Improvement of Artificial Bee Colony Algorithm Based on Self-Adaptive Random Optimization Strategy

Xin LIU*, Xiao-peng YANG and Zi-xuan SU

Institute of Information and Navigation, Air Force Engineering University, Xi’an 710077, China

*Corresponding author

Keywords: Swarm intelligence, Artificial Bee Colony (ABC), Bidirectional Random Optimization (BRO), Self-adaptive, Particle Swarm Optimization (PSO).

Abstract. In order to overcome the disadvantages that traditional ABC algorithm is inclined to fall into local optima and it has a low searching speed either, an improved ABC algorithm based on SRABC was proposed. Firstly, the improved algorithm was derived from the self-adaptive method to update the new location of ABC so as to improve the correlation within the bee colony. Secondly, BRO mechanism was used to restrain the direction of searching for fitness function in order to improve local searching ability. On the other hand, PSO algorithm was introduced at the initial stage of the improved ABC algorithm to increase the convergence rate. Finally, the simulation results in three benchmark functions show that the proposed algorithm has obviously better performance in search ability and convergence rate.

Introduction

In recent years, there is a new random search method in the optimization field, namely the ABC algorithm[1]. However, it also has disadvantages, such as the low convergence rate and inclination to “early mature”. In order to improve the convergence rate of ABC algorithm and solve the inclination into local optima in the later period, an improved algorithm based on the self-adaptive random optimization strategy was proposed to improve the local search ability of the algorithm by taking advantage of self-adaptive thought and bidirectional random optimization mechanism. Meanwhile, PSO was introduced at the initial stage of the improved algorithm to increase its convergence rate.

Principle of ABC Algorithm

In the ABC algorithm, artificial bee colony falls into three categories according to the labor distribution, namely honey-gather bee, observation bee and investigation bee[2]-[5]. In the algorithm, in order to generate a new candidate position $V_i$ according to the memorized position $X_i$, the following would be used for updating: $v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj})$.

In which, $k$ is a nectar source different from $i$, $j$ is the subscript of randomly selected nectar source, $\phi_{ij}$ is a random number between [-1,1]. According to the amount of honey in the nectar source, the probability for a nectar source to be selected by the observation bee was $P_i = \frac{fit(\theta_i)}{\sum_{j=1}^{S} fit(\theta_j)}$.

In which $S$ is the total number of nectar source, $\theta_i$ is the $i^{th}$ nectar source, $f(\theta_i)$ is the fitness of the nectar source at $\theta_i$, $i \in \{1,2,\ldots,S\}$.

Suppose after “limit” times of cycling search and update, the fitness of nectar source could still not be improved, then it would be given up and the honey-gather bee would turn to be investigation bee[6]. “Limit” was an important control parameter in ABC algorithm for the selection of investigation bee[7]. The procedures for the investigation bee funding a new position and replacing $X_i$ were shown as follows: $x_i^{l+1} = x_{\min}^{l} + rand(0,1)(x_{\max}^{l} - x_{\min}^{l})$. 

364
Improvement for the ABC Algorithm

Position-update Equation for the Self-adaptive Bee Colony
The best method for overall search was to adopt larger $\phi_{ij}$ at the initial stage of the algorithm so as to obtain excellent nectar source with higher search ability and improve the search precision, while in the latter period, smaller $\phi_{ij}$ was needed to improve the local search ability of the algorithm and speed up its convergence rate. Therefore, $\phi_{ij}$ was set as the function for iterations, and it would decrease with the increase in iterations. The $\phi_{ij}$ was defined as follows: $\phi_{ij}^k = \phi_{ij}^{k-1} - \frac{C(w_{\text{max}} - w_{\text{min}})}{C_{\text{max}}}$. 

In which $w_{\text{max}}$ and $w_{\text{min}}$ were initial and final weight respectively, $C_{\text{max}}$ was the maximum iterations, $C$ was the current iterations. As a result, the following would be used for updating: $v_i = x_i + \phi_{ij}^k (x_i - x_g)$. Thus to some degree, it played a guiding role in the search trend for the position of nectar source, overcoming the disadvantages such as strong randomness and slow convergence rate.

Bidirectional Random Optimization Mechanism
When calculating the fitness of nectar source with ABC algorithm, the observation bee would select a nectar source after comparing the ones around $\theta_i$, and the calculation for the position near the nectar source was shown as follows: $\theta_i (C+1) = \theta_i (C) + \phi_i (C)$. 

In which $\phi_i (c)$ was the progressive step length produced randomly near $\theta_i$. If after the calculation for fitness, $\text{fit}(C+1) > \text{fit}(C)$, then the observation bee would choose $\theta_i (C+1)$, or it would remain unchanged. If it wasn’t improved after finite cycling times, then it should be given up, and the honey-gather bee would turn to be investigation bee. There were certain disadvantages in the above stated method, namely, in each cycle, the nectar source in single direction would be searched, and as a result, it would incline to fall into local optima. In reference [8], a bidirectional random optimization mechanism was proposed in the process of the research on search hit rate and success rate under dynamic network environment, which effectively improved the search feature of the network. Inspired by this thought, the improved mechanism was introduced to improve the direction for the search of nectar source in this paper, if $\text{fit}(\theta + l) < \text{fit}(\theta)$, then $\theta_i = \theta_i + d$. If $\text{fit}(\theta - l) < \text{fit}(\theta)$, then $\theta_i = \theta_i - d$. Or $\theta_i$ would remain unchanged.

Algorithm Initialization Realized by the Particle Swarm Optimization Algorithm
The convergence rate was slow in the bee colony algorithm, while it was relatively faster in the particle swarm optimization algorithm, which was introduced into the initial stage for improving the algorithm for its fast convergence rate, namely the overall optimal solution would be obtained from the iterations by taking advantage of particle swarm optimization, and then the position of nectar source would be generated randomly near the optimal solution, later conduct optimizing process for calculating the position of nectar source with artificial bee colony. In which, the improved initial position of nectar source with ABC algorithm was shown as follows: $X_i = P_{g,\text{best}} + \phi_i^M \cdot P_{g,\text{best}}$. 

In which $P_{g,\text{best}}$ is $M$-dimensional vector, and each element equals $P_{g,\text{best}}$, $\phi_i^M$ was the $M$-dimensional vector among [-1,1] produced randomly. According to the above mentioned strategy, the specific procedures for improving the algorithm were shown as follows:

Step one: set related initial parameters for artificial bee colony and particle swarm optimization algorithm, the initial speed and position of $M$ particles were generated randomly.

Step two: determine the optimal solution $P_{g,\text{best}}$ within the cycling times $c$ through calculating the fitness value of each particle for comparison.

Step three: honey-gather bee searched a new nectar source and calculated its fitness, if it was better than the original position, then replace the original position with the new one.
Step four: the observation bee selected a nectar position according to the amount of honey in the nectar source, and generated a new position according to the bilateral random optimization mechanism and evaluated this position.

Step five: if the nectar source was given up, then the honey-gather bee at this nectar source would turn to be investigation bee. And keep down the current optimal position and fitness value.

**Analysis of Simulation Experiment**

In order to testify the effectiveness of algorithm and conduct related performance analysis, three benchmark functions were selected for comparison and test, which was different from the traditional ABC algorithm. The comparison for the performances of hybrid artificial bee colony was proposed in reference [1].

1). Rastrigin function \( f(x) = \left( \sum_{i=1}^{n} x_i^2 - 10 \cos(2\pi x_i) + 10 \right) \) is a multimodal function, whose optimal solutions distribute evenly, with the search scope within \([-20, 20]\) and the overall optimal solution 0.

2). Griewank function \( f(x) = \frac{1}{4000} \left( \sum_{i=1}^{n} x_i^2 \right) - \left( \prod_{i=1}^{n} \cos \left( \frac{x_i}{\sqrt{i}} \right) \right) + 1 \) is a multimodal function, whose optimal solutions distribute evenly. However its local optima increase with the increase in dimensions.

3). Sphere function \( f(x) = \sum_{i=1}^{n} x_i^2 \) is a continuous convex function with single peak, whose search scope is within \([-100, 100]\) and the lowest point of the function is 0.

The parameters for the algorithm in this paper was set as follows: as for the improved SRABC algorithm, the scale of the colony \( S=60, \) limit=60, the number of particle swarm optimization is 60, the cycling times of particle swarm optimization algorithm \( c=100; \) As for the colony scale of the standard ABC algorithm \( S=60, \) limit=60, the maximum iteration is 2500, and the algorithm would operate for 30 times independently. The comparison result for the maximum, minimum, average value and variance obtained after the independent operation of various functions in different dimensions for 30 times is shown in table one to three.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Dimension</th>
<th>Average value</th>
<th>Variance</th>
<th>Maximum value</th>
<th>Minimum value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC</td>
<td>30</td>
<td>1.13714E-015</td>
<td>2.97144E-016</td>
<td>1.79207E-015</td>
<td>5.46909E-016</td>
</tr>
<tr>
<td>SRABC</td>
<td>30</td>
<td>2.95155E-016</td>
<td>5.79213E-17</td>
<td>4.61564E-016</td>
<td>1.80046E-016</td>
</tr>
<tr>
<td>ABC</td>
<td>60</td>
<td>2.34037E-014</td>
<td>1.82907E-014</td>
<td>7.64184E-014</td>
<td>4.13668E-015</td>
</tr>
<tr>
<td>HABC</td>
<td>60</td>
<td>3.6088E-015</td>
<td>1.45665E-015</td>
<td>8.65843E-015</td>
<td>1.87149E-015</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Dimension</th>
<th>Average value</th>
<th>Variance</th>
<th>Maximum value</th>
<th>Minimum value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC</td>
<td>30</td>
<td>2.58239E-009</td>
<td>1.03788E-008</td>
<td>5.65922E-008</td>
<td>1.13687E-012</td>
</tr>
<tr>
<td>HABC</td>
<td>30</td>
<td>1.13687E-012</td>
<td>1.64169E-012</td>
<td>8.15703E-012</td>
<td>1.12578E-015</td>
</tr>
<tr>
<td>SRABC</td>
<td>30</td>
<td>1.32635E-014</td>
<td>1.78719E-14</td>
<td>5.68434E-014</td>
<td>0</td>
</tr>
<tr>
<td>ABC</td>
<td>60</td>
<td>0.167895</td>
<td>0.458241</td>
<td>1.99033</td>
<td>1.72463E-010</td>
</tr>
<tr>
<td>HABC</td>
<td>60</td>
<td>4.56788E-008</td>
<td>1.53871E-007</td>
<td>8.25186E-007</td>
<td>1.3074E-011</td>
</tr>
<tr>
<td>SRABC</td>
<td>60</td>
<td>121393E-010</td>
<td>2.62695E-010</td>
<td>1.2559E-009</td>
<td>1.7053E-013</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Dimension</th>
<th>Average value</th>
<th>Variance</th>
<th>Maximum value</th>
<th>Minimum value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRABC</td>
<td>30</td>
<td>6.25426E016</td>
<td>1.41953E-015</td>
<td>7.54952E-015</td>
<td>0</td>
</tr>
<tr>
<td>ABC</td>
<td>60</td>
<td>2.79554E-013</td>
<td>8.89572E-013</td>
<td>4.85578E-012</td>
<td>2.77556E-015</td>
</tr>
<tr>
<td>HABC</td>
<td>60</td>
<td>1.4011E-013</td>
<td>2.87342E-013</td>
<td>1.20581E-012</td>
<td>2.33147E-015</td>
</tr>
<tr>
<td>SRABC</td>
<td>60</td>
<td>1.72825E-015</td>
<td>1.94662E-015</td>
<td>8.54872E-015</td>
<td>1.11022E-016</td>
</tr>
</tbody>
</table>
It could be seen from table 1 that, for Sphere function with single peak, although the optimization result for this function under different dimensions was not improved greatly with SRABC algorithm, it was still better than that with standard ABC algorithm and HABC algorithm. In multimodal function Griewank and Rastrigin, there were complicated non-linear overall optimization problems. It could be known from table 2 and 3 that the precision of the simulation results of the two functions under different dimensions with SRABC algorithm is better than that with ABC and HABC algorithm. Especially for the 60-dimensional Rastrigin function could be converged to 0 rapidly. Thus it could be seen that the improved algorithm not only kept the feature of its original algorithm, but also improved the calculating precision and stability when compared to the traditional algorithm and HABC algorithm.

Figure 1. Comparison of different dimensions for Sphere function.

Figure 2. Comparison of different dimensions for Rastrigin function.

Figure 3. Comparison of different dimensions for Griewank function.
It could be seen from figure 1 to 3 that, under the same iteration times and targeting at the different dimensions of three benchmark functions, the iteration of optima nearly stopped in the later period, while there was certain improvement in the HABC algorithm when comparing to the ABC algorithm, and the performance of SRABC was improved greatly in the optimization process of SRABC algorithm. It could be seen from figure 1 that the SRABC algorithm of Sphere function could converge swiftly and obtain optima. Since the optimization result at the initial stage of algorithm with particle swarm optimization algorithm was regarded as the initial vale of SRABC algorithm, the search space was reduced greatly and the convergence rate of the algorithm was further improved. It would be seen from figure 2 and 3 that under certain iteration times, the optima of SRABC algorithm could be achieved in the early stage of Griewank and Rastrigin function which nearly decreased linearly, and the convergence rate was smaller than the other two algorithms.

It could be known that the self-adaptive random optimization strategy for bee colony could decrease the convergence of optimal in the whole optimization process, which could guide individuals to overall optima. There were great improvements in the in the precision and convergence rate of simulation results when comparing to ABC and SRABC algorithm, which also prevented the early mature of algorithm.

Conclusion

As a novel swarm intelligent optimization algorithm, artificial bee colony algorithm was characterized with easy realization, simple operation, little control parameters. Targeting at the weak local search ability, low search precision and slow convergence rate of ABC algorithm, particle swarm optimization algorithm was introduced at the initial stage of the algorithm to initialize the bee colony. Meanwhile, on the basis of self-adaptive thought and bidirectional random optimization mechanism, improved algorithm based on the bidirectional random optimization strategy was proposed to make up for the disadvantages such as strong randomness and single search direction in the process of optimization effectively, which avoided the inclination in falling into the local optima of the algorithm in certain degree. Through comparing traditional algorithms, the HABC algorithm proposed in reference [1] and the test of improved algorithm under three different benchmark functions showed that the algorithm in this paper was effective generally, which improved the optimization ability of the algorithm on the basis of raising the convergence rate of the algorithm.

References


