\tau) - \lambda(\tau - 1))
\]
+ \cdots + \sum_{\tau=N}^{+\infty} \lambda(N) - \lambda(\tau - 1) \\
= \sum_{\tau = 1}^{+\infty} \lambda(\tau) - \lambda(1) \\
= \sum_{\tau = 1}^{+\infty} \left( \lim_{t \to +\infty} \lambda(t) - \lambda(\tau - 1) \right) \\
= \sum_{\tau = 0}^{+\infty} (1 - \lambda(\tau)) \\
= \sum_{\tau = 0}^{+\infty} \left( 1 - \lambda(\tau) \right) \\
\therefore E_\gamma = \sum_{\tau = 0}^{+\infty} (1 - \lambda(\tau)).

The proof is completed.

In accordance with Theorem 3.1, it is difficult to compute \( E_\gamma \) because it is hard to acquire the value of \( \lambda(\tau) \). Therefore, its estimation is as follows. The proof of the following theorems can be referred to (Wegener, 2002).

Theorem 4.2: Let \( \mu \) and \( \nu \) be two random variables satisfying \( \mu \geq 0, \nu \geq 0 \). The distribution functions of \( \mu \) and \( \nu \) are \( D_\mu(\cdot) \) and \( D_\nu(\cdot) \). If \( D_\mu(\tau) \geq D_\nu(\tau) \) for any \( \tau \geq 0 \), then \( E_\mu \leq E_\nu \) is true.

Proof: Because \( D_\mu(\tau) = P\{\mu \leq \tau\} \) and \( D_\nu(\tau) = P\{\nu \leq \tau\} \) \( \forall \tau = 0, 1, 2, \cdots \) then,

\[ E_\mu = 0 \cdot D_\mu(0) + \sum_{\tau = 1}^{+\infty} \tau [D_\mu(\tau) - D_\mu(\tau - 1)] \\
= \sum_{\tau = 1}^{+\infty} \sum_{i=1}^{\tau} [D_\mu(\tau) - D_\mu(\tau - 1)] \\
= \sum_{i=0}^{+\infty} \{1 - D_\mu(i)\} \\
E_\mu - E_\nu = \sum_{i=0}^{+\infty} [1 - D_\mu(i)] - \sum_{i=0}^{+\infty} [1 - D_\nu(i)] \\
= \sum_{i=0}^{+\infty} [D_\nu(i) - D_\mu(i)] \leq 0 \\
\Rightarrow E_\mu \leq E_\nu
\]

The proof is completed.

Theorem 4.3: Given GBO an absorbing condition, Markov course \( \{\xi(\tau)\}_{\tau=0}^{+\infty} \) and optimum condition space \( U^* \subset U \), if \( \lambda(\tau) = P\{\xi(\tau) \in U^*\} \) such that \( 0 \leq D_\theta(\tau) \leq \lambda(\tau) \leq D_h(\tau) \leq 1(\forall \tau = 0, 1, 2, \cdots) \) and \( \lim_{\tau \to +\infty} \lambda(\tau) = 1 \), then:

\[ \sum_{i=1}^{\infty} (1 - D_i(\tau)) \leq \sum_{i=1}^{\infty} (1 - D_i(\tau)) \quad (2) \]

Proof: Given two random nonnegative variables, \( h \) and \( l, D_h(t) \) and \( D_l(t) \) denote the distribution functions of \( h \) and \( l \).

Because \( 0 \leq D_i(\tau) \leq \lambda(\tau) \leq D_h(\tau) \leq 1 \), then

\[ E_h \leq E_\mu \leq E_l \Rightarrow \sum_{\tau = 0}^{+\infty} (1 - D_h(\tau)) \leq E_\gamma \]

\[ = \sum_{\tau = 0}^{+\infty} (1 - D_l(\tau)) \]

The proof is completed.

Corollary 4.1: Given GBO an absorbing condition, Markov course \( \{\xi(\tau)\}_{\tau=0}^{+\infty} \) and optimum condition space \( U^* \subset U \); if \( \lambda(\tau) = P\{\xi(\tau) \in U^*\} \) and \( 0 \leq l(\tau) \leq \lambda(\tau) \leq u(\tau) \leq 1(\forall \tau = 0, 1, 2, \cdots) \), then

\[ \sum_{\tau = 0}^{+\infty} \sum_{i=0}^{r} [1 - \lambda(\tau)] \prod_{j=1}^{r} (1 - l(j)) \]

\[ E_\gamma \leq \sum_{\tau = 0}^{+\infty} \sum_{i=0}^{r} \prod_{j=1}^{r} (1 - l(j)) \]

The proof is completed.

Corollary 4.2: Given GBO an absorbing condition, Markov course \( \{\xi(\tau)\}_{\tau=0}^{+\infty} \) and optimum condition space \( U^* \subset U \) and \( \lambda(\tau) = P\{\xi(\tau) \in U^*\} \), if \( l \leq P\{\xi(\tau) \in U^*\} \leq u \) \( (\forall \tau = 0, 1, 2, \cdots) \), then \( \lim_{\tau \to +\infty} \lambda(\tau) = 1 \), then \( E_\gamma \) of GBO as follows:

\[ u^{-1} (1 - \lambda(0)) \leq E_\gamma \leq l^{-1} (1 - \lambda(0)) \]
Proof: From Theorem 3.5, 

\[
E_Y \leq (1 - \lambda(0)) \left( I + \sum_{r=2}^{+\infty} t \prod_{i=0}^{r-2} (1 - \lambda) \right)
\]

\[
\Rightarrow E_Y \leq (1 - \lambda(0)) \left( I + \sum_{r=2}^{+\infty} t l_l(t - 1)^{r - 1} \right)
\]

\[
\Rightarrow E_Y \leq l(1 - \lambda(0)) \left( \sum_{r=0}^{+\infty} \tau (1 - \lambda)^{r} + \sum_{r=0}^{+\infty} (1 - \lambda)^{r} \right)
\]

\[
\Rightarrow E_Y \leq l(1 - \lambda(0)) \left( \frac{1 - l}{l} + \frac{1}{l} \right)
\]

\[
= \frac{1}{l}(1 - \lambda(0))
\]

In the same way, \(E_Y \geq u^{-1}(1 - \lambda(0))\), then \(u^{-1}(1 - \lambda(0)) \leq E_Y \leq l^{-1}(1 - \lambda(0))\).

The proof is completed. 

The time complexity of GBO algorithm needs to calculate \(E_Y\). Based on the above corollary, the expression \(P[\xi(t) \in U^n | \xi(t - 1) \notin U^n]\) represents the probability that the seed of GBO algorithm will find the global optimal solution from the non-optimum condition. The estimation of expected convergence time \(E_Y\) can be estimated from the range of the expression \(P[\xi(t) \in U^n | \xi(t - 1) \notin U^n]\). GBO includes two important operations: a mechanical propagator and a second propagator. In this section, the equation is further analyzed to obtain the time complexity of GBO.

**Theorem 4.4:** Let GBO’s Markov course \(\{\xi(t)\}_{t=0}^{+\infty}\) and optimum condition space \(U^n \subset U\), then GBO is such that

\[
\frac{\nu(R_i) \times n_{sec}}{\nu(S)} \leq P[\xi(t + 1) \in U^n | \xi(t) \notin U^n] \leq \nu(R_i)
\]

\[
\times \left( \frac{n_{sec}}{\nu(S)} + \sum_{i=1}^{n} \frac{z_i}{\nu(L_i)} \right)
\]

(4)

Where \(\nu(R_i), \nu(S), \nu(L_i)\) are the Lebesgue measure values of \(R_i, S\) and \(L_i\), respectively. \(L_i\) is the seed diffusion range of \(i\)-th parent.

**Proof:** In terms of the steps of GBO, it includes two operations to generate the seeds: a mechanical propagator and a second propagator. So the following equation is obtained:

\[
P[\xi(t + 1) \in U^n | \xi(t) \notin U^n] = \frac{\nu(R_i) \times n_{sec}}{\nu(S)} + P(mec)
\]

(5)

where, \(P(mec)\) represents the probability that the seeds generated by all parents fall into the optimum region through the mechanical propagation operator, and the expression for \(P(mec)\) is as follows:

\[
P(mec) = \sum_{i=1}^{n} \frac{\nu(G_i \cap R_i \times z_i)}{\nu(G_i)}
\]

(6)

where, \(L_i\) denotes the diffusion range of seeds produced by the \(i\)-th parent; \(z_i\) is the number of seeds that the \(i\)-th parent generates.

Because \(0 \leq \nu(L_i \cap R_i) \leq \nu(R_i)\),

\[
0 \leq P(mec) = \sum_{i=1}^{n} \frac{\nu(R_i) \times z_i}{\nu(L_i)}
\]

\[
\leq \sum_{i=1}^{n} \frac{\nu(R_i \times z_i)}{\nu(L_i)}
\]

\[
= \nu(R_i) \sum_{i=1}^{n} \frac{z_i}{\nu(L_i)}
\]

and then,

\[
\frac{\nu(R_i \times n_{sec}}{\nu(S)} \leq P[\xi(t + 1) \in Y^n | \xi(t) \notin Y^n] \leq \nu(R_i)
\]

\[
\times \left( \frac{n_{sec}}{\nu(S)} + \sum_{i=1}^{n} \frac{z_i}{\nu(L_i)} \right)
\]

(4)

The proof is completed. 

Because the actual formula is difficult to calculate, the above theorem gives a rude result. To make the calculation result more accurate, the formula of \(P(mec)\) can be changed as follows:

\[
P(mec) = \sum_{i=1}^{n} \frac{\nu(G_i \cap R_i \times z_i)}{\nu(G_i)}
\]

(7)

As we know, the formulas \(\nu(G_i \cap R_i)\) and \(z_i\) in Equation (7) play a central role because they change dynamically as the algorithm runs, and the formula \(\nu(G_i \cap R_i)\) is related to the parent location.

According to the selection strategy of GBO, seeds with better fitness are more likely to enter the next population, so it can be assumed that only one parent can stay in the optimum region \(R_i\) in the same time, and there are the highest odds for the best seed to fall into the optimum region \(R_i\).

According to the design rules of GBO algorithm, the parent with the optimal adaptive value in the population can reproduce the largest number of seeds, and the mechanical transmission distance of these seeds is
also the longest. So it follows that \( v(L_i) \leq v(L_{\text{best}}), Z_i \leq Z_{\text{best}}, i \in (1, 2, \ldots, n), \) then
\[
\frac{v(L_i \cap R_e) \times Z_i}{v(L_i)} \leq \frac{v(L_{\text{best}} \cap R_e) \times Z_{\text{best}}}{v(L_{\text{best}})}.
\tag{8}
\]

Consider that at the beginning of the algorithm iteration there exists \((L_i \cap R_e) \cap (L_{\text{best}} \cap R_e) = \emptyset, i \in (1, 2, \ldots, n)\) and \(i \neq \text{best}, \) then
\[
P(\text{mec}) = \sum_{i=1}^{n} \frac{v(G_i \cap R_e) \times Z_i}{v(G_i)} < \frac{v(G_{\text{best}} \cap R_e) \times Z_{\text{best}}}{v(G_{\text{best}})} < \frac{v(G_{\text{best}}) \times Z_{\text{best}}}{v(G_{\text{best}})}.
\tag{9}
\]

The variation of Equation (7) is as follows:
\[
\frac{v(R_e) \times n_{\sec}}{v(S)} \leq P(\xi(\tau + 1) \in U^n | \xi(\tau) \notin U^n)
\]
\[
= v(R_e) \left(\frac{n_{\sec}}{v(S)} + \frac{Z_{\text{best}}}{v(G_{\text{best}})}\right).
\tag{10}
\]

According to Corollary 2.1, let \(l = \frac{v(R_e) \times n_{\sec}}{v(S)}, u = v(R_e)\)
\[
\left(\frac{n_{\sec}}{v(S)} + \frac{Z_{\text{best}}}{v(G_{\text{best}})}\right),
\]
then
\[
\frac{v(G_{\text{best}}) \times v(S)}{v(R_e) \times (n_{\sec} \times v(G_{\text{best}}) + Z_{\text{best}} \times v(S))} \times (1 - \lambda(0))
\]
\[
\leq E_y \leq \frac{v(S)}{v(R_e) \times n_{\sec}} \times (1 - \lambda(0)).
\tag{11}
\]

The initial population generated by GBO can be randomly distributed in the entire search space, so \(\lambda(\tau) = P(\xi(\tau) \in Y^*)\) and \(\lambda(0) = P(\xi(0) \in Y^*) \ll 1, \lambda(0) = 1, \) thus:
\[
\frac{v(G_{\text{best}}) \times v(S)}{v(R_e) \times (n \times v(G_{\text{best}}) + Z_{\text{best}} \times v(S))} \leq E_y
\]
\[
\leq \frac{v(S)}{v(R_e) \times n_{\sec}}.
\]

**Corollary 4.3:** GBO algorithm’s ECT \(E_y\), such that
\[
\frac{v(G_{\text{best}}) \times v(S)}{v(R_e) \times (n \times v(G_{\text{best}}) + Z_{\text{best}} \times v(S))} \leq E_y
\]
\[
\leq \frac{v(S)}{v(R_e) \times n_{\sec}}
\tag{12}
\]

It can be seen from this that the ECT of GBO algorithm is not only related to \(S\), but also to population size \(n\), the number of secondary propagation seeds \(n_{\sec}\), especially to the optimal individuals. It should be noted that the above results are obtained under certain assumptions. Therefore, to propose a more accurate analysis method, some equations of GBO need to be analyzed in detail.

### 5. Conclusion

The GBO algorithm is a novel swarm intelligence optimization algorithm. Compared with the classical evolutionary algorithm, this algorithm has its own advantages in searching for a target space through seed propagation. In this paper, the risk of convergence is also carried on the preliminary theoretical analysis, using the algorithm of swarm intelligence and analysis of other same Markov course; at the same time, the cluster algorithm convergence theorem is given. The basic concept of the Markov random process is defined and the global convergence of the algorithm is proved. In addition, the expected convergence time of the impatiens GBO algorithm in an approximate region is provided in this paper.

### Disclosure statement

No potential conflict of interest was reported by the author(s).

### Funding

This work is partially supported by Key specialized research and development breakthrough of Henan province [22210232 0456], National Key R&D Program of China [2018YFB1308800].

### Data availability statement

Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

### ORCID

Shengpu Li  
http://orcid.org/0000-0002-7569-3679

### References

Chen, T., Tang, K., Chen, G., & Yao, X. (2010). Analysis of computational time of simple estimation of distribution algorithms. *IEEE Transactions on Evolutionary Computation*, 14(1), 1–22. https://doi.org/10.1109/TEVC.2009.2040019

Ding, L., & Yu, J. (2012). Some techniques for analyzing time complexity of evolutionary algorithms. *Transactions of the Institute of Measurement and Control*, 34(4), 755–766. https://doi.org/10.1177/0142331211402914

Doerr, B., Johannsen, D., & Winzen, C. (2012). Multiplicative drift analysis. *Algorithmica*, 64(4), 673–697. https://doi.org/10.1007/s00453-012-9622-x

Droste, S., Jansen, T., & Wegener, I. (2002). On the analysis of the \((1+1)\) evolutionary algorithm. *Theoretical Computer Science*, 276(2002), 51–81.

Han, H., Hao, Z., & Qin, Y. (2008). Time complexity of evolutionary programming. *Journal of Computer Research and Development*, 45(11), 1850–1857.

Han, H., Hao, Z., Wu, C., & Qin, Y. (2007). Time convergence speed of Ant colony optimization. *Chinese Journal of Computer*, 30(8), 1344–1353.

He, J., & Yao, X. (2001). Drift analysis and average time complexity of evolutionary algorithms. *Artificial Intelligence*, 127(1), 57–85. https://doi.org/10.1016/S0004-3702(01)00058-3
He, J., & Yao, X. (2002). From an individual to a population: An analysis of the first hitting time of population-based evolutionary algorithms. *IEEE Transactions on Evolutionary Computation, 6*(5), 495–511. https://doi.org/10.1109/TEVC.2002.800886

He, J., & Yao, X. (2003). Towards an analytic framework for analysing the computation time of evolutionary algorithms. *Artificial Intelligence, 145*(1–2), 59–97. https://doi.org/10.1016/S0004-3702(02)00381-8

Huang, H., Wu, C., & Hao, Z. (2009). A pheromone-rate-based analysis on the convergence time of ACO algorithm. *IEEE Transactions on Systems, Man, and Cybernetics. Part B: Cybernetics, 39*(4), 910–923. https://doi.org/10.1109/TSMCB.2009.2012867

Lai, X., Zhou, Y., He, J., & Zhang, J. (2014). Performance analysis on evolutionary algorithms for the minimum label spanning tree problem. *IEEE Transactions on Evolutionary Computation, 18*(6), 860–872. https://doi.org/10.1109/TEVC.2013.2291790

Lehre, P. K., & Yao, X. (2014). Runtime analysis of the (1 + 1) EA on computing unique input output sequences. *Information Sciences, 259*(2014), 510–531. https://doi.org/10.1016/j.ins.2010.01.031

Li, S., & Sun, Y. (2020). Garden balsam optimization algorithm. *Concurrency & Computation Practice & Experience, 32*(2). https://doi.org/10.1002/cpe.5456

Li, S., & Sun, Y. (2020). A novel numerical optimization algorithm inspired from garden balsam. *Neural Computing & Applications, 32*(22), 16783–16794. https://doi.org/10.1007/s00521-018-3905-3

Li, S. P., Sun, Y. Z., & Wang, X. (2019). Speech emotion recognition using hybrid GBO algorithm-based ANFIS. *Indian Journal of Pharmaceutical Sciences, 81*(1), S52–S53.

Liu, J., Zheng, S., & Tan, Y. (2014, July 6–11). Analysis on global convergence and time complexity of fireworks algorithm. 2014 IEEE Congress on evolutionary computation, IEEE, 3207–3214. 2014 IEEE Congress on Evolutionary Computation (CEC), Beijing, China.

Oliveto, P. S., He, J., & Yao, X. (2007). Time complexity of evolutionary algorithms for combinatorial optimization: A decade of results. *International Journal of Automation and Computing, 4*(3), 281–293. https://doi.org/10.1007/s11633-007-0281-3

Oliveto, P. S., He, J., & Yao, X. (2009). Analysis of the (1 + 1)-EA for finding approximate solutions to vertex cover problems. *IEEE Transactions on Evolutionary Computation, 13*(5), 1006–1029. https://doi.org/10.1109/TEVC.2009.2014362

Sudholt, D. (2013). A new method for lower bounds on the running time of evolutionary algorithms. *IEEE Transactions on Evolutionary Computation, 17*(3), 418–435. https://doi.org/10.1109/TEVC.2012.2202241

Wegener, I. (2002). Methods for the analysis of evolutionary algorithms on pseudo-Boolean functions. In R. Sarker, X. Yao, & M. Mohammadian (Eds.), *Evolutionary optimization* (pp. 349–369). Kluwer.

Xia, X., Zhou, Y., & Lai, X. (2015). On the analysis of the (1 + 1) evolutionary algorithm for the maximum leaf spanning tree problem. *International Journal of Computer Mathematics, 92*(9–10), 2023–2035. https://doi.org/10.1080/00207160.2014.964695

Yang, Y., & Zhou, Z. (2008). A new approach to estimating the expected first hitting time of evolutionary algorithms. *Artificial Intelligence, 172*(15), 1809–1832. https://doi.org/10.1016/j.artint.2008.07.001

Yi, S., Chen, M., & Zeng, Z. (2011). Convergence analysis on a class of quantum-inspired evolutionary algorithms, seventh IEEE International Conference on natural computation. *ICNC, 2011*(2), 1072–1076. https://doi.org/10.1109/icnc.2011.6022161

Yu, Y., Qian, C., & Zhou, Z. H. (2015). Switch analysis for running time analysis of evolutionary algorithms. *IEEE Transactions on Evolutionary Computation, 19*(6), 777–792. https://doi.org/10.1109/TEVC.2014.2378891

Yu, Y., & Zhou, Z. H. (2008). A new approach to estimating the expected first hitting time of evolutionary algorithms. *Artificial Intelligence, 172*(15), 1809–1832. https://doi.org/10.1016/j.artint.2008.07.001

Zhou, Y., Lai, X., & Li, K. (2015). Approximation and parameterized runtime analysis of evolutionary algorithms for the maximum cut problem. *IEEE Transactions on Cybernetics, 45*(8), 1491–1498. https://doi.org/10.1109/TCYB.2014.2354343