An improved grey wolf optimization for solving TSP

Bin Lei[a], Hailong Liua[*], Wanying Wang[a], Guannan Zhong[a], Huo Chai[b]
[a]Mechatronics T&R Institute, Lanzhou Jiaotong University, Lanzhou 730070, Gansu, China; [b]School of Traffic and Transportation, Lanzhou Jiaotong University, Lanzhou 730070, Gansu, China

ABSTRACT

An improved grey wolf optimization (IGWO) is proposed for the travel salesman problem. Based on traditional grey wolf optimization, a genetic gene sequence was introduced for the first time to initialize the population to improve population quality. Subsequently, the cosine adaptive function is used to balance the global and local search abilities. Finally, a distance heuristic factor is introduced into the update strategy to enhance the local search capability and improve the robustness of the algorithm. Several cases were randomly selected from the TSPLIB database for the simulation experiments, and it was found that the IGWO has higher stability and accuracy under the same conditions.

Keywords: Traveling salesman problem, grey wolf optimization, cosine adaptive function, update strategy

1. INTRODUCTION

In the real world, there are many combinatorial optimization problems, such as transportation planning, computer network distribution and logistics location selection, which need to choose optimal solutions under limited constraints[1]. As one of the typical representatives of combinatorial optimization problems, Traveling Salesman Problem (TSP) has been a hot research topic for a long time. Actually, solving the traveling salesman problem can be understood as completely arranging a number of known coordinate points in a certain reference frame. Early researchers solved it by precise algorithms such as branch and bound[2], linear programming[3] and dynamic programming[4]. However, with the increase of coordinate points, the feasible scheme of arrangement will increase exponentially, and the precise algorithm can't work out an effective feasible scheme in a limited time. Heuristic algorithms, such as genetic algorithm[5], simulated annealing algorithm[6], whale optimization algorithm[7] have attracted the attention of scholars at home and abroad because of their advantages of strong development and fast operation in the application of traveling salesman problem.

2. DESCRIPTION AND MATHEMATICAL MODEL OF TSP

The essence of the TSP can be understood as the traveler starts from the departure city x_i, crosses several cities with defined coordinates $x_2 \sim x_n$ and finally returns to the departure city x_i, form a complete arrangement, which is the feasible solution, and all the arrangement schemes with the shortest distance are the optimal solutions[8]. Therefore, TSP can be modeled as follows according to its characteristics:

$$\min D = \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij} \cdot x_{ij} \tag{1}$$

$$s_g t_g \begin{cases} \sum_{i=1}^{n} x_{ij} = 1, i \in n \\ \sum_{j=1}^{n} x_{ij} = 1, j \in n \\ \sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij} \le |s| - 1, s \subset n, s \ne \varnothing \\ x_{ij} \in \{0,1\}, i \in n, j \in n \end{cases} \tag{2}$$

[*] lhl20160902@163.com

Third International Conference on Computer Science and Communication Technology (ICCSCT 2022)
edited by Yingfa Lu, Changbo Cheng, Proc. of SPIE Vol. 12506, 125060H

In the above model, equation (1) denotes the objective function, where D denotes the path length, c_{ij} denotes the distance between i and j, and equation (2) is the constraint, indicating that each city is traversed only once and no sub-loop is generated, where if $x_{ij}=1$ then the traveller passes through cities i and j, if $x_{ij}=0$ then the traveller does not pass through cities i and j.

3. GREY WOLF OPTIMIZATION

3.1 Algorithm introduction

The Grey Wolf Optimizer (GWO) is a computational model, which simulates the hunting process of the Grey Wolf group in nature[9]. In the natural environment, wolves follow a certain hierarchy of wolves, from high to low as α, β, δ and ω, Wolves finally achieve the goal of catching prey through three steps: tracking, encircling and attacking.

In the GWO, the hunting process of the gray wolf is defined as follows:

$$\vec{D}=\left|\vec{C}\cdot\overrightarrow{X_P(t)}-\overrightarrow{X(t)}\right| \tag{3}$$

$$\overrightarrow{X(t+1)}=\overrightarrow{X_P(t)}-\vec{\mathrm{A}}\cdot\vec{\mathrm{D}} \tag{4}$$

$$\vec{A}=2a\cdot\vec{r_1}-a \tag{5}$$

$$\vec{C}=2\vec{r_2} \tag{6}$$

$$a=2(1-\frac{t}{T}) \tag{7}$$

$$\begin{cases}\overrightarrow{D_\alpha}=\left|\overrightarrow{C_1}\cdot\overrightarrow{X_\alpha}-\vec{X}\right| \\ \overrightarrow{D_\beta}=\left|\overrightarrow{C_2}\cdot\overrightarrow{X_\beta}-\vec{X}\right| \\ \overrightarrow{D_\delta}=\left|\overrightarrow{C_3}\cdot\overrightarrow{X_\delta}-\vec{X}\right|\end{cases} \tag{8}$$

$$\begin{cases}\overrightarrow{X_1}=\overrightarrow{X_\alpha}-\overrightarrow{A_1}\cdot(\overrightarrow{D_\alpha}) \\ \overrightarrow{X_2}=\overrightarrow{X_\beta}-\overrightarrow{A_2}\cdot(\overrightarrow{D_\beta}) \\ \overrightarrow{X_3}=\overrightarrow{X_\delta}-\overrightarrow{A_3}\cdot(\overrightarrow{D_\delta})\end{cases} \tag{9}$$

$$\overrightarrow{X(t+1)}=(\overrightarrow{X_1}+\overrightarrow{X_2}+\overrightarrow{X_3})/3 \tag{10}$$

where $\vec{D}$ denotes the distance between the individual wolf and the prey, $\overrightarrow{X_P(t)}$ denotes the location coordinates of the prey, $\overrightarrow{X(t)}$ denotes the location coordinates of the wolf, $\vec{A}$ and $\vec{C}$ are a vector of coefficients. $\vec{r_1}$ and $\vec{r_2}$ are a random vector modulo between [0,1] and an adaptive function that decreases linearly from 2 to 0. When $a\in[0,1]$, the grey wolf performs a local search, $a\in(1,2]$, the wolf performs a global search. t and T denote the number of current iterations and the maximum number of iterations, respectively. $\overrightarrow{D_\alpha}$, $\overrightarrow{D_\beta}$ and $\overrightarrow{D_\delta}$ denote the distance between α, β, δ wolves and other individual wolves.

3.2 Improvements to the grey wolf optimization

GWO has attracted much attention for its fast convergence speed, few parameters and simple structure in many fields[10[. However, due to GWO's dependence on the initial population, it's easy to fall into local, this improved algorithm is based on GWO, and improved from three aspects: the selection of the initial population, the adaptive function and update strategy.

3.2.1 Improving Initial Populations. Different from the traditional GWO which randomly generates the initial population, the gene sequence of the genetic algorithm is introduced during the initialization of the population, and the high-quality population is selected through selection, crossover and mutation. Assuming that there are x cities in the TSP, the algorithm population of N gray wolves would represent N candidate solutions, in which each gray wolf consists of a gene sequence interconnected by x cities.

3.2.2 Improved Adaptive Convergence Factor. The traditional GWO adopts linear descent in the adaptive process, which is not conducive to global search. Therefore, the cosine convergence function is introduced to improve the global search ability of the population in the early stage and accelerate its convergence speed in the later stage.

$$\begin{cases} a = 1 + [\cos((t-1)\pi / (T-1))]^n \\ t \le \frac{1}{2}T \\ a = 1 - \left|\cos((t-1)\pi / (T-1))\right|^n \\ \frac{1}{2}T < t \le T \end{cases} \tag{11}$$

where $n \in [0,5]$.

3.2.3 Improving the Way Populations Are Renewed. Because the traditional GWO does not play the role of the current high-quality solution in the iterative updating process, a distance heuristic factor is introduced to assign weights to the high-quality solution, so as to drive the population to approach the optimal solution and prevent it from falling into a local optimum.

$$\overrightarrow{X(t+1)} = \left(\frac{\overrightarrow{X_\alpha}}{\left|\overrightarrow{D_\delta}\right|} + \frac{\overrightarrow{X_\beta}}{\left|\overrightarrow{D_\beta}\right|} + \frac{\overrightarrow{X_\delta}}{\left|\overrightarrow{D_\alpha}\right|}\right) \cdot \frac{\left|\overrightarrow{D_\alpha}\right| + \left|\overrightarrow{D_\beta}\right| + \left|\overrightarrow{D_\delta}\right|}{3} \tag{12}$$

3.2.4 Algorithm Flow. For express the algorithm flow more conveniently, we will express the flow in Figure 1.

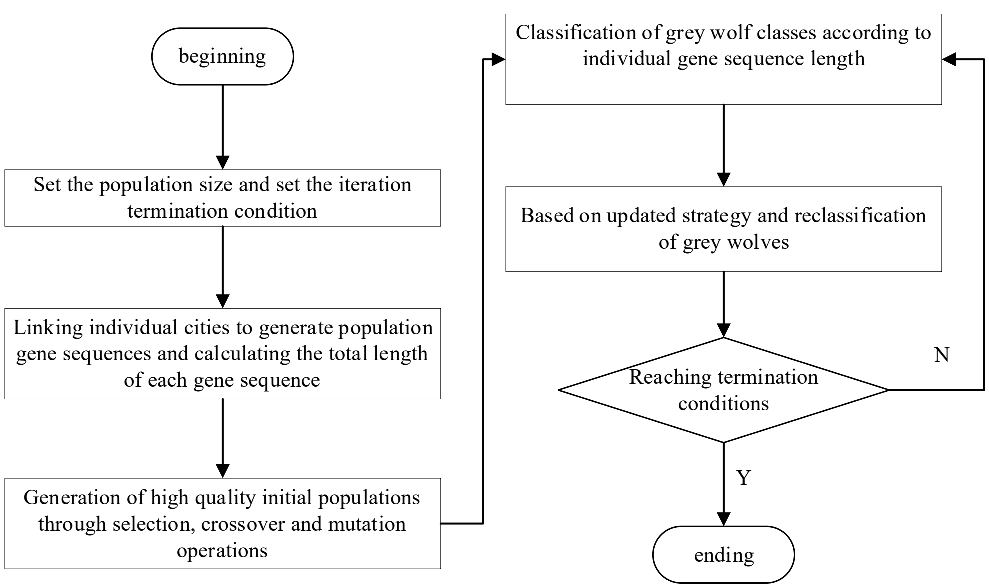


Figure 1. Algorithm flow chart.

4. EXPERIMENTAL ANALYSIS

For the sake of verify the effectiveness of the improved algorithm, several examples are randomly selected from the TSPLIB database for simulation analysis, and the proposed improved algorithm was compared with the genetic algorithm and simulated annealing algorithm, which are widely used for TSP at present.

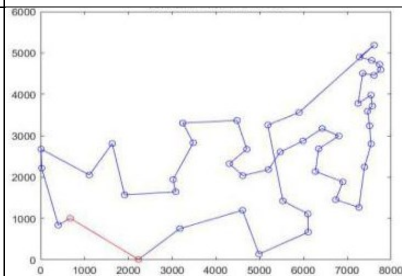
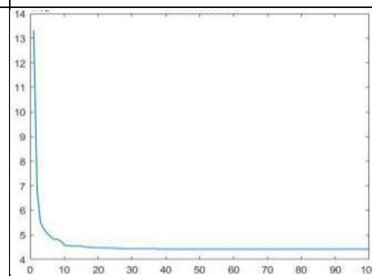
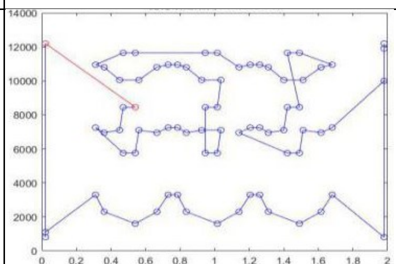
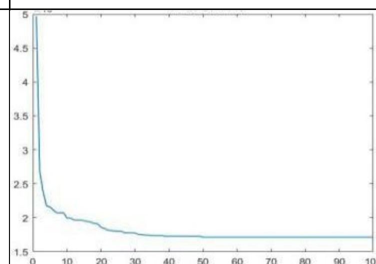
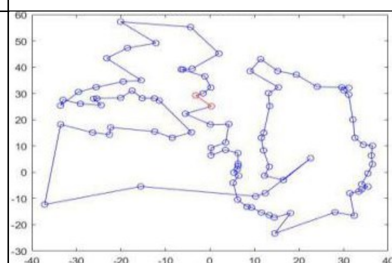
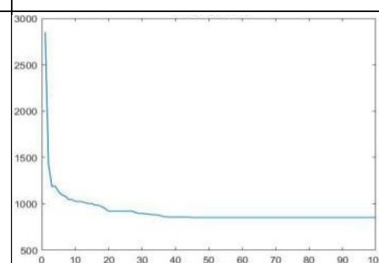
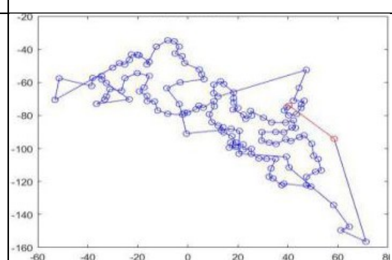
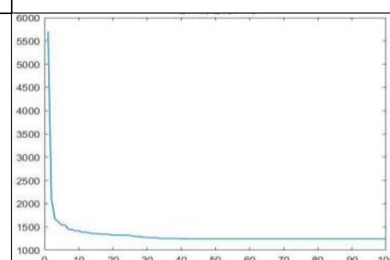
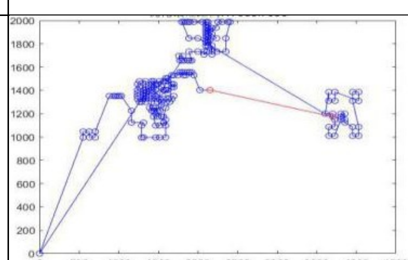
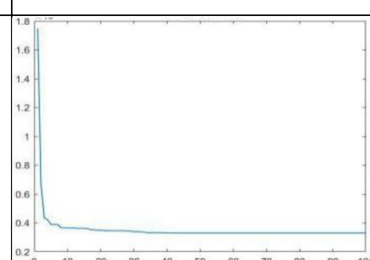
The results are compared with the optimal solution, mean, standard deviation and average time taken after 30 consecutive executions of each algorithm, with the results retained to four decimal places. The experimental data are detailed in Table 1.

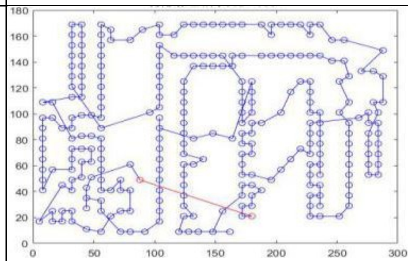
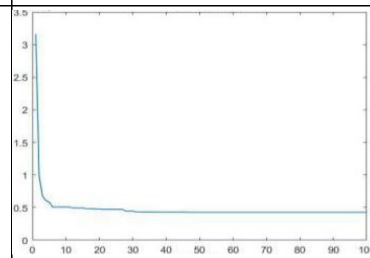
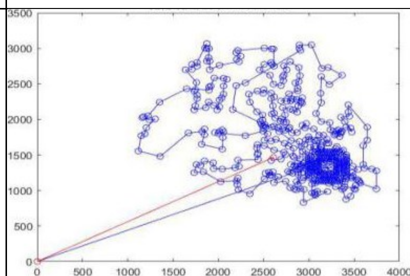
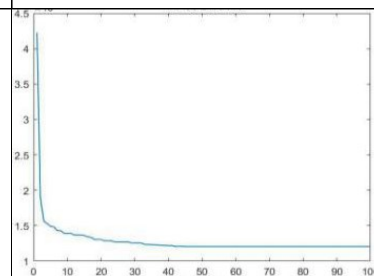
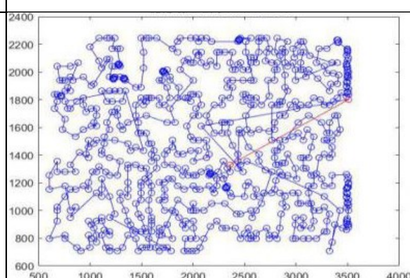
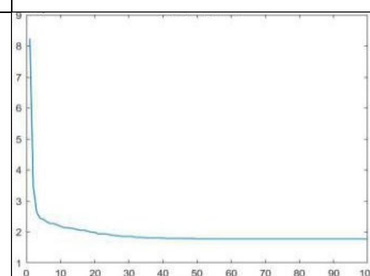
Table 1. Comparison of experimental results.

Cases	Methods	Optimum	Average	Standard deviation	Average time spent (s)
ATT48	IGWO	33444.0405	33956.9247	217.1600	7.8506
	GA	37981.9947	46431.1877	3735.6227	1.5737
	SA	36093.9511	39169.1110	2356.8665	87.8603
PR76	IGWO	117823.3285	120623.9764	1304.6328	7.6546
	GA	145683.3729	171325.2896	13069.5142	3.3747
	SA	129855.0567	146410.4523	9581.0888	141.7006
GR96	IGWO	697.9733	711.9301	6.9091	9.3748
	GA	815.5928	936.5920	51.5084	3.9731
	SA	676.7912	815.9422	76.2375	76.2375
GR137	IGWO	956.6556	962.5351	2.3807	13.5496
	GA	1486.2578	1673.1033	95.6635	5.0318
	SA	1116.4836	1237.3701	73.4981	260.6727
D198	IGWO	23024.7311	23146.9250	86.2047	16.0132
	GA	47576.9933	54197.8338	3621.4246	6.2114
	SA	31992.6467	35232.9721	1914.4961	376.8255
A280	IGWO	2709.1786	2709.1786	0.0000	25.9077
	GA	9710.6770	10666.6649	457.1040	10.1864
	SA	5931.7739	6477.7257	354.3674	527.1442
D493	IGWO	108071.5727	110371.1229	789.5293	45.5010
	GA	155078.7760	162325.8562	5176.5141	19.0277
	SA	97010.0685	100887.2970	2490.3894	957.6644
U724	IGWO	153954.9827	155950.7540	759.9781	73.3198
	GA	305475.0081	320270.7757	7133.5143	32.1986
	SA	164409.9699	179785.8008	7979.2320	1426.9803

In order to demonstrate the effectiveness of IGWO more intuitively, the path diagram and convergence diagram of the simulation results are given below. The experimental results are shown in Table 2.

Table 2. Simulation experiment.

Case	**Figure**	
	Route map	Convergence diagram
ATT48		
PR76		
GR96		
GR137		
D198		

Case	Figure	
	Route map	Convergence diagram
A280		
D493		
U724		

Analysis of the experimental data is as follows.

(1) In terms of the accuracy of the solution, IGWO is superior to other intelligent algorithms in terms of the optimal solution and average value, except GR96 and D493, whose optimal solution was second only to SA.

(2) In terms of robustness, the deviation between the results of IGWO and the standard deviation of 30 consecutive solutions is the smallest among the listed algorithms, indicating that GWO is more robust than the other two algorithms.

(3) In terms of time complexity, according to the average solution time of each algorithm, although GA takes the least time among the listed algorithms, for small-scale problems, the time change is basically smooth, while the time for solving D493 and U724 is significantly increased; It takes the longest time to solve SA, and with the increase of the number of target nodes, its solution time is also increasing.

In conclusion, compared with the simulated annealing algorithm and genetic algorithm, IGWO algorithm has certain advantages in solving TSP in terms of precision, robustness, time complexity and convergence.

5. CONCLUSION

In this paper, aiming at the TSP, an improved IGWO algorithm was proposed. In the initial stage, introducing gene sequence to screen high quality population. Then, according to the cosine adaptive function, the global search ability is improved in the early stage, and the speed is accelerated in the late stage. By adding distance heuristic to the update strategy, the local search ability of the algorithm is further enhanced, and the robustness of the algorithm is also improved. The experimental results show that, under the same conditions, IGWO has higher solution accuracy than genetic algorithm and faster solution speed than simulated annealing. With the increase of the scale of the problem, the solution time of IGWO becomes smoother.

REFERENCES

[1] Odili, J. B., Noraziah, A. and Zarina, M., "A comparative performance analysis of computational intelligence techniques to solve the asymmetric travelling salesman problem," Computational Intelligence and Neuroscience, 6625438-6625438 (2021).

[2] Dell, A. M., Montemanni, R., et al., "Algorithms based on branch and bound for the flying sidekick traveling salesman problem," Omega 104, 102493 (2021).

[3] Taillard, É. D., "A linearithmic heuristic for the travelling salesman problem," European Journal of Operational Research 297(2), 442-450 (2022).

[4] Li, X. L., et al., "Modified dynamic programming algorithm for optimization of total energy consumption in flexible manufacturing systems," IEEE Trans. Automation Science and Engineering 16(2), 691-705 (2019).

[5] Deng, Y. L., Xiong, J. X., et al., "A hybrid cellular genetic algorithm for the traveling salesman problem," Mathematical Problems in Engineering (12), 1-16 (2021).

[6] Absalom, E. S. E., Aderemi, O. A. and Marc, E. F., "Simulated annealing based symbiotic organisms search optimization algorithm for traveling salesman problem," Expert Systems with Applications 77, 189-210 (2017).

[7] Zhang, J., Hong, L., et al., "An improved whale optimization algorithm for the traveling salesman problem," Symmetry 13(1), 48 (2020).

[8] Víctor, P. V., Nodari, V., et al., "Simple constructive, insertion, and improvement heuristics based on the girding polygon for the Euclidean traveling salesman problem," Algorithms 13(1), 5-35 (2019).

[9] Elsisi, M., "Improved grey wolf optimizer based on opposition and quasi learning approaches for optimization: Case study autonomous vehicle including vision system," Artificial Intelligence Review, 1-24 (2022).

[10] Nadimi-Shahraki, M. H., Taghian, S. and Mirjalili, S., "An improved grey wolf optimizer for solving engineering problems," Expert Systems with Applications 166, 113917 (2021).