Journal of Physics: Conference Series

PAPER • **OPEN ACCESS**

A Group Learning based Optimization Algorithm Applied to UWB Positioning

To cite this article: XiuWei Zhang *et al* 2022 *J. Phys.: Conf. Ser.* **2294** 012001

A Group Learning based Optimization Algorithm Applied to UWB Positioning

XiuWei Zhang[1], **WenYi Zheng**[1], **Yong chen**[1*]

[1] School of Computer and Information Science, Chongqing Normal University, Chongqing, 401331, China

[*] Corresponding author's e-mail: 1871620018@qq.com

Abstract. In dealing with optimization problems, metaheuristic algorithms have attracted much attention due to their simple structure and flexible characteristics. Inspired by the principle of teaching students in accordance with their aptitude, this paper proposed a novel metaheuristic algorithm——Group Learning based Optimization (GLBO) Algorithm, this algorithm is suitable for continuous optimization problems. The main idea of this method is to divide a class into three study groups according to their scores, and formulate different study strategies for them according to the characteristics of each group, so as to improve the scores of the whole class. To verify the performance of the algorithm, it is tested on the CEC21 Benchmark suit and applied to UWB positioning. The results show that the proposed method has excellent performance when dealing with continuous optimization problems.

1. Introduction

In real life, there are many problems that can be transformed into optimization problems to be solved by optimization algorithms. However, many real-life optimization problems are complex, and therefore, development of efficient optimization algorithms is one of the important research areas in evolutionary computing. Since the genetic algorithm was proposed, many scientific researchers have successively proposed a series of new optimization algorithms, such as differential evolution algorithm (DE)[1], Particle Swarm Optimization (PSO)[2], Ant Colony Optimization(ACO)[3], Cuckoo Search (CS)[4] and Firefly Algorithm (FA)[5], these are some well-known algorithms. There are also many scholars who have improved these classic algorithms to improve their performance in dealing with optimization problems, such as AGSK[6], MadDE[7] and IMODE[8], these are some recent modifications to these classic algorithms by some scholars.

Ultra-Wideband (UWB) technology is an impulse radio technology with low power consumption and good ability to pass obstacles, enabling it to achieve high-precision ranging in indoor areas. UWB positioning is also regarded as one of the best solutions for high-precision positioning requirements and is widely used in various fields. Traditional UWB positioning algorithms include Chan algorithm, Taylor positioning algorithm, Kalman filter and other methods. Recent studies have found that the UWB localization problem can be transformed into an optimization problem and solved by optimization algorithm.

2. Group learning based optimization algorithm

Teaching students according to their aptitude is an educational principle that originated in ancient China. The talents and interests of each student are take into account in this principle, and adopts targeted educational methods according to the specific conditions of different students. Inspired by this educational principle, the GLBO algorithm is proposed. This algorithm treats a population as a class, and the individuals in the population are the students in the class. The GLBO algorithm divides the students in the class into three groups according to their scores (fitness value). The first group is students with good scores, the second group is students with average scores, and the third group is students with the worst scores in the class. Generally, students with good scores have stronger learning ability, and students with poor scores have weaker learning ability. If students with strong learning ability and those with weak learning ability are put together to study, not only the students with poor learning ability cannot keep up with the learning progress, but also the learning progress of students with strong learning ability will be affected. Therefore, the GLBO algorithm formulates different learning strategies for different groups. The GLBO algorithm consists of a population initialization phase, a group learning phase and a selection phase.

2.1. Initial population

The GLBO algorithm is a population-based evolutionary algorithm, in which the population can be understood as a class, and the individuals in the population are the students in the class. In a D-dimensional optimization problem, each student is a $1*D$ array, representing a solution to the optimization problem, which is defined as: $student = [x_{1,}, x_2, ..., x_D]$. By substituting this student into the fitness function, the student's scores, that is, the fitness value, can be calculated. $Fitness = f(student) = f(x_1, x_2, ..., x_D)$. A class consists of NP students, so a class can be represented by an *NP*D* matrix. $Class = [student_1; student_2; ...; student_{NP}]$. In the initialization phase, each student in the class is randomly assigned by the equation (1), where $i \in (1, 2, ..., NP),\ \ j \in (1, 2, ..., D)$, *xl* and *xu* denote the lower and upper bounds of the individual in each dimension, respectively.

$$student_{i,j} = xl + rand \times (xu - xl) \tag{1}$$

At this point, the initialization phase is completed, and the next step is the group learning phase.

2.2. Group Learning

Before the start of this phase, an individual with the best fitness should be selected as a teacher in the class, and then the students will be divided into three study groups according to their scores. The number of students in the first and third groups is *NP/4* in each group, and the number of students in the second group is *NP/2*. The reason for this division is that most of the students are ordinary people with average learning ability, only a small number of outstanding students with strong learning ability, and only a small number of students with poor learning ability.

For the first group of students, their learning ability is relatively strong. In addition to the teacher's guidance, they only need to randomly select a classmate in the group as a comparison, and learn from each other's strengths. As shown in equation (2), a random number α is required to control the learning ability, and the larger the α, the stronger the learning ability. $G1_i$ and $G1_{rand}$ denote the *i-th* student and a randomly selected student in the first group, respectively.

$$newG1_i = G1_i + \alpha \times [(Teacher - G1_i) + (G1_i - G1_{rand})] \tag{2}$$

The second group of students, they have a certain learning ability. They need the guidance of the teacher, and they need to choose a classmate from the first group and the third group of students as a reference to summarize their learning experience to improve their scores and avoid detours. In addition, they also need to randomly select a classmate in the group as a competitor to motivate themselves to learn. As shown in equation (3), a random number α is also required to control the learning ability. $G2_i$ and $G2_{rand}$ denote the *i-th* student and a randomly selected student in the second group, respectively, $G3_{rand}$ represents a student randomly selected from the third group.

$$newG2_i = G2_i + \alpha \times [(Teacher - G2_i) + (G1_i - G3_{rand}) + (G2_i - G2_{rand})] \tag{3}$$

The third group of students is the students with poor learning ability in the class. Only relying on the teacher's guidance can no longer improve their scores. They also need the first group of students to help them learn one-on-one. As shown in equation (4), a random number α is also required to control the learning ability. $G3_i$ represents the *i-th* student in the third group.

$$newG3_i = G3_i + \alpha \times [(Teacher - G3_i) + (G1_{rand} - G3_i)] \tag{4}$$

In addition, an extracurricular study period is arranged for students in all groups, and students' knowledge reserves can be improved by reading extracurricular books, thereby improving their scores. At this phase, the knowledge acquired by students is random, and the improvement of students' performance is also random. The specific operation of this step in this algorithm is to randomly select n dimensions for each individual student, and assign a new random value to these n dimensions according to equation (1).

2.3. Selection

After the group learning phase, each group moves to a selection phase. In the group learning stage, each student generates new individuals $newG1_i$, $newG2_i$ and $newG3_i$ through learning, compares these new individuals with the original individuals, and selects individuals with better fitness to form a new population and enter the next iteration, as shown in equation (5).

$$nextGn_i = \begin{cases} newGn_i & if\ f(newGn_i) < f(Gn_i) \\ Gn_i & otherwise \end{cases} ,n \in [1,2,3], i \in [1,2,...,Group\ size] \tag{5}$$

At the end of one iteration, the three groups need to be combined into one class, as they will be regrouped according to their latest grades at the beginning of the next iteration.

3. Comparison of GLBO with other optimization algorithms

The CEC21 Benchmark suit is an objective function suit designed to test the performance of optimization algorithm. It includes 10 test functions, all of which are minimal optimization problems. In order to verify the performance of the proposed algorithm, GLBO was tested on the CEC21 Benchmark suit (Bias, Shift, Rotation) and compared with the latest excellent algorithm.

Tables 1 and 2 show the performance of GLBO and several current state-of-the-art evolutionary algorithms on ten 10-dimensional test functions and 20-dimensional test functions, respectively. In the 10-dimensional function test, GLBO outperforms the other algorithms on 7 test functions, for comparison, AGSK and MadDE outperform the other algorithms on 4 test functions, while IMODE only performs on 1 test function optimal. In the 20-dimensional function test, GLBO outperforms the other algorithms on 3 test functions, while ASGK MadDE outperforms the other algorithms on 3 and 4 test functions, respectively, and IMODE only performs on 2 test function optimal. Experiments show that the proposed GLBO has excellent performance when dealing with optimization problems.

4. Application

UWB positioning technology usually adopts distance positioning. According to the distance between the transmitting base station (Tag) and the receiving base station (Anchor), the relationship equation is constructed to solve the position information of the Tag. Usually, 3D positioning using UWB requires one Tag and 4 Anchors. Common UWB localization algorithms include Chan positioning algorithm, Taylor positioning algorithm and Kalman filter positioning.

4.1. Problem modeling

The group learning phase of the GLBO algorithm is used to explore the search space, while the selection phase ensures that promising solutions can be further exploited to find the exact location of the Tag through continuous iteration of the population.

For the three-dimensional positioning problem, the coordinates of the i-th Anchor base station are (x_i, y_i, z_i), and the distance from the Tag to the Anchor is measured by the device. The population of the GLBO algorithm is initialized as a point *(x, y, z)* in the three-dimensional space. When the distance from the individual in the population to the Anchor is equal to the distance from the Tag to the Anchor, then the coordinates of the individual are the coordinates of the Tag. However, there will be certain errors in practical applications, so the error can be regarded as the fitness, and the smaller the fitness, the more accurate the positioning. The fitness function is shown in equation (6) and equation (7), where R_i represents the distance from Tag to the *i-th* Anchor. This transforms a positioning problem into a minimal optimization problem.

$$f_i(x,y,z) = \left| \sqrt{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2} - R_i \right| \tag{6}$$

$$Fitness = \sum\nolimits_{i=1}^{4} f_i(x,y,z) \tag{7}$$

4.2. Experiment

In this experiment, Anchor base stations are placed in four corners of the room and the coordinates of Anchors are listed in Table 3, and the Tag base station sends signals to all Anchor base stations to measure the distance between them, the measured distance are shown in figure 1. Because the measured distance value contains noise and errors, it needs some processing before it can be used in the GLBO algorithm. As shown in figure 2, in this experiment, the moving average method is used to process the data, and then the available data can be obtained by subtracting the hardware system error.

Table 1. the performance of GLBO and several state-of-the-art evolutionary algorithms on ten 10-dimensional test functions.

Functions	**GLBO**	**AGSK**	**IMODE**	**MadDE**
F1	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
F2	5.81E+01	2.32E+01	1.78E+00	**3.75E-01**
F3	1.36E+01	8.91E+00	1.10E+01	1.13E+01
F4	**0.00E+00**	4.33E-01	1.04E-05	1.29E-01
F5	2.08E-01	2.08E-01	3.38E+00	**0.00E+00**
F6	**2.51E-02**	5.71E-02	2.95E-01	8.02E-02
F7	**3.54E-07**	1.39E-05	7.66E-02	3.72E-03
F8	**0.00E+00**	**0.00E+00**	4.56E-02	1.12E+01
F9	**0.00E+00**	**0.00E+00**	2.74E+01	**0.00E+00**
F10	**1.00E+02**	1.00E+02	1.26E+02	3.98E+02

Table 2. the performance of GLBO and several state-of-the-art evolutionary algorithms on ten 20-dimensional test functions.

Functions	**GLBO**	**AGSK**	**IMODE**	**MadDE**
F1	1.09E+03	**0.00E+00**	**0.00E+00**	**0.00E+00**
F2	2.44E+00	1.32E+01	5.85E+00	**1.25E-01**
F3	**2.02E+01**	2.06E+01	2.14E+01	2.04E+01
F4	9.99E-01	1.27E+00	8.91E-01	**5.14E-01**
F5	1.13E+01	1.31E+01	6.19E+01	**4.29E+00**
F6	4.04E-01	**1.81E-01**	5.07E-01	1.96E-01
F7	5.45E-01	**3.12E-01**	1.70E+01	4.17E-01
F8	8.11E+01	1.00E+02	**6.85E+01**	1.00E+02
F9	**1.80E+01**	1.00E+02	1.00E+02	1.00E+02
F10	**3.99E+02**	4.00E+02	4.00E+02	4.14E+02

Table 3. Anchor coordinates.

Axis	Anchor A1	Anchor A2	Anchor A3	Anchor A4
x-axis	0	0	6000	6000
y-axis	0	6000	6000	0
z-axis	3000	3000	3000	3000

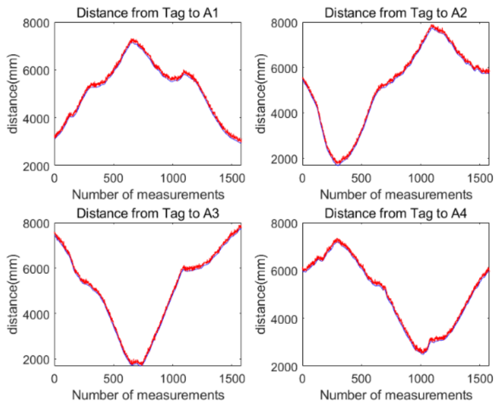

Figure 1. Distance from Tag to Anchor.

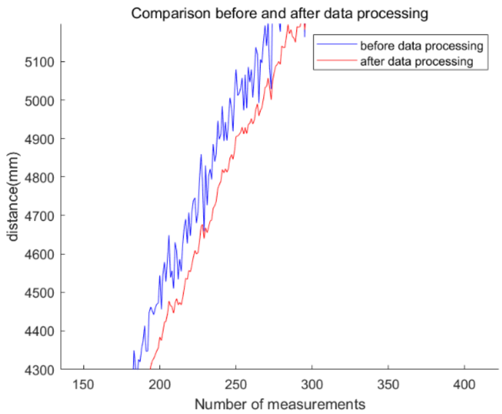

Figure 2. Comparison before and after data processing.

4.3. Experimental results

The results of UWB positioning using the GLBO algorithm are shown in the figure 3, in which the blue point is the true moving track of the Tag, and the red point is the positioning track of the GLBO algorithm. Figure 4 is the positioning error of the GLBO algorithm. It can be seen that the positioning trajectory and the real trajectory are highly coincident, which shows that the GLBO algorithm has high accuracy for UWB positioning.

5. Conclusion

Inspired by the principle of teaching students in accordance with their aptitude, this paper proposed a novel metaheuristic optimization algorithm. The population is divided into three groups according to their fitness, and each group adopts different learning strategies according to its characteristics, so that each group can get the greatest development. Through the test and the comparison with the recent excellent optimization algorithm to illustrate the performance of the GLBO. The results show that GLBO has high accuracy in finding the global optimal solution. And GLBO is applied to the UWB positioning problem, and good results are obtained.

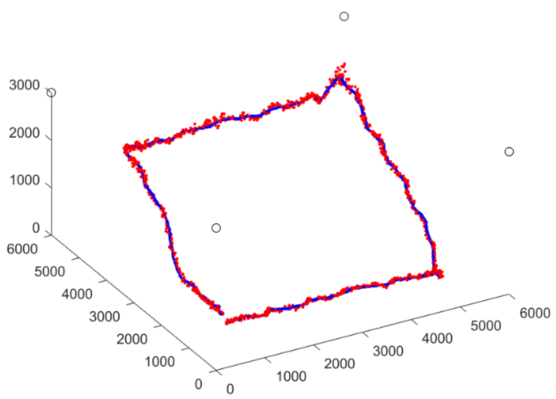

Figure 3. positioning track of the GLBO algorithm

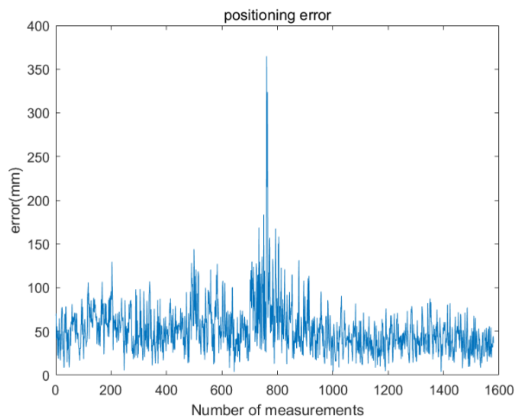

Figure 4. positioning error of the GLBO algorithm

Acknowledgments
The work was supported by the Innovation and Entrepreneurship Training Program for College Students (202110637008).

References

[1] Storn R and Price K. (1997) Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *Journal of Global Optimization*, **11**: 341–59.
[2] Blackwell T M, Kennedy J and Poli R. (2007) Particle Swarm Optimization. *Swarm Intelligence*, **1**: 33-57.
[3] Dorigo M, Maniezzo V and Colorni A. (1996) Ant system: Optimization by a colony of cooperating agents. *IEEE Trans. Syst. Man Cybern B*, **26**: 29–41.
[4] Yang X and Deb S, 2009. Cuckoo Search via L{é}vy Flights. In: 2009 World Congress on Nature & Biologically Inspired Computing. Coimbatore. 210–14.
[5] Yang X, 2010. Firefly Algorithm, Stochastic Test Functions and Design Optimisation. https://doi.org/10.48550/arXiv.1003.1409.
[6] Ali Wagdy M and Anas A H, 2020. Evaluating the Performance of Adaptive Gaining- Sharing Knowledge Based Algorithm on CEC 2020 Benchmark Problems. In: 2020 IEEE Congress on Evolutionary Computation (CEC). Glasgow. 1-8.
[7] Subhodip B, Debanjan S and Shuvodeep D, 2021. Improving Differential Evolution through Bayesian Hyperparameter Optimization. In: IEEE Congress on Evolutionary Computation. Krakow. 832-40.
[8] Karam M, Saber M and Ripon K, 2020. Improved Multi-operator Differential Evolution Algorithm for Solving Unconstrained Problems. In: 2020 IEEE Congress on Evolutionary Computation (CEC). Glasgow. 1-8.