Research on the Ideological and Political System of Logistics System Simulation Course in the Context of Artificial Intelligence

Yan Tang(✉), Anqi Cui, and Yang Hu

School of Management, Tianjin University of Technology, Tianjin 300384, China
tangyan@email.tjut.edu.cn

Abstract. Universities are trying to integrate algorithms into the teaching of logistics system simulation courses. However, students have difficulty in understanding algorithms due to the complex teaching formulas and computational processes. The main purpose of this paper is to solve the current teaching dilemma of algorithms. Taking the representative partial example group algorithm of optimization algorithm as an example, introducing the detailed steps and modeling process of the partial example group algorithm. Based on this, the algorithm is applied to the actual logistics system simulation modeling, and the computer is used with classroom teaching to help students understand the algorithm quickly. This paper shows that the use of algorithms can largely improve the efficiency and optimize the results of logistics system simulation, and provides a paradigm for teaching similar undergraduate teaching contents and knowledge points.

Keywords: Artificial Intelligence · Logistics System Simulation · Ideological and political education · Partial Particle Swarm Optimization Algorithm

1 Introduction

Artificial intelligence refers to making computers simulate certain human thought processes and intelligent behaviors, and its related technologies are distributed in various fields, such as robotics, language recognition, image recognition and language processing [1]. In recent years, more and more experts have applied AI technology to the field of education, which has largely changed the original education model. Logistics system simulation courses require students to apply their knowledge flexibly to solve real-world problems. In the standardized process of problem solving, analysis and modeling have high requirements for students' knowledge mastery and flexible application, while problem solving requires students to be able to skillfully apply professional tools and software to achieve secondary programming, computational solution and parametric analysis [2]. If teachers emphasize too much on theoretical learning and neglect tool solving in teaching or students' learning process, students are bound to be able to build mathematical models of applied problems, but are at a loss to solve them [3]. In the long run, students' ability to solve practical problems will be weakened, which will eventually affect the teaching objectives and ability development required by the curriculum [4].

F. Huang et al. (Eds.): ICAIE 2023, AHCS 15, pp. 519–525, 2024.
https://doi.org/10.2991/978-94-6463-242-2_64

Under the influence of the development trend in the field of computer simulation, combined with the previous teaching work and teaching and research project accumulation [5], this paper tries to introduce computer simulation technology into the ideological and political teaching development of optimization algorithm to help students quickly grasp the advanced calculation steps of the principle of algorithm.

2 Introduction of Partial Particle Swarm Optimization Algorithm

2.1 Steps of Partial Particle Swarm Optimization Algorithm

The partial particle swarm optimization algorithm is derived from the group behavior of birds in the biological world. The feasible region of the research problem is compared to the space of bird swarm flight, and the extreme point of the objective function is regarded as the position of "food". In this space, each "particle" searching for the optimal solution alone is analogous to a bird. In the search process, each particle constantly adjusts the value and variation of the decision variable, that is, the direction and speed of the bird's flight. More importantly, all particles will synchronously update/share the optimal solution within the known search range during the search process.

Taking the optimization problem of a two-dimensional variable as an example:

$$\left.\begin{array}{c} maxG = f(x, y) \\ S.t.\mathrm{l_x} \le x \le \mathrm{u_x} \\ \mathrm{l_y} \le y \le \mathrm{u_u} \end{array}\right\} \tag{1}$$

The partial particle swarm optimization algorithm to solve the maximization problem can be simply divided into two steps, the first step is initialization. In this step, for each of the particles in the particle swarm. Each particle randomly generates its position in the feasible region and its position change. The speed of, i.e.:

$$\left.\begin{array}{c} PositionX(i) = Random(l_x, u_x) \\ VelocityX(i) = Random(0,) \\ PositionY(i) = Random(l_y, u_y) \\ VelocityY(i) = Random(0, s_y) \end{array}\right\} \tag{2}$$

Then, the objective function of the position of the particle, i.e., the fitness function, is calculated value.

$$\mathrm{Fitness(i)} = \mathrm{G(i)} = \mathrm{f(PositionX(i), PositionY(i))} \tag{3}$$

At this point, the current personal optimal solution of the initialized particle and the objective function.

values are:

$$\left.\begin{array}{c} PersonalBestFitness(i) = G(i) \\ PersonalBestPositionX(i) = PositionX(i) \\ PersonalBestPositionY(i) = PositionY(i) \end{array}\right\} \tag{4}$$

Subsequently, the optimal objective function value among all the current particles is determined, with and the corresponding position coordinates:

$$\left.\begin{array}{c} GlobalBestFitness = PersonalBestFitness(k) = \\ max\{\text{PersonalBestFitness(i)}\} \\ GlobalBestPositionX\ = PersonalBestPositionX\,(k) \\ GlobalBestPositinY\ = PersonalBestPositionY\,(k) \end{array}\right\} \tag{5}$$

Step 2 of the algorithm is an iterative search process for the particles. Specifically specifically, the particle will adjust its velocity based on the shared information of the population optimal solution, as well as its current In particular, the particle will adjust its velocity and update its position according to the shared population optimal solution information, as well as its current state and velocity. Specifically, each particle will update its position according to its current velocity (with a certain inertia) and its optimal position. In particular, each particle will calculate its velocity based on its current velocity (with a certain inertia), its own position preference of the optimal solution and the position preference of the population optimal solution: the position preference of its own optimal solution and the position preference of the group optimal solution.

After calculating the velocity of the particle, we can obtain the unit time interval between the position after the interval is:

$$\text{PositionX(i)} - \text{PositionX(i)} + \text{VelocityX(i)} \tag{6}$$

$$\text{PositionY(i)} = \text{PositionY(i)} + \text{VelocityY(i)} \tag{7}$$

Then, the target function value of the particle at the new position is recalculated according to Eq. (3). And based on this, the current optimal solution of each particle is updated and the optimal solution of the population. The particle swarm algorithm will repeat the calculation shown in step 2 until it finds the Optimal solution.

2.2 Simulation Modeling of the Solution Process of Partial Particle Swarm Algorithm

The population intelligence algorithm represented by the biased particle swarm algorithm involves decision making process of multiple independent decision objects and the information between the objects interactions, they are more suitable for modeling and simulation of multi-intelligences with which they are naturally connected Simulation; realizing its problem solving process. In the multi-intelligence modeling approach, each particle will be set as an intelligent object with independent decisions. It records its own search trajectory and search results, while continuously exchanging information with other particles.

The behavior of each particle in the biased particle swarm algorithm is relatively simple, simply adjusting stressfully to the population optimal solution and its own optimal solution. However, the interaction of information about the population optimal solution between the swarm of particles enables the swarm of particles to computationally solve complex optimization problems.

3 Development of Simulation Courseware for Biased Particle Swarm Algorithms

3.1 Examples of Complex Optimization Problems

For classroom demonstration of the algorithmic capabilities of the biased particle swarm algorithm, the teaching the following two problems are considered. The first problem is a deterministic, continuum problem.

$$\left.\begin{array}{c} maxG = -\left[x^2 + y^2\right] \\ s.t. - 100 \le x \le 100 \\ -100 \le y \le 100 \end{array}\right\} \tag{8}$$

The second problem adds randomness to the first problem factor ρ (subject to the standard 0–1 uniform distribution).

$$\left.\begin{array}{c} maxG = -\left[x^2 + y^2\right] \bullet \rho \\ s.t. - 100 \le x \le 100 \\ -100 \le y \le 100 \\ \rho \sim Uniform(0, 1) \end{array}\right\} \tag{9}$$

Whether it is the first problem or the second problem, we can get the same optimal solution and value:

$$G^* = 0, (x^* = 0, y^* = 0) \tag{10}$$

However, compared to the first problem, the second problem has several local minima, which makes the algorithm more laborious to search.

3.2 Code Development for Multi-Intelligence Simulation

In order to make the simulation and algorithm demonstration more intuitive in the classroom, Netlogo 6.0.1 was chosen as the software for the multi-intelligence simulation. Compared to multi-intelligence modeling and simulation tools such as Swarm, Repast-Netlogo software platform is simpler and smarter to develop than Swarm, Repast and other multi-intelligence modeling and simulation tools. The second optimization problem shown in Eq. (10), for example, can be implemented in the Netlogo software platform with only one line of code.

ask patches.

set Fitness.

-(distancexy 0 0) * (random-float 1.0)].

To enhance the visibility of the problem solution space, different grayscales are used to display the objective function values/Fitness values at different locations.

3.3 The Main Content of Simulation Courseware

Based on the development of the simulation program, the design of intelligent optimization algorithm class the teaching information of the class is as follows:

(1) Teaching objectives. In the undergraduate teaching of "Logistics System Simulation," the common intelligent optimization algorithms for solving complex application problems are introduced. The purpose of teaching includes introducing the algorithm ideas, calculation steps and result interpretation of specific intelligent optimization algorithms, and introducing the applicability of intelligent optimization algorithms in the field of operations research.
(2) Teaching content 1. In the actual classroom teaching and demonstration, the calculation process of partial particle swarm optimization algorithm is introduced by studying the visualization and computer simulation of the solution process of problem 1.

Figure 1 shows the global optimal solution of partial particle swarm in the calculation process is given.

The process of change (formula (5)). It can be seen from Fig. 1 that the global optimal solution will soon be converge to the optimal solution of the application problem.

(3) Teaching element 2: Based on the simulation demonstration of research problem 1, we reconstruct research problem 2. The problem is re-constructed based on the simulation demonstration of Research Problem 1, and is solved by the partial particle swarm algorithm. The problem is solved computationally by the partial particle swarm algorithm.

However, the application problem 2 shown in Eq. (9) is generated several times, and the partial particlswarm algorithm simulation also shows the following results. As shown in Fig. 2, the particle swarm quickly converges to the non-origin position the particle swarm quickly converges to a non-original location, i.e., it stays at a local optimal solution. Although the objective function of this suboptimal solution value (=-0.112) is very close to the true optimal solution, but it is not the optimal solution.

The partial particle swarm algorithm does not necessarily calculate the optimal solution to the applied problem, but it can obtain an efficient solution. The logistics system simulation process is difficult to find the optimal solution due to its diversity. The algorithm can improve the operational efficiency of logistics system simulation and find the

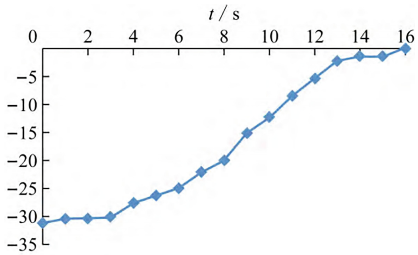

Fig. 1. Apply the change process of Global Best Fitness in question 1

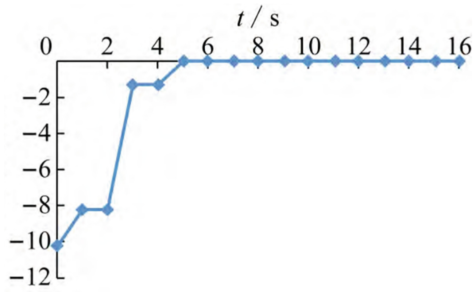

Fig. 2. Apply the calculation of Problem 2 to the procedure of Calculation 2

relatively optimal solution. The algorithm's optimization practice exploration provides suggestions for teaching courses with similar content. Scientific rigor is also reflected in the actual teaching process.

4 Conclusion

This paper incorporates the partial particle swarm algorithm into the teaching of the logistics system simulation course. Through detailed explanation and practical application of the partial particle swarm algorithm, it promotes students' understanding of the algorithm. The algorithm can provide a relatively optimal solution for the logistics system simulation process, and also provides a reference solution for teaching other similar courses.

Acknowledgement. Supported by Task Book for the Construction of the Special Course Reform of "Curriculum Ideology and Politics" at Tianjin University of Technology (KG22–01).

References

1. Munuzuri J, Cortes P, Grosso R, et al. Teaching heuristic methods to industrial engineers: A problem-based experience [J] INFOR MS Transactions on Education,2016, 16(2) : 60–67.
2. Lojo M P. Improving undergraduate student performance on the little field simulation [J]. INFORMS Transactions on Education,2015,16(2): 54-59.
3. Y. Cui. Information management technology of education system in the context of Internet+ technology [J]. Information Technology, 2023(05): 142-147. 10.13274.
4. Bai Shuang, Liang Chen. Teaching exploration and practice of "deep learning algorithms and implementation" graduate course [J]. Industrial and Information Technology Education,2023(05):21-25.
5. Peng Liang, David Li. The vision, technical risks and responses of big data to promote educational governance [J]. Educational Science Exploration,2023,41(02):74-80.